Notes on the Braid Isotopy Problem

Patrick DEHORNOY

Laboratoire de Mathématiques Nicolas Oresme, Université de Caen, 14032 Caen, France

E-mail address: dehornoy@math.unicaen.fr

URL: //www.math.unicaen.fr/∼dehornoy

1991 *Mathematics Subject Classification.* 20F36, 20C40, 57M60, 57M07

Key words and phrases. Braid groups; mapping class groups; Garside structure; free group automorphisms; laminations

Contents

CHAPTER I

Braids

1. The Braid Isotopy Problem

1.1. Material braids. We start from the intuitive idea of a material braid, as illustrated in Figure 1. What we shall take into account is not the metric aspects of such an object (length and thickness of the strands), but only *crossings*: which strand goes over which one, in which order, etc. Braid theory is a calculus of crossings.

FIGURE 1. A material braid

1.2. Braids in history. Braid diagrams appear in Gauss' notebooks (end of the 1700's, beginning of 1800's), without any theory.

The first mathematical use seems to be in Hurwitz' works, around 1900: not a theory of braids, but a description of an action of braids on sequences of elements of a group under conjugacy (the 'Hurwitz action").

Braids themselves have been investigated by E. Artin around 1925 [**1**], and then in [**2**].

1.3. Braid diagrams. The starting point for a mathematical theory of braids is the notion of a *braid diagram*. Informally, an n-strand braid diagram is a plane figure that consists of n curves, with an interruption of one of the two curves when they are close to cross, as shown on Figure 2, which corresponds to $n = 3$. The strands connect 3 points located on a vertical line on the left to 3 points located on a parallel vertical line on the right, and the rule is that strands keep moving from left to right[1]—a more precise definition will be given below.

FIGURE 2. A 3-strand braid diagram.

The *Braid Isotopy Problem* is the following problem:

Given two braid diagrams D, D', recognize whether one can *deform* D into D'.

Remark 1.1.— Solving the Braid Isotopy Problem means finding a uniform algorithm that works for all initial pairs of braid diagrams, not only for some particular pairs.

To make the problem precise, we must say what deforming a diagram means. To this end, we go to $\mathbb{R}^3$ and view an n-strand braid diagram as the projection of a 3D-object, called a *geometric braid*, the interruptions in the projections corresponding to one strand lying in front (the uninterrupted one), and the other lying behind (the one that is interrupted).

[1]U-turns are forbidden: if we allow them, we obtain a more general class of objects, called *string link diagrams*

Definition 1.2.— An n-strand *geometric braid* is a collection of n disjoint curves that connect the points $(1,0,0)$, ..., $(n,0,0)$ to the points $(1,0,1)$, ..., $(n,0,1)$ inside the band $\mathbb{R}^2 \times [0,1]$.

For such 3D-objects we have a natural notion of deformation of the ambient space.

Definition 1.3.— (i) Assume that X, Y are topological spaces and f, f' are homeomorphisms of X into Y. Then f, f' are called *isotopic* if there exists a continuous map F of $X \times [0,1]$ into Y satisfying $F(-,0) = f$ and $F(-,1) = f'$ and such that $F(-,t)$ is a homeomorphism for each t.

(ii) Two geometric braids b, b' are called *(ambient) isotopic* if there exists an isotopy F from the identity map of $\mathbb{R}^2 \times [0,1]$ to a homeomophism h of $\mathbb{R}^2 \times [0,1]$ that maps b to b' and is such that, for each t, the restriction of $F(-,t)$ to the planes $\mathbb{R}^2 \times \{0\}$ and $\mathbb{R}^2 \times \{1\}$ is the identity.

So two geometric braids are isotopic if there is a continuous deformation of the ambient space that deforms one into the other, by a deformation that keeps every point in the two bordering planes fixed—see Figure 3.

Remark 1.4.— An ambient isotopy between two geometric figures Γ, Γ' of $\mathbb{R}^3$—or, here, $\mathbb{R}^2 \times [0,1]$—is more than a homeomorphism of Γ to Γ'. As a topological space, any n-strand geometric braid is homeomorphic to the union of n copies of $[0,1]$, and so any two such geometric braids are homeomorphic. This does not mean that they are isotopic, because the homeomorphisms that connect them need not be the restriction of a homeomorphism of the ambient space.

FIGURE 3. Isotopy transforming the left braid diagram into the right one: the front strand (dotted) is moved to the right, while the crossing of the back strands is moved to the left.

We can now come back to braid diagrams and put a precise definition.

Definition 1.5.— Two braid diagrams D, D' are called *isotopic*, denoted $D \approx D'$, if they are projections of isotopic geometric braids, A *braid* is defined to be an isotopy class of braid diagrams. The set of all n-strand braids is denoted B_n.

Exercise 1.6.— Show that, if two geometric braids project to the same braid diagram, then they are isotopic. Deduce that a braid is also an isotopy class of geometric braids.

1.4. Why is the Braid Isotopy Problem interesting? Braids appear in many domains of mathematics. In that respect, solving the Braid Isotopy Problem is a preliminary step to any attempt to use braids in an effective way, *i.e.*, to develop braid algorithms. Indeed, defining braids to be the equivalence classes of braid diagrams—we shall see in Section 3 that there are good reasons to do so, *i.e.*, not to define a braid to be simply a braid diagram. In practice, braids are always *represented* by braid diagrams—just as integers are represented by numbers, *i.e.*, finite sequences of digits—so that the question of recognizing whether two diagrams are isotopic is just the question of recognizing of which braid one talks about. If the Braid Isotopy Problem were unsolvable, one could never know that, exactly as if an integer might be represented by different numbers and there were no way of comparing these numbers.

In particular, there are applications of braids to cryptography in which braids replace numbers. Clearly, one can use braids for such purpose only if one can recognize that two braids are equal.

Also, the Braid Isotopy Problem is connected with other mathematical problem, in particular the Knot and Link Isotopy Problems. The closure procedure of Figure 4 associates with every braid diagram a link diagram. Isotopic braid diagrams give isotopic link diagrams, but, conversely, non-istopic braid diagrams may have isotopic closures: the Link Isotopy Problem is (much) more difficult than the Braid Isotopy Problem.

Exercise 1.7.— Show that the closure of isotopic braid diagrams are isotopic link diagrams. Give a counter-example for the converse, *i.e.*, exhibit two braid diagrams that are not isotopic but whose closure are isotopic link diagrams. [Hint: Use the braids of Figure 5 and note that we do not consider the same notion of isotopy: in the case of links, no plane is supposed to be fixed.]

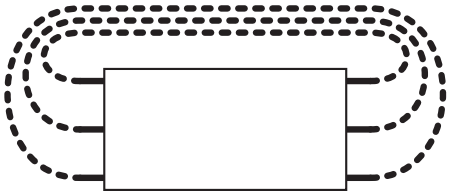


FIGURE 4. Closure of a braid diagram, to obtain a link diagram.

2. Basic remarks and first attempts

2.1. Decidability problems. The Braid Isotopy Problem belongs to the general family of decision problems. As every such problem, it consists of two half-problems of a different flavour.

Proving an isotopy is a priori easy or, more exactly, is easy to check when one has guessed a solution: if two diagrams are isotopic and we found a candidate for transforming the first diagram into the second, checking that the candidate-isotopy is actually an isotopy should be easy—as in the case of Figure 3.

Proving a non-isotopy seems more difficult: the fact that we cannot find an isotopy between two diagrams is not a proof that such an isotopy cannot exist, and it is not clear what can be done to prove this.

In such a context, the usual solution consists in trying to use an isotopy *invariant*, *i.e.*, a map I from the set of all braid diagrams to some space X with the property that, if D and D' are isotopic, then $I(D)$ and $I(D')$ are equal. Then, if we have $I(D) \neq I(D')$—in which case one says that I *separates* D and D'—we can deduce that D and D' are not isotopic. The question is to find an invariant (or a family of invariants) that separates all pairs of non-isotopic diagrams. Such an invariant (or family of invariants) is called *complete*.

2.2. The permutation of a braid. Let us look for simple isotopy invariants of braid diagrams.

The first example is the permutation of a braid. By definition, the strands of an n-strand braid diagram connect the n points $(i, 0, 0)$ to the n-points $(i, 1, 0)$. These points come with a natural numbering from 1 to n. Let $f_D(i)$ be the initial position of the strand that finishes at position i in a diagram D[2]. As the strands cannot touch each other in a geometric braid, f_D is a permutation of $\{1, ..., n\}$. By definition, all isotopies we consider leave the points of the planes $\mathbb{R}^2 \times \{0\}$ and $\mathbb{R}^2 \times \{1\}$ fixed, hence isotopic diagrams must give the same permutation: the maps $D \mapsto f_D$ is an isotopy invariant.

Hence, if we have $f_D \neq f_{D'}$, we can conclude that D and D' are not isotopic, see Figure 5.

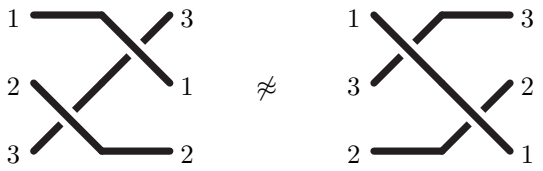


FIGURE 5. Two diagrams that give different permutations are not isotopic.

(As can be expected), the permutation is not a complete invariant: there exists nonisotopic diagrams with the same permutation. This will be proved subsequently, precisely when we have more powerful ways of proving non-isotopy results.

2.3. Counting crossings. Another idea for attaching a parameter to a braid diagram is to count crossing. This is *not* an isotopy invariant: Figure 6 shows that a diagram with 2 crossings may be isotopic to a diagram with 0 crossing.

However, an invariant is obtained when one considers the number of crossings *mod.* 2, or when one counts the crossings with a sign, for instance $+1$ when the front strand is the one that comes from the top, -1 when the front strand is the one that comes from the bottom—the proof that these are isotopy invariants follows from the more precise description of isotopy that will be given in Chapter **??**: for the moment, we are informal.

[2]It could seem more natural to consider the final position of the strand that starts at position i: the current choice is more convenient for the sequel, as it is the one that yields homomorphisms (and not antihomomorphisms) of the braid groups to the symmetric groups.

FIGURE 6. The number of crossings in a braid diagram is not an isotopy invariant.

2.4. Linking numbers. Still another invariant is obtained by considering the linking number of two fixed strands. Let D be an n-strand braid diagram, and let $1 \leqslant i < j \leqslant n$. When we consider the ith and the jth strands in isolation, we just see a sequence of oriented half-turns. Counting them with a sign that corresponds to the orientation yields an invariant, called the *linking number* of i and j.

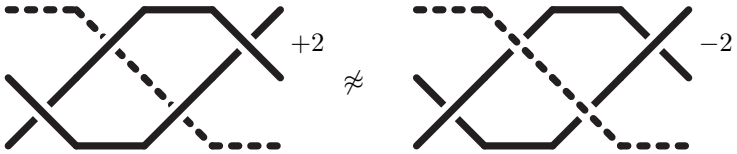


FIGURE 7. The linking number of the plain strands is $+2$ on the left, and -2 on the right, hence the diagrams are not isotopic.

Exercise 2.1.— Show that, if $N(D)$ denotes the algebraid number of crossings in the braid diagram D and if $L_{i,j}(D)$ denotes the (i,j)-linking number of the ith and the jth strands in D, then we have $N(D) = \sum_{i<j} L_{i,j}(D)$.

More generally, for each subset I of $\{1, ..., n\}$ with p elements, one obtains a projection from n-strand braid diagrams to p-strand braid diagrams by forgetting the $n-p$ strands that start at positions not in I. The isotopy type of the projection is an invariant.

In this way, we obtained a series of isotopy invariants for braid diagrams. Is this family complete, *i.e.*, can we separate any pair of non-isotopic diagrams? The answer is negative: Figure 8 displays two diagrams that are not isotopic—this will be proved in the sequel—but nevertheless have the same permutation, the same algebraic number of crossings, and the same linking numbers.

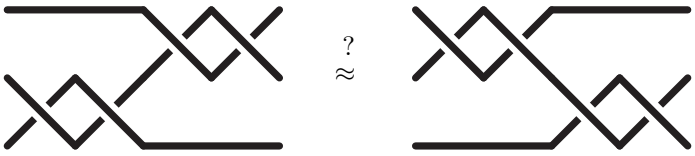


FIGURE 8. Two braid diagrams that are separated by none of the easy invariants described in this section.

The Braid Isotopy Problem is a problem of medium difficulty, not a trivial problem. More sophisticated methods are needed.

3. Braid groups

All solutions to the Braid Isotopy Problem rely on the fact that braids can be given a *group* structure. The existence of this structure is what makes the Braid Isopoty Problem much easier than the Link Isotopy Problem.

3.1. Product of braid diagrams. First we define a product of braid diagrams.

Definition 3.1.— (Figure 9) If D_1, D_2 are n-strand braid diagrams, their *product* D_1D_2 is defined to be the n-strand braid diagram obtained by concatenating the right ends of the strands of D_1 with the left ends of the strands of D_2.

The product of braid diagrams is the projection of a similar product on geometric braids; we occasionally denote $\Gamma_1\Gamma_2$ the product of two geometric braids Γ_1, Γ_2.

Remark 3.2.— Formally, if we insist that the strand of a braid diagram start on the line $y = 0$ and finish on the line $y = 1$, defining the product forces to rescale the diagrams by a factor $1/2$ along the y-axis.

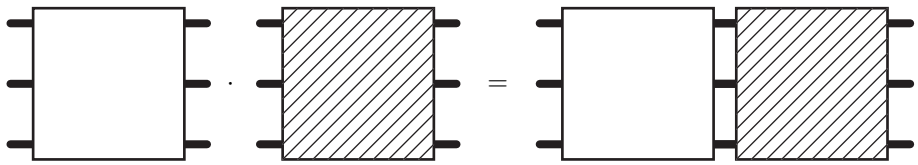


FIGURE 9. Product of two braid diagrams.

Lemma 3.3.— *The product of braid diagrams is compatible with isotopy.*

PROOF. Assume that D_1, D_1', D_2, D_2' are projections of geometric braids $\Gamma_1, \Gamma_1', \Gamma_2, \Gamma_2'$, respectively. If F_1 is an isotopy of Γ_1 to Γ_1' and F_2 is an isotopy of Γ_2 to Γ_2', on obtains an isotopy of $\Gamma_1\Gamma_2$ to $\Gamma_1'\Gamma_2'$ by applying (a rescaled version of) F_1 in $\mathbb{R}^2 \times [0, 1/2]$ and (a rescaled and translated version of) F_2 in $\mathbb{R}^2 \times [1/2, 1]$. The point is that, by definition, the two transformations agree on the median plane $\mathbb{R}^2 \times \{1/2\}$. □

Hence the product of diagrams induces a well-defined operation, also called product, on braids, *i.e.*, on isotopy classes of braid diagrams.

Lemma 3.4.— *The product of braids is associative, and the class of the trivial diagram is a neutral element.*

PROOF. Do it. □

3.2. Inverses. The advantage of considering braids, *i.e.*, isotopy classes, rather than braid diagrams becomes clear when one consider possible inverses for the braid product. Concatenating any diagram to a diagram that contains at least one crossing can never result in a diagram with no crossing, so no braid diagram except the trivial one admits an inverse. But, up to isoopy, inverses always exist.

Lemma 3.5.— *For each n-strand braid diagram D, let $\widetilde{D}$ denote the image of D in a vertical mirror. Then $D\widetilde{D}$ and $\widetilde{D}D$ both are isotopic to the trivial n-strand diagram.*

PROOF. (See Figure 10) Crossings pairwise cancel starting from the middle. □

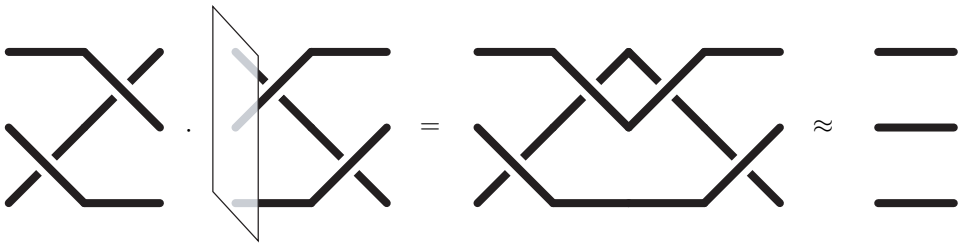


FIGURE 10. Inverse for the braid product.

We deduce

Proposition 3.6.— *For each n, the set B_n equipped with the product is a group: the n-strand braid group.*

3.3. The Braid Triviality Problem. The first benefit of the group structure on B_n is to reduce the Braid Isotopy Problem to the Braid Triviality Problem:

> Given one braid diagram D, recognize whether one can deform D into the trivial diagram.

Indeed, $D \approx D'$ is equivalent to $\widetilde{D}D' \approx 1$, where we use 1 as a generic notation for any trivial braid diagram. A priori, we may expect the Braid Triviality Problem (a one-variable problem) to be more simple than the Braid Isotopy Problem (a two-variable problem).

CHAPTER II

Presentation of braid groups

In order to possibly use the group structure of B_n to solve the Braid Isotopy Problem (or the Braid Triviality Problem), we need to characterize the group B_n in some way. To this end, we use a *presentation*. In this chapter, we first explain what a presentation is, and then describe a presentation of B_n.

1. Monoid presentations

Before explaining group presentations, we begin with monoid presentations, which are more simple, and which will be needed anyway.

1.1. Free monoids.

Definition 1.1.— A *monoid* is a set equipped with a binary operation that is associative and admits a neutral element.

Every group is a monoid. Conversely, $(\mathbb{N}, +, 0)$ is a monoid that is not a group. By default (and contrary to the latter example), the neutral element in a monoid is denoted 1 and the product is omitted. One often says "the monoid M" instead of "the monoid $(M, \cdot, 1)$".

Exercise 1.2.— Prove that a monoid contains one neutral element only.

We now construct particular monoids that will play an important role.

Definition 1.3.— Let S be a nonempty set. A *word* on S is a finite sequence w of elements of S, *i.e.*, a mapping of some interval $\{1, 2, ..., \ell\}$ to S. The parameter ℓ is called the *length* of w, and denoted $|w|$. The set of all words on S is denoted S^*. For u, v in S^*, the *product* of u and v, denoted $u \cdot v$ or uv, is the word w of length $|u| + |v|$ defined by $w(i) = u(i)$ for $i \leqslant |u|$ and $w(i) = v(i - |u|)$ for $i > |u|$. The unique word of length 0 is called the *empty word* and denoted by ε.

In the above context, S is often called an *alphabet*, and its elements are referred to as *letters*. Also, for s in S, it is customary to identify s with the length 1 word (s). Therefore, a length ℓ word whose successive letters are $s_1, ..., s_\ell$ is denoted as $s_1...s_\ell$ instead of $(s_1, ..., s_\ell)$.

Proposition 1.4.— *For each S, the structure $(S^*, \cdot, \varepsilon)$ is a monoid.*

Exercise 1.5.— Prove it.

As in the case of groups, we have the natural notion of a *submonoid* of a monoid.

Definition 1.6.— A subset M' of a monoid M is said to be a *submonoid* of M if M' equipped with the restriction of the product of M and its neutral element is itself a monoid.

Exercise 1.7.— Prove that M' is a sumonoid of M if and only if M' contains 1 and is closed under product, *i.e.*, the product of any two elements of M' belongs to M'.

Then we have the notion of submonoid *generated by* a set S. If M is a monoid and $S \subseteq M$ holds, the submonoid of M generated by S is the smallest submonoid of M that includes S. It exists because any intersection of submonoids is a submonoid (prove it) and, therefore, the intersection of all submonoids of M that include S is the smallest submonoid that includes S.

Next, we have the notion of *generating set* in a monoid: S is said to *generate* M if the submonoid of M generated by S is M itself.

Exercise 1.8.— (i) Prove that the submonoid of M generated by a set X is the union of $\{1\}$ and the set of all elements of M that can be expressed as products of (finitely many) elements of X.

(*ii*) Deduce that X generates M if and only if every element of M except possibly 1 can be expressed as a finite product of elements of X.

(*iii*) Show that a subset X of S^* generates S^* if and only if X includes S. [Hint: Letters have length 1, so the only way to decompose a letter into a product of words is to write $s = \varepsilon...\varepsilon s\varepsilon...\varepsilon$.]

1.2. Evalution mappings. The following result says that every monoid generated by a set S is a homomorphic image of the monoid S^*. This shows that S^* is a sort of universal object among all monoids generated by S. One says that the monoid S^* is a *free monoid with base* S—see Appendix.

Proposition 1.9.— *Assume that $(M, *, 1)$ is a monoid generated by S. Then there exists a unique homomorphism of S^* onto M that is the identity on S, namely the* M-evaluation mapping *defined by*

$$\mathrm{eval}_M(\varepsilon) = 1 \quad \textit{and} \quad \mathrm{eval}_M(s_1...s_\ell) = s_1 * ... * s_\ell. \tag{1.10}$$

Proof. The only potential problem is that (1.10) could be ambiguous if decomposition of words as products of letters were not unique. But they are: the only decomposition of a length ℓ word w into a product of letters is $w(1)...w(\ell)$—we recall that a length ℓ sequence is considered as a mapping with domain $\{1, ..., \ell\}$, so $w(i)$ means the ith letter of w. Then eval_M is a homomorphism by construction. The hypothesis that S generates M implies that eval_M is surjective. Finally, any homomorphism of S^* to M must satisfy (1.10), hence it coincides with eval_M. □

Definition 1.11.— In the context of Proposition 1.9, ones says that a word w *represents* an element a of M if we have $a = \mathrm{eval}_M(w)$.

When M and S are clear, one often writes $a = \overline{w}$ for $a = \mathrm{eval}_M(w)$.

1.3. Monoid congruences. The result that every monoid generated by S is a homomorphic image of S^* can be restated in terms of quotient-monoid.

Lemma 1.12.— *Assume that $\equiv$ is an equivalence relation on a monoid M. Then the operation of M induces a well defined operation on the quotient-set $M/\equiv$ if and only if the product and the relation are compatible, i.e., if the conjunction of $a' \equiv a$ and $b' \equiv b$ implies $a'b' \equiv ab$.*

Exercise 1.13.— Prove it.

Definition 1.14.— An equivalence relation on a monoid that is compatible with the product is called a *congruence*.

Lemma 1.15.— *Let M, M' be two monoids. Then the following are equivalent:*
(*i*) *There exists a surjective homomorphism of M onto M';*
(*ii*) *There exists a congruence $\sim$ on M such that M' is isomorphic to $M/\sim$.*

Proof. Assume that f is a surjective homomorphism of M to M'. Define a binary relation $\sim$ on M by $a \sim a'$ if $f(a) = f(a')$. Then $\sim$ is a congruence on M, and f induces a well defined mapping of $M/\sim$ to M' by $\overline{a} \mapsto f(a)$.

Conversely, if $\sim$ is a congruence on M, the map $a \mapsto \overline{a}$ is a surjective homomorphism of M onto $M/\sim$. Composing the latter by an isomorphism gives a surjective homomorphism of M onto every monoid that is isomorphic to $M/\sim$. □

Proposition 1.9 and Lemma 1.15 imply that every monoid M generated by a set S is a quotient of the free monoid S^*. This just corresponds to the fact that every element of M is a product of elements of S, *i.e.*, it is the evaluation in M of some word of S^* using (1.10). Now two different words of S^* may represent the same element of M, *i.e.*, a given element of M may have several expressions in terms of the elements of S. Applying Lemma 1.15, we deduce from Proposition 1.9:

Corollary 1.16.— *Assume that M is a monoid generated by S. Then M is isomorphic to $S^*/\sim_M$, where $\sim_M$ is the congruence* $\mathrm{eval}_M(w) = \mathrm{eval}_M(w')$ *on S^*.*

1.4. Monoid presentation. One way of specifying a monoid M (up to isomorphism) consists of giving a generating set S and the corresponding congruence $\sim_M$ on S^*. In good cases, S may be finite. However, even in that case, $\sim_M$ is an infinite object, as there are infinitely many words on S. So we look for more economical (= possibly finite) ways of specifying a congruence.

Definition 1.17.— Assume that M is a monoid, and R is a subset of M^2. A finite sequence $(a_0, ..., a_n)$ of elements of M is called an *R-derivation* from a_0 to a_n if, for each i, there exist a pair (b, b') in R and elements x, y of M satisfying $\{a_{i-1}, a_i\} = \{xby, xb'y\}$.

Lemma 1.18.— *Assume that M is a monoid, and $R \subseteq M^2$. Then there exists a smallest congruence $\equiv_R$ that includes R, namely the relation "there is exists an R-derivation from a to a'".*

Proof. First $\equiv_R$ is a congruence on M that includes R. Indeed, $\equiv_R$ is reflexive: the sequence (a) is an R-derivation from a to a. Next, $\equiv_R$ is symmetric: if $(a_0, ..., a_n)$ is an R-derivation from a to a', then $(a_n, ..., a_0)$ is an R-derivation from a' to a. Then, $\equiv_R$ is transitive: if $(a_0, ..., a_n)$ is an R-derivation from a to a' and $(a'_0, ..., a'_{n'})$ is an R-derivation from a' to a'', then $(a_0, ..., a_n, a'_1, ..., a'_{n'})$ is an R-derivation from a to a''. So $\equiv_R$ is an equivalence relation. Moreover, assume $a \equiv_R a'$. Let $(a_0, ..., a_n)$ be an R-derivation from a to a'. Then $(xa_0y, ..., xa_ny)$ is an R-derivation from xay to $xa'y$. Hence $\equiv_R$ is a congruence. Finally, assume $(b, b') \in R$. Then (b, b') is an R-derivation from b to b'.

Conversely, assume that $\sim$ be any congruence on M that includes R. First both $(y, y') \in R$ and $(y', y) \in R$ imply $y \sim y'$ since $\sim$ is symmetric, and $xyz \sim xy'z$ since $\sim$ is a congruence. Now, assume that $(a_0, ..., a_n)$ is an R-derivation from a to a'. By the previous remark, we have $a_{i-1} \sim a_i$ for each i, hence $a = a_0 \sim a_n = a'$ since $\sim$ is transitive. Therefore $a \equiv_R a'$ implies $a \sim a'$, *i.e.*, $\sim$ includes $\equiv_R$ as a set of pairs. □

Definition 1.19.— In the context of Lemma 1.18, we say that $\equiv_R$ is the congruence *generated* by R.

Exercise 1.20.— Let $\equiv$ be the relation "to have the same length" on S^*. Show that $\equiv$ is a congruence on S^* and that $\equiv$ is generated by $S \times S$, *i.e.*, by the set of all pairs (s, t) with s, t in S.

As every monoid is a quotient of a free monoid under a convenient congruence, we obtain a compact way of specifying a monoid by using a generating set of elements for the monoid and a generating set of pairs for the associated congruence.

Definition 1.21.— Let M be a monoid, S be a subset of M, and R be a subset of $S^* \times S^*$. We say that (S, R) is a *presentation* of M with set of generators S and set of relations R, also denoted

$$M = \langle S \mid R \rangle^+, \tag{1.22}$$

if S generates M and R generates the congruence $\sim_M$, *i.e.*, we have $\sim_M = \equiv_R$.

(Using equality in (2.5) is an abuse as $\langle S \mid R \rangle^+$ only defines M up to isomorphism.)

Example 1.23.— The congruence of S^* generated by the empty set is the smallest congruence on S^*, which is equality. So the monoid $\langle S \mid \emptyset \rangle^+$ is $S^*/=$, *i.e.*, it is S^* itself.

If (S, R) is a presentation of M, then $\sim_M$ includes R. Hence, for each pair (u, v) in R, we have $\mathrm{eval}_M(u) = \mathrm{eval}_M(v)$ in M: putting the pair (u, v) in the set of relations amounts to force the words u and v to represent the same element of the monoid. This explains the following notation.

Convention 1.24.— When displaying the relations of a presentation, one usually writes $u = v$ instead of (u, v). Also, one writes $\langle s_1, ..., s_n \mid r_1, ..., r_N \rangle^+$ instead of $\langle \{s_1, ..., s_n\} \mid \{r_1, ..., r_N\} \rangle^+$, and (often) 1 instead of ε.

Exercise 1.25.— Show that the congruence on $\{\mathtt{a}, \mathtt{b}\}^*$ generated by the relation $\mathtt{ab} = \mathtt{ba}$ is the congruence $|w|_\mathtt{a} = |w'|_\mathtt{a}$ & $|w|_\mathtt{b} = |w'|_\mathtt{b}$, where $|w|_s$ denotes the number of s in w. Deduce that the monoid $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab} = \mathtt{ba} \rangle^+$ is (isomorphic to) the monoid $(\mathbb{N}, +)^2$. [Hint: Show that $w \mapsto (|w|_\mathtt{a}, |w|_\mathtt{b})$ induces the expected isomorphism.]

Exercise 1.26.— (*i*) Let p be a fixed positive integer. Show that the congruence on $\{\mathtt{a}\}^*$ generated by the relation $\mathtt{a}^p = 1$ is the congruence $|w| = |w'| \pmod p$. Deduce that the monoid $\langle \mathtt{a} \mid \mathtt{a}^p = 1 \rangle^+$ is (isomorphic to) the group $(\mathbb{Z}/p\mathbb{Z}, +)$.

(*ii*) Describe $\langle \mathtt{a} \mid \mathtt{a}^{m+p} = \mathtt{a}^m \rangle^+$.

The following statement is easy, but it is very useful.

Proposition 1.27.— *Assume that M is a monoid generated by S, and R is a set of relations on S (i.e., a subset of S^2). Then the following are equivalent:*

(*i*) *The relations of R hold in M, i.e.,* $\mathrm{eval}_M(u) = \mathrm{eval}_M(v)$ *holds for each relation $u = v$ of R.*
(*ii*) *The monoid M is a quotient of $\langle S \mid R\rangle^+$.*

PROOF. Both (*i*) and (*ii*) are equivalent to saying that $\sim_M$ includes $\equiv_R$, *i.e.*, that any two $\equiv_R$-equivalent words have the same evaluation in M. (Check details.) □

1.5. The Word Problem. Once again, saying that S generates a monoid M implies that every element of M can be expressed as a product of elements of S, *i.e.*, is the M-evaluation of some word of S^*.

Definition 1.28.— Assume that M is a monoid generated by S. The *Word Problem* of M relative S is the algorithmic question: Given two words w, w' in S^*, decide whether w and w' are M-equivalent, *i.e.*, whether $\mathrm{eval}_M(w) = \mathrm{eval}_M(w')$ holds.

For the monoids of Exercises 1.25 and 1.26, we have an explicit description of $\sim_M$ and we easily deduce a solution to the Word Problem. In the case of Exercise 1.25, the algorithm could be:

Count the letters a in w and w';
  If the numbers are different, say NO;
  Otherwise count the letters b in w and w';
    If the numbers are different, say NO;
    Otherwise say YES.

(Being more precise would require defining a universal language for specifying algorithms.)

When we start with a finite monoid presentation, *i.e.*, a presentation with a finite set of generators and a finite set of relations, we might hope to obtain a universal way for solving the word problem. This is *not* the case.

Theorem 1.29 (Markov, Post, 1944**).**— *There exists a finite monoid presentation whose Word Problem is unsolvable.*

The reason why there is no simple solution is the following one. Owing to Lemma 1.18, two words of S^* have the same $\langle S \mid R\rangle^+$-evaluation if and only if they are $\equiv_R$-equivalent, *i.e.*, there exists an R-derivation from w to w'.

Lemma 1.30.— *Assume that (S, R) is a finite monoid presentation. Then there exists an algorithm that, for each finite sequence $(w_0, ..., w_n)$ in S^*, recognizes whether it is an R-derivation.*

PROOF. As R is finite, deciding whether a length 2 sequence (w_1, w_2) is an R-derivation is doable by systematically enumerating all decompositions of w_1 and applying all possible relations of R. Then $(w_0, ..., w_n)$ is an R-derivation if and only if (w_{i-1}, w_i) is an R-derivation for each i. □

Given a finite monoid presentation (S, R) and two words w, w' of S^*, we can always enumerate all R-derivationes that start from w, *i.e.*, apply to w all possible relations of R in all possible ways, thus obtaining a list of words $w_0 = w$, $w_1, ...$ If w' is R-equivalent to w, then it will appear in the list and, therefore, we shall know that w and w' are $\equiv_R$-equivalent after a finite number of steps. On the other hand, if w' is not $\equiv_R$-equivalent to w, we shall in general never know it for sure: at each step, we can only see that w' has not yet occurred in the list, but this does not prove in general that it cannot appear later. The theorem of Markov and Post says that there is no uniform way to overcome the problem.

Of course, this does not say that, for one given presentation, the Word Problem is necessarily unsolvable. For instance, Exercise 1.25 shows that the Word Problem of $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab} = \mathtt{ba}\rangle^+$ is solvable. But the solutions to the Word Problem must be specific.

Exercise 1.31.— Show that, if for each word w of S^* only finitely many words may be $\equiv_R$-equivalent to w, then the Word Problem of (S, R) is solvable. Apply this to the case when the relations of R preserve the length, *i.e.*, are of the form $u = v$ with $|u| = |v|$.

1.6. Normal forms. A usual method for solving a Word Problem consists in using normal forms, *i.e.*, in identifying one distinguished element (= "normal element") in each equivalence class.

Proposition 1.32.— *Assume that M is generated by S and N is a subset of S^* that contains exactly one element in each $\sim_M$-equivalence class. Then M is isomorphic to $(N, *, \mathrm{NF}(\varepsilon))$, where $*$ is defined by*

$u * v = \text{NF}(uv)$ and $\text{NF}(w)$ denotes the unique element of N that is M-equivalent to w. Moreover, if the map NF is computable, the Word Problem of M relative S is solvable.

Exercise 1.33.— Prove it.

The problem is that, in many cases, it is possible to guess a subset N of S^* that contains *at least one* element in each $\sim_M$-equivalence class, typically by playing with the relations when M is given by an explicit presentation, but, then, it is difficult to prove that the elements of N are pairwise M-inequivalent (as a general rule, it is more difficult to prove a non-equivalence than an equivalence). The usual solution is to look for an $\equiv_R$*-invariant*, namely a map I defined on S^* that takes the same value for all $\equiv_R$-equivalent words, so that $I(w) \neq I(w')$ implies that w, w' are not $\equiv_R$-equivalent.

Example 1.34.— As in Exercise 1.25, let $S = \{\mathtt{a}, \mathtt{b}\}$ and let $R = \{\mathtt{ab} = \mathtt{ba}\}$. An easy induction shows that every word of S^* is $\equiv_R$-equivalent to a special word, defined as those words of the form $\mathtt{a}^p\mathtt{b}^q$ with $p, q \geqslant 0$ (prove it). It follows that each $\equiv_R$-equivalence class contains at least one special word. In order to prove that each $\equiv_R$-equivalence class contains one special word exactly, we have to prove that distinct special words are not $\equiv_R$-equivalent. Let $I_{\mathtt{a}}(w)$ be the number of letters $\mathtt{a}$ in w. We observe that $I_{\mathtt{a}}(\mathtt{ab}) = I_{\mathtt{a}}(\mathtt{ba})$ is true, and that the relation $I_{\mathtt{a}}(w) = I_{\mathtt{a}}(w')$ is a congruence. Hence this relation includes the congruence $\equiv_R$ generated by R, *i.e.*, $w \equiv_R w'$ implies $I_{\mathtt{a}}(w) = I_{\mathtt{a}}(w')$. Similarly the number of letters $\mathtt{b}$ give a similar invariant $I_{\mathtt{b}}$. Now we observe that, if two special words $\mathtt{a}^p\mathtt{b}^q$ and $\mathtt{a}^{p'}\mathtt{b}^{q'}$ are distinct, at least one of the two invariants $I_{\mathtt{a}}, I_{\mathtt{b}}$ separates them.

Exercise 1.35.— Treat the case of the monoid $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab}^2 = \mathtt{ba} \rangle^+$ similarly. (This is more tricky.)

2. Group presentations

Every group is a monoid, and we can apply the previous notion of presentation to groups, up to some adaptation.

2.1. From monoids to groups. A group is a particular monoid in which every element admits an inverse, *i.e.*, for each g there exists a (necessarily unique) element g^{-1} satisfying $gg^{-1} = g^{-1}g = 1$.

A subset G' of a group G is a subgroup if and only if the operations of G, namely the product and the inverse operation, induce the structure of a group on G'. It follows that G' is a subgroup if and only if it contains 1 and it is closed under the product and the inverse operation. In particular, a subgroup is always a submonoid, but the converse is not true: for instance, $\mathbb{N}$ is a submonoid of the group $\mathbb{Z}$, but it is not a subgroup.

It follows that the submonoid of a group G generated by a subset S is in general smaller in the subgroup generated by S: the former is the closure of $S \cup \{1\}$ under product, whereas the latter is the closure both on closure and inverse. For instance, the submonoid of the group $\mathbb{Z}$ generated by 1 is $\mathbb{N}$, whereas the subgroup generated by 1 is $\mathbb{Z}$.

It follows in turn that the notion of generating subset differ according to whether we look at a group as at a monoid.

Lemma 2.1.— *Assume that G is a group and S is a subset of G. Then the subgroup generated by S is the submonoid generated by $S \cup S^{-1}$. In particular, if G is generated as a group by S, then it is generated by $S \cup S^{-1}$ as a monoid.*

Proof. The submonoid of G generated by $S \cup S^{-1}$ contains 1 and is closed under product and inverse, hence it is a subgroup of G. Conversely, every subgroup of G that includes S also includes S^{-1}, hence it includes the submonoid generated by $S \cup S^{-1}$. □

A congruence on a group G is an equivalence relation on G that is compatible with the operations of the structure, namely the product and the inverse operation—whereas a congruence on a monoid is supposed to be compatible with the product only. However, there is no problem here.

Lemma 2.2.— *Assume that G is a group and $\equiv$ is a monoid congruence on G. Then $\equiv$ is also a group congruence, i.e., it is automatically compatible with the inverse operation.*

Proof. Assume $g \equiv g'$. Then we deduce

$$g^{-1} = g^{-1}g'g'^{-1} \equiv g^{-1}gg'^{-1} = g'^{-1},$$

hence $\equiv$ is compatible with the inverse operation and it is a group congruence. □

2.2. Group presentation.

Proposition 2.3.— *Assume that G is a monoid generated by S, and R is a set of relations on $S \cup S^{-1}$ (i.e., a subset of $(S \cup S^{-1})^2$). Then the following are equivalent:*

(i) the relations of R hold in G, i.e., $\mathrm{eval}_G(u) = \mathrm{eval}_G(v)$ holds for each relation $u = v$ of R.

(ii) the group G is a quotient of $\langle S \cup S^{-1} \mid R \cup \{ss^{-1} = s^{-1}s = 1 \mid s \in S\}\rangle^+$.

Proof. The hypothesis that G is generated by S as a group implies that it is generated by $S \cup S^{-1}$ as a monoid.

Assume (i). As G is a group, the relations $ss^{-1} = 1$ and $s^{-1}s = 1$ hold in G for each s in S. By Proposition 1.27, we deduce that, as a monoid, G is a quotient of the monoid $\langle S \cup S^{-1} \mid R \cup \{ss^{-1} = s^{-1}s = 1 \mid s \in S\}\rangle^+$. By Lemma 2.2, there is no need to distinguish between quotient-monoid and quotient-group.

Conversely, assume (ii). By Proposition 1.27, the relations of R hold in G. □

Definition 2.4.— Let G be a group, S be a subset of M, and R be a subset of $((S \cup S^{-1})^*)^2$. We say that (S, R) is a *presentation* of G with set of generators S and set of relations R, also denoted

$$G = \langle S \mid R\rangle, \tag{2.5}$$

if S generates G and R plus all relations $ss^{-1} = s^{-1}s = 1$ for s in S generate the congruence $\sim_G$, *i.e.*, if G admits, as a monoid, the presentation $\langle S \cup S^{-1} \mid R \cup \{ss^{-1} = s^{-1}s = 1 \mid s \in S\}\rangle^+$.

Then we can restate Proposition 2.3 as follows.

Proposition 2.6.— *Assume that G is a group generated by S, and R is a set of relations on $S \cup S^{-1}$. Then the following are equivalent:*

(i) the relations of R hold in G, i.e., $\mathrm{eval}_G(u) = \mathrm{eval}_G(v)$ holds for each relation $u = v$ of R.

(ii) the group G is a quotient of $\langle S \mid R\rangle$.

Remark 2.7.— In the case of a group, the relations can always be assumed to be of the form $u = 1$.

2.3. Free groups.

Definition 2.8.— Let S be a nonempty set. A group admitting the presentation $\langle S \mid \emptyset\rangle$ is called *free* of with base S.

Proposition 2.6 immediately implies:

Proposition 2.9.— *Every group generated by a set S is isomorphic to a quotient of (any) free group based on S.*

Exercise 2.10.— Show that $\mathbb{Z}$ is a free group on one generator.

Definition 2.8 characterizes a free group only up to isomorphism only. We shall now describe a more concrete free group of base S. By construction, the monoid

$$\langle S \cup S^{-1} \mid \{ss^{-1} = s^{-1}s = 1 \mid s \in S\}\rangle^+,$$

i.e., the quotient $S^*/\equiv$ where $\equiv$ is the smallest congruence on $(S \cup S^{-1})^*$ that contains all relations $ss^{-1} = s^{-1}s = 1$ with s in S is a free group based on S. As in Section 1.6, we shall extract a subset of S^* that contains exactly one element in each $\equiv$-class.

Definition 2.11.— A word over $S \cup S^{-1}$ is called *reduced* if it contains no subword $s^{-1}s$ or ss^{-1}. The set of all reduced words over $S \cup S^{-1}$ is denoted F_S.

The notation $\equiv$ always referring to the congruence generated by the relations $ss^{-1} = s^{-1}s = 1$ with s in S, we shall prove

Proposition 2.12.— *Every word in $(S \cup S^{-1})^*$ is $\equiv$-equivalent to a unique reduced word.*

Applying Proposition 1.32, we deduce

Corollary 2.13.— *For w in $(S \cup S^{-1})^*$, let $\mathrm{red}(w)$ denote the unique reduced word that is $\equiv$-equivalent to w. Then $(F_S, *)$ is a free group based on S, where $*$ is defined by $u * v = \mathrm{red}(uv)$.*

To prove Proposition 2.12, one introduces a *rewrite rule* on $(S \cup S^{-1})^*$, *i.e.*, a binary relation.

Definition 2.14.— For w, w' in $(S \cup S^{-1})^*$, we say that w *reduces to* w' *in one step*, denoted $w \to w'$, if there exist words u, v and a letter s in S such that w is $uss^{-1}v$ or $us^{-1}sv$ and w' is uv. We we say that w *reduces to* w', denoted $w \to^* w'$, if there exists a finite sequence $(w_0, ..., w_n)$ satisfying $w_0 = w$, $w_n = w'$ and $w_{i-1} \to w_i$ for each i.

Thus $w \to^* w'$ holds if one can go from w to w' by deleting subwords of the form ss^{-1} or $s^{-1}s$. The words that are terminal for $\to^*$ are the reduced words. As reduction diminishes the length by 2, there is no infinite sequence of reductions, and, therefore, every word reduces to a reduced word in finitely many steps. However, it is not a priori obvious that this reduced word is unique.

Lemma 2.15.— *Every word reduces to a unique reduced word.*

Proof. Write $w \to^p$ if w reduces to w' in exactly p steps. First, we claim that, if $w \to^p w_1$ and $w \to^q w_2$ hold, then there exists w' and $p' \leqslant p$, $q' \leqslant q$ satisfying $w_1 \to^{p'} w'$ and $w_2 \to^{q'} w'$. We use induction on $p + q$. The cases $p = 0$ and $q = 0$ are trivial. The case $p = q = 1$ is treated directly. Then the induction is easy (do it).

Now, assume $w \to^* w_1$ and $w \to^* w_2$ where w_1 and w_2 are reduced. By Lemma 2.15, there exists w' satisfying $w_1 \to^* w'$ and $w_2 \to^* w'$. As w_1 and w_2 are reduced, the only possibility is $w' = w_1 = w_2$. □

Hereafter we denote by $\mathrm{red}(w)$ the unique reduced word w' satisfying $w \to^* w'$.

Proof of Proposition 2.12. By Lemma 2.15, and because $w \to w'$ implies $w \equiv w'$, every $\equiv$-equivalence class contains at least one reduced word.

Now consider the relation $\mathrm{red}(w) = \mathrm{red}(w')$ on $(S \cup S^{-1})^*$. As red is a well defined mapping, this relation is an equivalence relation. Moreover, it is compatible with the product. Indeed, assume $\mathrm{red}(w) = \mathrm{red}(w')$ and let u, v be arbitrary words. First $w \to^* \mathrm{red}(w)$ implies $uwv \to^* u\mathrm{red}(w)v$. We deduce $uwv \to^* \mathrm{red}(u\mathrm{red}(w)v)$, hence $\mathrm{red}(uwv) = \mathrm{red}(u\mathrm{red}(w)v)$. Similarly, $w' \to^* \mathrm{red}(w)$ implies $uw'v \to^* u\mathrm{red}(w)v$, and we deduce

$$\mathrm{red}(uw'v) = \mathrm{red}(u\mathrm{red}(w)v) = \mathrm{red}(uw'v).$$

So the relation $\mathrm{red}(w) = \mathrm{red}(w')$ is a congruence. Moreover we have $\mathrm{red}(ss^{-1}) = \mathrm{red}(s^{-1}s) = \varepsilon = \mathrm{red}(\varepsilon)$ for each s in S. By definition of $\equiv$, this implies that $w \equiv w'$ implies $\mathrm{red}(w) = \mathrm{red}(w')$. It follows that an $\equiv$-class contains at most one reduced word. □

Example 2.16.— Assume $S = \{\mathtt{a}, \mathtt{b}\}$. Then we consider words over the alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{a}^{-1}, \mathtt{b}^{-1}\}$. In examples, it is convenient to use $\mathtt{A}$ for $\mathtt{a}^{-1}$ and $\mathtt{B}$ for $\mathtt{b}^{-1}$. Reduced words are those words that contain no factor $\mathtt{aA}$, $\mathtt{Aa}$, $\mathtt{bB}$, or $\mathtt{Bb}$. The free group based on $\{\mathtt{a}, \mathtt{b}\}$ is the family of all such reduced words, equipped with the product $*$. For instance, we find

$$(\mathtt{abA}) * (\mathtt{aB^2a}) = \mathrm{red}(\mathtt{ab\underline{Aa}B^2a}) = \mathrm{red}(\mathtt{a\underline{bB}Ba}) = \mathtt{aBa}.$$

Exercise 2.17.— Show that the group $\langle S \mid R\rangle$ is the quotient of the free group F_S by the normal subgroup generated by the elements $\mathrm{red}(u^{-1}v)$ for $u = v$ a relation of R.

2.4. The Word Problem. The Word Problem for a group is similar to the Word Problem for a monoid. The only difference is that, as inverses always exist, it is enough to decide whether one word represents 1: indeed, if G is a group generated by S, then two words w, w' of $(S \cup S^{-1})^*$ represent the same element of G if and only if the word $w^{-1}w'$ represents 1—where w^{-1} denotes the word obtained from w by reversing the order of letters and exchanging s and s^{-1} everywhere.

As groups are particular cases of monoids, the Word Problem of groups is a more restricted problem than the Word Problem of monoids, and therefore it might be easier. This is not the case.

Theorem 2.18 (P. Novikov, 1952**).**— *There exists a finite group presentation whose Word Problem is unsolvable.*

So, once again, we cannot hope for a uniform method for solving the Word Problem, but only for solutions in particular cases.

Exercise 2.19.— Solve the Word Problem for the group $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab} = \mathtt{ba} \rangle$. What is this group?

Example 2.20.— Let G be the group $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{a}^2 = \mathtt{b}^2 = 1, \mathtt{aba} = \mathtt{bab} \rangle$. Let us try to recognize this group. Let $\equiv_R$ denote the congruence associated with the relations of the presentation (including the relations $\mathtt{aA} = 1$, etc.). First, $\mathtt{a}^2 \equiv_R 1$ implies $\mathtt{A} \equiv_R \mathtt{a}$ and, similiarly, $\mathtt{b}^2 \equiv_R 1$ implies $\mathtt{B} \equiv_R \mathtt{b}$. Hence every word on $\{\mathtt{a}, \mathtt{b}, \mathtt{A}, \mathtt{B}\}$ is $\equiv_R$-equivalent to a word on $\{\mathtt{a}, \mathtt{b}\}$. Next we claim that every such word is $\equiv_R$-equivalent to at least one of the six words ε, $\mathtt{a}$, $\mathtt{b}$, $\mathtt{ab}$, $\mathtt{ba}$, $\mathtt{aba}$. The reason is that, when we append $\mathtt{a}$ or $\mathtt{b}$ to the right of any one of these six words, the resulting word is again $\equiv_R$-equivalent to one of the six words. Hence, $\equiv_R$ has at most six classes, and G has *at most* 6 elements.

To prove that G has exactly six elements (and to recognize G) demands to prove that the six words ε, ..., $\mathtt{aba}$ are not pairwise $\equiv$-equivalent. This can be done using an $\equiv_R$-invariant. Define $I : \{\mathtt{a}, \mathtt{b}\} \to \mathfrak{S}_3$ by $I(\mathtt{a}) = (1,2)$ and $I(\mathtt{b}) = (2,3)$. The transpositions $(1,2)$ and $(2,3)$ satisfy the relations of R: we have $(1,2)^2 = (2,3)^2 = \mathrm{id}$ and $(1,2)(2,3)(1,2) = (2,3)(1,2)(2,3)$, hence I induces a well defined homomorphism of G to $\mathfrak{S}_3$. One checks that the images under I of the six words ε, ..., $\mathtt{aba}$ are pairwise distinct permutations of $\{1,2,3\}$, hence they are paiwise $\equiv_R$-unequivalent. Hence G has six elements. As I is injective, it is also surjective, and I is an isomorphism of G onto $\mathfrak{S}_3$. In other words, $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{a}^2 = \mathtt{b}^2 = 1, \mathtt{aba} = \mathtt{bab} \rangle$ is a presentation of the symmetric group $\mathfrak{S}_3$.

3. Presentation of braid groups

We shall now find a proesentation for the braid groups B_n of Chapter **??**.

3.1. Generators. We recall that the elements of the group B_n are isotopy classes of n-strand braid diagrams, *i.e.*, of projections of n-strand geometric braids. There are many such diagrams, and, in order to find a generating family that is not too large, we will show that we can restrict to particular diagrams, *i.e.*, that every diagram is isotopic to a diagram of this particular type.

Definition 3.1.— A geometric braid is called *piecewise linear* if it consists of line segments. A geometric braid is called *regular* if it is piecewise linear and its projection is a braid diagram in which any two crossings have different z-coordinate.

Lemma 3.2.— *Every geometric braid is isotopic to a regular geometric braid.*

Proof. Let Γ be a geometric braid. By definition, each strand of Γ is compact, so there exists a positive constant δ so that any two strands of Γ remain at distance $\geqslant \delta$, and their projection contains finitely many crossings only. Replacing the portions of curves that separate the crossings of the projection by line segments correspond to an isotopy, see Figure 1. Next, if two crossings of the projection have the same z-coordinate, we push one to the left and one to the right to obtain a regular braid. □

Figure 1. Normalization of a braid diagram and expression as a product of elementary diagrams σ_i et σ_i^{-1}.

It is then easy to identify a family of generators for B_n.

Lemma 3.3.— *Let Γ_i be the n-strand geometric braid consisting of $n+1$ segments $[(k,0,0),(k,0,1)]$ for $k \neq i, i+1$, plus $[(i,0,0),(i+1,0,1)]$, $[(i+1,0,0),(i+1/2,1/2,1/2)]$ and $[(i+1/2,1/2,1/2),(i,0,1)]$, and let σ_i be the isotopy class of Γ_i. Then the group B_n is generated by $\sigma_1, ..., \sigma_{n-1}$.*

Proof. Assume that Γ is a regular geometric braid. By an isotopy, we can transform Γ into a regular geometric braid Γ' that consists of translated copies of the braids Γ_i and their mirror images[1]— see Figure 1. Then, the braid diagram that is the projection of Γ' can be cut into slices so that each slice contains exactly one crossing. By definition, such an elementary diagram is either a σ_i, or the mirror

[1]When doing that, a diagram with ℓ crossings is drawn in $\{1, ..., n\} \times \{0, ..., \ell\}$, and not in $\{1, ..., n\} \times [0,1]$ as it should be: rescaling is easy.

image of a σ_i, *i.e.*, σ_i^{-1}. This means that every braid that is not 1 can be expressed as a product of braids σ_i and σ_i^{-1}, *i.e.*, that $\{\sigma_1, ..., \sigma_{n-1}\}$ generates the group B_n. □

Remark 3.4.— The final normalization step in the proof of Lemma 3.3 is superfluous: whenever Γ is a regular geometric braid Γ, we can encode Γ by a sequence of letters σ_i and σ_i^{-1}, *i.e.*, by a word over the alphabet of σ_i's and σ_i^{-1}'s—called a *braid word*—that indicates which strands are involved in the successive crossings of the projection of Γ. Such a braid word will be called the *code* of the geometric braid Γ.

Remark 3.5.— The braids σ_i are called *the Artin generators* of the braid group B_n. It might be reasonable to write $\sigma_{i,n}$ rather than σ_i in order to indicate the considered total number of strands. We shall see below that B_n embeds as a subgroup of B_{n+p} so that there is no danger in not distinguishing $\sigma_{i,n}$ and $\sigma_{i,n+p}$.

3.2. Relations. For $n \geqslant 3$, the braid group B_n is not a free group: some relations other than the trivial group relations $\sigma_i\sigma_i^{-1} = \sigma_i^{-1}\sigma_i$ connect the braids σ_i. It is easy to find such relations by looking at braid diagrams.

Lemma 3.6.— *The following relations are satisfied in the group B_n:*

$$\sigma_i\sigma_j = \sigma_j\sigma_i \quad \text{for } |i-j| \geqslant 2, \tag{3.7}$$

$$\sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j \quad \text{for } |i-j| = 1. \tag{3.8}$$

Proof. See Figure 2. □

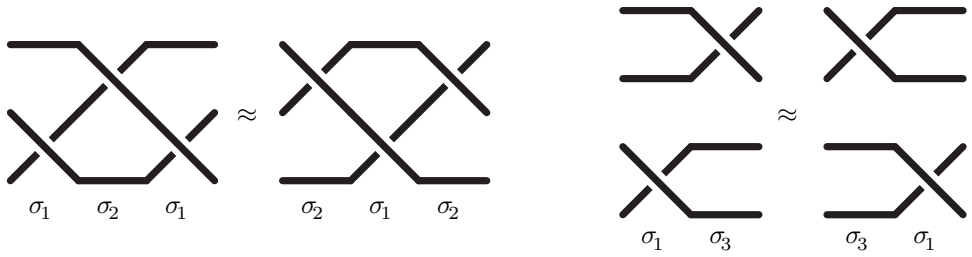


Figure 2. Two types of relations connecting the Artin generators σ_i.

Saying that B_n satisfies the relations of Lemma 3.6 does *not* prove that these relations make a presentation of B_n: there might still exist other relations that hold in B_n and that are not consequences of the relations of Lemma 3.6. Actually, this is not the case.

Theorem 3.9 (Artin, 1925**).**— *The group B_n admits the presentation*

$$\left\langle \sigma_1, \dots, \sigma_{n-1} \;\middle|\; \begin{array}{ll} \sigma_i\sigma_j = \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle. \tag{3.10}$$

The problem is to prove that, if two diagrams are isotopic, then one can transform one into the other using the relations of (**??**), or, more exactly that, if two geometric braids Γ, Γ' are isotopic, then one can transform Γ into Γ' using elementary steps that, when coded into braid words, correspond to applying the relations of (**??**) plus the relations $\sigma_i\sigma_i^{-1} = \sigma_i^{-1}\sigma_i = 1$. To do that, we shall consider the notion of a Δ-move between piecewise linear geometric braids.

Definition 3.11.— (See Figure 3.) Assume that Γ, Γ' are regular geometric braids. We say that Γ' is obtained by a Δ-*move* from Γ if Γ' is obtained by replacing one segment $[AB]$ of Γ with two adjacent segments $[AC]$, $[CB]$ such that no strand of Γ intersect the triangle $[ABC]$.

Lemma 3.12.— *Two regular geometric braids Γ, Γ' are isotopic if and only if Γ can be transformed into Γ' using a finite sequence of Δ-moves.*

Proof. An isotopy can be decomposed into steps that only involve the neighbourhood of one segment at a time, and, from there, can themselves be decomposed into a sequence of Δ-moves. Compactness guarantees that only a finite number of Δ-moves are needed. □

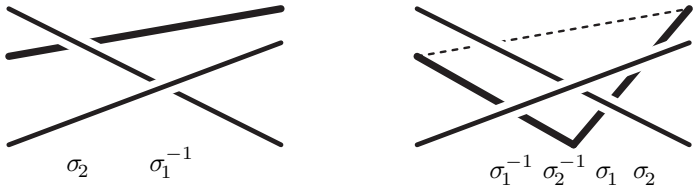


FIGURE 3. Δ-move: when the bold segment on the left is replaced with the two bold segments on the right, the crossings of the projection change, hence the coding by a braid word changes as well, but one goes from the old coding to the new coding by applying a relation that is a consequence of the relations of Lemma 3.6. In the current example, the relation is $\sigma_2\sigma_1^{-1} = \sigma_1^{-1}\sigma_2^{-1}\sigma_1\sigma_2$, which is a consequence of $\sigma_1\sigma_2\sigma_1 = \sigma_2\sigma_1\sigma_2$ and of the implicit relations $\sigma_i\sigma_i^{-1} = \sigma_i^{-1}\sigma_i = 1$.

PROOF OF THEOREM 3.9. Assume that Γ, Γ' are isotopic geometric braids. By Lemma 3.2, we can assume without loss of generality that Γ and Γ' are regular. Then, by Lemma 3.12, Γ and Γ' are connected by a sequence of Δ-moves, and it is enough to consider one Δ-move, *i.e.*, to compare the braid words that code the two geometric braids occurring in a Δ-move. The point is that only finitely many cases may occur, according to the relative positions of the various segments that possibly cross in the interval between the old and the new crossings. Figure 3 gives a typical example, in which we see that the old code, here $\sigma_2\sigma_1^{-1}$, and the new code, here $\sigma_1^{-1}\sigma_2^{-1}\sigma_1\sigma_2$, are connected by a relation that follows from (**??**):

$$\sigma_2\sigma_1^{-1} = \sigma_1^{-1}\sigma_1\sigma_2\sigma_1^{-1} = \sigma_1^{-1}\sigma_2^{-1}\sigma_2\sigma_1\sigma_2\sigma_1^{-1} = \sigma_1^{-1}\sigma_2^{-1}\sigma_1\sigma_2\sigma_1\sigma_1^{-1} = \sigma_1^{-1}\sigma_2^{-1}\sigma_1\sigma_2.$$

The other cases are similar. □

3.3. Back to the Braid Isotopy Problem. Once a presentation of the groups B_n is known, one can restate the Braid Isotopy Problem. Theorem 3.9 implies

Corollary 3.13.— *The Braid Isotopy Problem in the case of n-strand braid diagrams is equivalent to the Word Problem for the presentation* (**??**) *of the group* B_n*:*

Given two braid diagrams D, D'*, that can be supposed to be regular:*

- 1. Code the crossings of D *and* D' *by two braid words* w, w'*;*

- 2. Then D *and* D' *are isotopic if and only if* $w \equiv w'$ *holds, where* $\equiv$ *is the congruence on braid words generated by the relations of* (**??**).

Example 3.14.— Consider the diagrams D, D' of Figure **??**.8, the ones that could not be distinguished using easy invariants. The coding of the crossings of D is $\sigma_1^2\sigma_2^2$, that of D' is $\sigma_2^2\sigma_1^2$. Proving D and D' are not isotopic (this is what will be done eventually) reduces to proving $\sigma_1^2\sigma_2^2 \not\equiv \sigma_2^2\sigma_1^2$, or, equivalently, $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2 \not\equiv \varepsilon$. In concrete example, it is convenient to use $\mathtt{a} = \sigma_1$, $\mathtt{b} = \sigma_2$, ... and $\mathtt{A} = \sigma_1$, $\mathtt{B} = \sigma_2$, ... In this way, the question is to prove $\mathtt{BBAAbbaa} \not\equiv \varepsilon$.

Of course, we are not yet done. The Novikov theorem says that there is no uniform method for solving a Word Problem. Therefore the best we can hope for is to find a specific method that works for the presentations (**??**).

CHAPTER III

Braid monoids

At this point, we reduces the Braid Isotopy Problem for n-strand braids to an algebraic problem, namely the Word Problem of the braid group B_n with respect to the presentation

$$(*) \qquad \left\langle \sigma_1, \dots, \sigma_{n-1} \;\middle|\; \begin{array}{ccc} \sigma_i\sigma_j = \sigma_j\sigma_i & \text{for} & |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j & \text{for} & |i-j| = 1 \end{array} \right\rangle.$$

By Novikov's Theorem, we know that there is no hope to find a uniform method for solving this kind of Word Problem. Hence we have to find a specific method taking the particular relations of $(*)$ into account.

1. Using monoids

The method we shall describe here was discovered by F.A. Garside at the end of the 1960's [**11**]. It relies on using monoids.

1.1. The monoids B_n^+ and $\overline{B_n^+}$. The relations of $(*)$ involve no letter σ_i^{-1}. Hence it makes sense to introduce the monoid for which $(*)$ is a presentation.

Definition 1.1 (monoid B_n^+).— We denote by B_n^+ the monoid that admits the presentation

$$(1.2) \qquad \left\langle \sigma_1, \dots, \sigma_{n-1} \;\middle|\; \begin{array}{ccc} \sigma_i\sigma_j = \sigma_j\sigma_i & \text{for} & |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j & \text{for} & |i-j| = 1 \end{array} \right\rangle^+.$$

On the other hand, whenever G is a group and S is a subset of G, there exists a smallest submonoid of G that is generated by S, namely the family of all elements of G that can be expressed as products of elements of S, plus 1.

Definition 1.3 (monoid $\overline{B_n^+}$).— We denote by $\overline{B_n^+}$ the submonoid of the group B_n generated by the elements $\sigma_1, ..., \sigma_{n-1}$.

By construction, all relations of (1.2) are satisfied in $\overline{B_n^+}$. Hence Proposition II.1.27 implies

Proposition 1.4.— *For each n, the monoid $\overline{B_n^+}$ is a quotient of the monoid B_n^+.*

By construction, the elements of the monoids B_n^+ and $\overline{B_n^+}$ are represented by words over the alphabet $\{\sigma_1, ..., \sigma_{n-1}\}$, whereas the elements of the group B_n are represented by words over the larger alphabet $\{\sigma_1, ..., \sigma_{n-1}, \sigma_1^{-1}, ..., \sigma_{n-1}^{-1}\}$.

Definition 1.5 (positive braid word, braid word equivalence).— A word over the alphabet $\{\sigma_1, ..., \sigma_{n-1}, \sigma_1^{-1}, ..., \sigma_{n-1}^{-1}\}$ is called a *braid word*; a word over the alphabet $\{\sigma_1, ..., \sigma_{n-1}\}$ is called a *positive braid word*. Two braid words w, w' are called *equivalent*, denoted $w \equiv w'$, if they represent the same element of B_n. Two positive braid words w, w', are called *positively equivalent*, denoted $w \equiv^+ w'$, if they represent the same element of B_n^+.

According to the description of Chapter II, two braid words w, w' are equivalent if and only if there exists an R-derivation from w to w', where R consists of the relations of $(*)$ plus the free group relations $\sigma_i\sigma_i^{-1} = \sigma_i\sigma_i = 1$, whereas two positive braid words w, w' are positively equivalent if and only if there exists an R^+-derivation from w to w', where R^+ consists of the relations of $(*)$, *i.e.*, of (1.2), exclusively. Hence it is clear that, for w, w' positive,

$$(1.6) \qquad w \equiv^+ w' \quad \text{implies} \quad w \equiv w'.$$

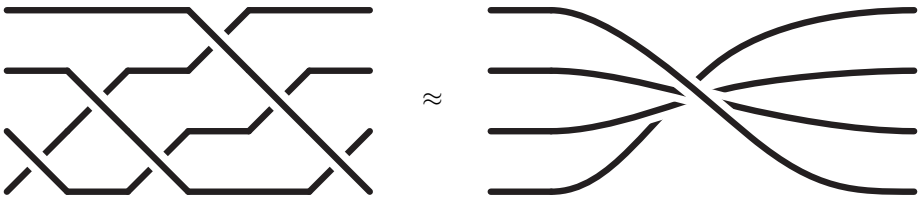


FIGURE 1. Two diagrams representing the fundazmental braid Δ_4.

By construction we have

$$B_n^+ = \{\sigma_1, ..., \sigma_{n-1}\}^*/\equiv^+ \quad \text{and} \quad \overline{B_n^+} = \{\sigma_1, ..., \sigma_{n-1}\}^*/\equiv^{1},$$

so (1.6) is just another statement of Proposition 1.4.

Question 1.7.— *Is the implication of* (1.6) *an equivalence? In other words: Is B_n^+ isomorphic to $\overline{B_n^+}$?*

We shall eventually see that the answer is positive but, at the moment, this is not clear at all.

1.2. Reducing to $\overline{B_n^+}$. For the moment we leave Question 1.7 open and return to the Word Problem of B_n. Here we shall reduce this problem to the Word Problem of the monoid $\overline{B_n^+}$ with respect to $\{\sigma_1, ..., \sigma_{n-1}\}$. To this end, we introduce a specific positive braid that plays an important role.

Definition 1.8 (fundamental braid Δ_n).— For $n \geqslant 1$, the *fundamental braid* Δ_n is defined by

$$\Delta_1 = 1 \quad \text{and} \quad \Delta_n = \Delta_{n-1}\,\sigma_{n-1}...\sigma_2\sigma_1 \quad \text{for } n \geqslant 2. \tag{1.9}$$

The braid Δ_n corresponds to a complete half-turn of the n strands, see Figure 1.

In the sequel, we also denote by Δ_n the positive braid word recursively defined by (1.9). So the *word* Δ_n represents the *braid* Δ_n—as the letter σ_i represents the braid σ_i. We hope the context will make it clear whether we speak of braids or of braid words.

We begin with three technical results involving positive braid equivalence.

Lemma 1.10.— *For $i \leqslant j < k$, we have*

$$\sigma_j\ (\sigma_k\sigma_{k-1}...\sigma_{i+1}\sigma_i) \equiv^+ (\sigma_k\sigma_{k-1}...\sigma_{i+1}\sigma_i)\ \sigma_{j+1}, \tag{1.11}$$

$$\sigma_{j+1}\ (\sigma_i\sigma_{i+1}...\sigma_{k-1}\sigma_k) \equiv^+ (\sigma_i\sigma_{i+1}...\sigma_{k-1}\sigma_k)\ \sigma_j. \tag{1.12}$$

PROOF. We prove (1.11) using induction on $|k-i|$. For $k = i+1$, we necessarily have $j = i$, and (1.11) reduces to $\sigma_i(\sigma_{i+1}\sigma_i) \equiv^+ (\sigma_{i+1}\sigma_i)\sigma_{i+1}$, which is true. Assume $k \geqslant i+2$. Assume first $j = i$. Then we have $\sigma_j\sigma_k \equiv^+ \sigma_k\sigma_j$ and $|(k-1)-i| < |k-i|$, so, using the induction hypothesis, we find

$$\sigma_j\ (\sigma_k\sigma_{k-1}...\sigma_i) = \sigma_j\,\sigma_k\ (\sigma_{k-1}...\sigma_i) \equiv^+ \sigma_k\,\sigma_j\ (\sigma_{k-1}...\sigma_i) \equiv^+ \sigma_k\ (\sigma_{k-1}...\sigma_i)\ \sigma_{j+1},$$

which is (1.11). Assume now $j \geqslant i+1$. Then we have $\sigma_{j+1}\sigma_i \equiv^+ \sigma_i\sigma_{j+1}$ and $|k-(i+1)| < |k-i|$, so, using the induction hypothesis, we find

$$\sigma_j\ (\sigma_k...\sigma_{i+1}\sigma_i) = \sigma_j\ (\sigma_k...\sigma_{i+1})\ \sigma_i \equiv^+ (\sigma_k...\sigma_{i+1})\ \sigma_{j+1}\ \sigma_i \equiv^+ (\sigma_k...\sigma_{i+1})\ \sigma_i\ \sigma_{j+1},$$

again the expected result. The proof of (1.12) is entirely symmetric (do it). □

Lemma 1.13.— *For each n, we have*

$$\Delta_n \equiv^+ \sigma_1\sigma_2...\sigma_{n-1}\Delta_{n-1}. \tag{1.14}$$

PROOF. Use induction on $n \geqslant 1$. The result is clear for $n = 1$ and $n = 2$. Assume $n \geqslant 3$. Using the induction hypothesis, and using the fact that Δ_{n-2} contains only letters σ_i with $i \leqslant n-3$, hence letters that satisfy $\sigma_i\sigma_{n-1} \equiv^+ \sigma_{n-1}\sigma_i$, we find

$$\begin{aligned}\Delta_n = \Delta_{n-1}\sigma_{n-1}...\sigma_1 &\equiv^+ (\sigma_1\ ...\ \sigma_{n-2}\Delta_{n-2})\ (\sigma_{n-1}\ ...\ \sigma_1) = (\sigma_1\ ...\ \sigma_{n-2})\ \Delta_{n-2}\ \sigma_{n-1}\ (\sigma_{n-2}\ ...\ \sigma_1)\\ &\equiv^+ (\sigma_1\ ...\ \sigma_{n-2})\ \sigma_{n-1}\ \Delta_{n-2}\ (\sigma_{n-2}\ ...\ \sigma_1) = \sigma_1\ ...\ \sigma_{n-1}\Delta_{n-1},\end{aligned}$$

as expected. □

[1] more exactly $\{\sigma_1, ..., \sigma_{n-1}\}^*$ quotiented by the restriction of $\equiv$ to $\{\sigma_1, ..., \sigma_{n-1}\}^*$

Lemma 1.15.— *For $1 \leqslant i \leqslant n-1$, we have*

$$\sigma_i\ \Delta_n \equiv^+ \Delta_n\ \sigma_{n-i}. \tag{1.16}$$

PROOF. Use induction on $n \geqslant 1$. The result is vacuously true for $n = 1$. For $n = 2$, it reduces to $\sigma_1^2 \equiv^+ \sigma_1^2$. Assume $n \geqslant 3$. Assume first $i \leqslant n-2$. Applying the induction hypothesis and (1.11) for 1, $n-i-1$ and $n-1$, we find

$$\sigma_i\Delta_n = \sigma_i\ \Delta_{n-1}\ (\sigma_{n-1}...\sigma_1) \equiv^+ \Delta_{n-1}\ \sigma_{n-i-1}\ (\sigma_{n-1}...\sigma_1) \equiv^+ \Delta_{n-1}\ (\sigma_{n-1}...\sigma_1)\ \sigma_{n-i}.$$

Assume now $i = n-1$. Using (1.14), (1.12), and the induction hypothesis, and (1.14) again, we find

$$\sigma_{n-1}\Delta_n \equiv^+ \sigma_{n-1}\ (\sigma_1...\sigma_{n-1})\ \Delta_{n-1} \equiv^+ (\sigma_1...\sigma_{n-1})\ \sigma_{n-2}\ \Delta_{n-1} \equiv^+ (\sigma_1...\sigma_{n-1})\ \Delta_{n-1}\ \sigma_1 \equiv^+ \Delta_n\sigma_1. \quad \square$$

Lemma 1.17.— *For $1 \leqslant i \leqslant n-1$, there exists a positive word $w_{i,n}$ satisfying $\Delta_n \equiv^+ \sigma_i\ w_{i,n}$.*

PROOF. Use induction on $n \geqslant 1$. The result is vacuously true for $n = 1$, and trivial for $n = 2$. Assume $n \geqslant 3$. For $i \leqslant n-2$, the induction hypothesis gives a word $w_{i,n-1}$ satisfying $\Delta_{n-1} \equiv^+ \sigma_i w_{i,n-1}$, and, applying (1.9), we find

$$\Delta_n = \Delta_{n-1}\ (\sigma_{n-1}...\sigma_1) \equiv^+ \sigma_i\ w_{i,n-1}\ (\sigma_{n-1}...\sigma_1),$$

i.e., the result is true with $w_{i,n} = w_{i,n-1}\sigma_{n-1}...\sigma_1$.

There only remains the case of σ_{n-1}. By construction, the word Δ_{n-1} only contains letters σ_i with $1 \leqslant i \leqslant n-2$. Each of these letters σ_i is eligible for Lemma 1.10 with respect to $\sigma_{n-1}...\sigma_1$, *i.e.*, satisfies $\sigma_i\ (\sigma_{n-1}...\sigma_1) \equiv^+ (\sigma_{n-1}...\sigma_1)\ \sigma_{i+1}$. It follows that one has $\Delta_n \equiv^+ \sigma_{n-1}...\sigma_1 w$, where w is the braid word obtained from Δ_{n-1} by shifting all indices by $+1$, *i.e.*, replacing σ_i with σ_{i+1} everywhere. We obtained the expected result with $w_{n-1,n} = \sigma_{n-2}...\sigma_1 w'$. $\square$

Remark 1.18.— The above positive equivalences (involving $\equiv^+$) imply similar equivalences (involving $\equiv$). The latter correspond to diagrams that could easily be seen to be isotopic. However, as long as we did not prove a positive answer to Question 1.7, such diagrams would prove nothing about the relation $\equiv^+$. In the current section, results involving $\equiv$ would be sufficient, but the stronger versions involving $\equiv^+$ will be needed in Section 2, and that is why we established them here.

With the previous results at hand, we can easily deduce the following results about the group B_n and its submonoid $\overline{B_n^+}$.

Proposition 1.19.— (*i*) *For each n, the braid Δ_n^2 belongs to the center of B_n.*
(*ii*) *For $1 \leqslant i \leqslant n-1$, the braid $\sigma_i^{-1}\Delta_n$ belongs to $\overline{B_n^+}$.*

PROOF. (*i*) Applying Lemma 1.15 twice shows that $\sigma_i\Delta_n^2 \equiv^+ \Delta_n^2\sigma_i$ holds for $1 \leqslant i \leqslant n-1$. This implies $\sigma_i\Delta_n^2 \equiv \Delta_n^2\sigma_i$, so, in the group B_n, the element Δ_n^2 commutes with every element of the generating family $\{\sigma_1, ..., \sigma_{n-1}\}$. Hence Δ_n^2 commutes with every element of B_n, *i.e.*, it belongs to its center.

(*ii*) Applying Lemma 1.17, we obtain that the words Δ_n and $\sigma_i w_{i,n}$ are positively equivalent, hence equivalent. It follows that, in the group B_n, the element $\sigma_i^{-1}\Delta_n$ is represented by the positive word $w_{i,n}$. This means that $\sigma_i^{-1}\Delta_n$ belongs to the submonoid $\overline{B_n^+}$, which, by definition, consists of those braids that can be represented by at least one positive braid word. $\square$

Corollary 1.20.— *Each braid β in B_n can be expressed as $\Delta_n^{-2d}\beta'$ for some $d \geqslant 0$ and $\beta' \in \overline{B_n^+}$.*

PROOF. Assume that w is a braid word containing d letters σ_i^{-1}. Let w' be the word obtained from w by replacing each letter σ_i^{-1} with $w_{i,n}\Delta_n$, where $w_{i,n}$ is as in Lemma 1.17. Then, by construction, w' is a positive braid word, and $\Delta_n^{2d}w'$ is equivalent to w. Indeed, starting from $\Delta_n^{2d}w'$, we can freely move the factors Δ_n^2 up to braid equivalence, since $\sigma_i^{-1}\Delta_n^2 \equiv \Delta_n^2\sigma_i^{-1}$ holds for each i, and $\sigma_i^{-1}\Delta_n^2$ is equivalent to $w_{i,n}\Delta_n$ by Lemma 1.17. Let β' be the positive braid represented by w'. Then, by construction, we have $\beta' = \Delta_n^{2d}\beta$, hence $\beta = \Delta_n^{-2d}\beta'$ in B_n. $\square$

Actually, we have more, namely an algorithm for transforming an arbitrary n-strand braid word into an equivalent word which is positive up to an initial possible negative power of the word Δ_n.

Algorithm 1.21.— *Input: An n-strand braid word w.*
Output: A nonnegative integer d and positive braid word w_1 satisfying $w \equiv \Delta_n^{-2d}\ w_1$.
Method: - Let d be the number of negative letters σ_i^{-1} in w;

- Let w_1 be the positive braid word obtained by replacing each letter σ_i^{-1} of w with the word $w_{i,n}\Delta_n$.

Corollary 1.22.— *The Word Problem of the group B_n (with respect to $\sigma_1, ..., \sigma_{n-1}$) is equivalent to the Word Problem of the monoid $\overline{B_n^+}$ (with respect to $\sigma_1, ..., \sigma_{n-1}$).*

PROOF. Having to decide whether $w \equiv ?\varepsilon$ is true, we apply Algorithm 1.21 to find a positive word w_1 satisfying $w \equiv \Delta_n^{-2d} w_1$. Then $w \equiv \varepsilon$ is equivalent to $\Delta_n^{2d} \equiv w_1$, which is an instance of the Word Problem for the monoid $\overline{B_n^+}$. So any algorithm solving the latter will provide an algorithm for the former. □

Example 1.23.— Let w be the braid word $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$, which corresponds to the quotient of the braids that resisted to all naive invariants of Chapter I. As above we use the simplified expression `BBAAbbaa` corresponding to $\mathtt{a} = \sigma_1$, $\mathtt{b} = \sigma_2$, ... $\mathtt{A} = \sigma_1^{-1}$, $\mathtt{B} = \sigma_2^{-1}$, ... In order to decide $w \equiv ?\varepsilon$, we run Algorithm 1.21. First, the proof of Lemma 1.17 gives the values $w_{1,3} = \mathtt{ba}$, $w_{2,3} = \mathtt{ab}$. So, starting from $w = \mathtt{BBAAbbaa}$, we find

$$w_1 = (\mathtt{ababa})\ (\mathtt{ababa})\ (\mathtt{baaba})\ (\mathtt{baaba})\ \mathtt{bbaa},$$

and $w \equiv ?\varepsilon$ is equivalent to $\Delta_3^8 \equiv ?w_1$, *i.e.*, $(\mathtt{aba})^8 \equiv ?$ `ababaababababaabababbaa`, an instance of the Word Problem of the monoid B_3^+.

Exercise 1.24.— Compute the words $w_{i,4}$. [Solution: $w_{1,4} = \mathtt{bacba}$, $w_{2,4} = \mathtt{abcba}$, $w_{3,4} = \mathtt{babcb}$.]

Exercise 1.25.— (*i*) Fix n. Prove that the map $\Phi_n : \sigma_i \mapsto \sigma_{n-i}$ extends into a well defined automorphism of the group B_n, as well as into a well defined automorphism of the monoids $\overline{B_n^+}$ and B_n^+. Show that Φ_n is the inner automorphism of B_n associated with Δ_n. [Use Lemma 1.15.]

(*ii*) Use Lemma 1.15 and the automorphism Φ_n to refine the method used to prove Corollary 1.20 and show that, if an n-strand braid β can be represented by a word containing d letters σ_i^{-1}, then $\Delta_n^d\beta$ belongs to $\overline{B_n^+}$.

1.3. The Word Problem of B_n^+.

We begin with a trivial observation.

Proposition 1.26.— *The Word Problem of the monoid B_n^+ is decidable, i.e., there is an algorithm that solves it.*

PROOF. All relations appearing in the presentation (1.2) are of the form $u = v$ with u and v of equal length (2 or 3 in the current case). An induction on the length of a derivation shows that any two positively equivalent words have the same length. On the other hand, $\{\sigma_1, ..., \sigma_{n-1}\}$ is a finite set, so, for each ℓ, there exists only finitely many positive n-strand braid words of length ℓ. □

We deduce the following stupid algorithm for solving the word problem of B_n^+ (cf. Exercise II.II.1.31).

Algorithm 1.27.— *Input: Two positive braid words u, v.*
Ouput: YES or NO;
Method: - Enumerate all *positive words that are $\equiv^+$-equivalent to u by systematically applying all braid relations at all possible positions, until no more word appears*
- Return YES if v appears in the list so obtained, and NO otherwise.

At this point, we reduced the Word Problem of B_n to that of $\overline{B_n^+}$ and, on the other hand, we know how to solve the Word Problem of B_n^+. So there remains an obvious question, namely Question 1.7: Are $\overline{B_n^+}$ and B_n^+ isomorphic? Equivalently: Is (1.2) a presentation of $\overline{B_n^+}$? At the moment, we know of no presentation of $\overline{B_n^+}$, and there might very well exist in B_n relations that involve positive braids and are not consequences of the braid relations of (1.2).

1.4. Ores's Theorem.

We need a criterion for connecting the monoids B_n^+ and $\overline{B_n^+}$. It will be provided by the following result.

Proposition 1.28 (Ore's Theorem, 1931**).**— *Assume that M is a cancellative monoid and any two elements of M admit a common right-multiple. Then there exists a group G, unique up to isomorphism, with the following properties:*

(*i*) *there is an injective homomorphism I of M into G;*

(*ii*) *every element of G can be expressed as $I(a)I(b)^{-1}$ for some a, b in M.*

Moreover, if $\langle S \mid R\rangle^+$ is a monoid presentation of M, then $\langle S \mid R\rangle$ is a group presentation of G.

First, some words must be defined.

Definition 1.29 (cancellative).— A monoid M is said to be *left-cancellative* (*resp.right-cancellative*) if $xa = xb$ (*resp.*$ax = bx$) implies $a = b$ in M. It is said to be *cancellative* if it is both left- and rihght-cancellative.

Definition 1.30 (divisor, multiple).— Assume that M is a monoid. We say that an element a of M is a *left-divisor* b, denoted $a \preccurlyeq b$, or, equivalently, that b is a *righth-multiple* of a, if there exists c satisfying $ac = b$.

The proof of Ore's Theorem is a little long, but it is not difficult. It is a natural extension of the construction of $(\mathbb{Z}, +)$ from $(\mathbb{N}, +)$, or of $(\mathbb{Q}, *)$ from $(\mathbb{Z}, *)$, namely constructing a group by considering pairs (= fractions) of elements of the monoid.

Proof of Ore's Theorem. Define a binary relation $\sim$ on $M \times M$ by

$$(a, b) \sim (a', b') \quad \Leftrightarrow \quad \exists x, x' \ (ax = a'x' \ \& \ bx = b'x'). \tag{1.31}$$

Claim 1. *The relation $\sim$ is reflexive and symmetric.*

Indeed, say that (x, x') witnesses for $(a, b) \sim (a', b')$ if we have $ax = a'x'$ and $bx = b'x'$. Then $(1, 1)$ witnesses for $(a, b) \sim (a, b)$, and, if (x, x') witnesses for $(a, b) \sim (a', b')$, then (x', x) witnesses for $(a', b') \sim (a, b)$.

Claim 2. *The relation $(a, b) \sim (a', b')$ holds if and only we have $(*)$ $\forall y, y' \ (ay = a'y' \ \Leftrightarrow \ by = b'y')$.*

Indeed, assume that (x, x') is a witness for $(a, b) \sim (a', b')$ and we have $ay = a'y'$. By hypothesis, the elements x and y admit a common right-multiple in M, *i.e.*, there exist c, d satisfying $xd = yc$. Then we find $a'x'd = axd = ayc = a'y'c$, hence $x'd = y'c$ since a' is left-cancellable in M. We deduce $byc = bxd = b'x'd = b'y'c$, hence $by = b'y'$ since c is right-cancellable in M. So $(a, b) \sim (a', b')$ implies $(*)$.

Conversely, $(*)$ implies $(a, b) \sim (a', b')$. Indeed, assume $(*)$. By hypothesis, a and a' admit a common right-multiple, so there exist x, x' satisfying $ax = a'x'$. Applying our hypothesis, we deduce $bx = b'x'$, *i.e.*, (x, x') is a witness for $(a, b) \sim (a', b')$.

Claim 3. *The relation $\sim$ is an equivalence relation.*

Indeed, after Claim 1, it remains to verify that $\sim$ is transitive. Assume $(a, b) \sim (a', b') \sim (a'', b'')$. By hypothesis, the elements a, a', a'' admit some common right-multiple, say $ax = a'x' = a''x''$. By Claim 2, the hypothesis $(a, b) \sim (a', b')$ plus the equality $ax = a'x'$ imply $bx = b'x'$. Similarly, the hypothesis $(a', b') \sim (a'', b'')$ plus the equality $a'x' = a''x''$ imply $b'x' = b''x''$. Therefore, we have $ax = a''x''$ and $bx = b''x''$, hence $(a, b) \sim (a'', b'')$, and $\sim$ is an equivalence relation.

For a, b in M, we shall denote by a/b the $\sim$-class of (a, b). We denote by G the quotient-set $M^2/\sim$.

Claim 4. *The map $I : M \to G$ defined by $I(a) = a/1$ is injective.*

Indeed, assume $I(a) = I(a')$. This means that we have $(a, 1) \sim (a', 1)$, *i.e.*, there exist x, x' satisfying $ax = a'x'$ and $1x = 1x'$. This forces $x = x'$ and then $a = a'$ since x is right-cancellable in M.

Claim 5. *For a, b, c, d, x, y satisfying $bx = cy$, the $\sim$-class of (ax, dy) does not depend on x and y.*

Indeed, assume $bx = cy$ and $bx' = cy'$. The elements y and y' admit a common right-multiple in M, so there exist t, t' satisfying $yt = y't'$. Then we have $bxt = cyt = cy't' = bx't'$, hence $xt = x't'$ since b is left-cancellable in M. So, we have $dyt = dy't'$ and $axt = ax't'$, hence $(ax, by) \sim (ax', by')$.

Therefore, we can unambiguously define a well defined (external) binary operation $*$ of M^2 to G by

$$(a, b) * (c, d) = (ax)/(dy) \quad \text{whenever } bx = cy. \tag{1.32}$$

Claim 6. *The operation $*$ induces an (internal) binary operation on G.*

The question is to prove that $(a, b) * (c, d)$ only depends on the $\sim$-classes of (a, b) and (c, d). Assume $(a, b) \sim (a', b')$, and $(c, d) \sim (c', d')$. By hypothesis, the elements b, b', c, c' admit a common right-multiple, *i.e.*, there exist x, x', y, y' satisfying $bx = b'x' = cy = c'y'$. By definition, we have $(a, b) * (c, d) = (ax)/(dy)$ and $(a', b') * (c', d') = (a'x')/(d'y')$. Now, by Claim 2, $bx = b'x'$ implies $ax = a'x'$, and $cy = c'y'$ implies $dy = d'y'$, whence $(ax)/(dy) = (a'x')/(d'y')$.

Hereafter, we use $\cdot$ (or nothing) for the binary operation induced by $*$ on G.

Claim 7. *The product on G is associative.*

Let a, b, c, d, e, f belong to M. Choose x, y, z, t, u, v satisfying $bx = cy$, $dz = et$, and $yu = zv$—see Figure 2. By definition, we have

$$a/b \cdot (c/d \cdot e/f) = a/b \cdot (cz)/(ft) = (axu)/(ftv), \tag{1.33}$$

$$(a/b \cdot (c/d) \cdot e/f = (ax)/(dy) \cdot e/f = (axu)/(ftv). \tag{1.34}$$

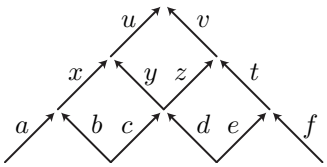


FIGURE 2. Associativity of the product of fractions in the proof of Ore's Theorem; we associate an s-labeled arrow with each element s, and represent equalities of products (for instance $bx = cy$) by a (commutative) square; in this way any two paths from one vertex to another one give the same value.

Indeed, the second equality in (1.33) is true because we have $b \cdot xu = cyu = cz \cdot v$, and that in (1.34) because we have $dy \cdot u = dzv = e \cdot tv$.

Claim 8. *The set G equipped with $\cdot$ is a group with neutral element $1/1$; the inverse of a/b is b/a.*

Let a, b be arbitrary elements of M. We have $b \cdot 1 = 1 \cdot b$, hence $a/b * 1/1 = (a \cdot 1)/(1 \cdot b) = a/b$ by defintion. Similarly, we have $1 \cdot a = a \cdot 1$, hence $1/1 * a/b = (1 \cdot a)/(b \cdot 1) = a/b$, so $1/1$ is neutral on the left. A similar computation shows it is neutral on the right. Moreover, we have $(x, x) \sim (1, 1)$ for each x, hence $x/x = 1/1$. Finally, we have $a/b * b/a = (ab)/(ab) = 1/1$, and $b/a * a/b = (ba)/(ba) = 1/1$.

Claim 9. *The map I is an embedding (= injective homomorphism) of M into G.*

Let a, b be arbitrary elements of M. We have $1 \cdot b = b \cdot 1$, hence

$$I(a) \cdot I(b) = a/1 \cdot b/1 = (a \cdot b)/(1 \cdot 1) = (ab)/(1) = I(ab).$$

Claim 10. *Every element of G has an expression as $I(a)I(b)^{-1}$ with a, b in M.*

Indeed, by construction, we have $a/b = a/1 \cdot 1/b = a/1 \cdot (b/1)^{-1} = I(a)I(b)^{-1}$

Claim 11. *If a subset S of M generates M as a monoid, then $I(S)$ generates G as a group.*

Indeed, the hypothesis guarantees that every element of G can be expressed as a finite product of elements of $I(S)$ and of inverses of elements of $I(S)$.

Assume now that $\langle S \mid R\rangle^+$ is a monoid presentation of M. By Claim 10, we know that $I(S)$ generates G. We want to prove that G admits the presentation $\langle I(S) \mid I(R)\rangle$, where $I(R)$ consists of I-copies of the relations of R, *i.e.*, of one relation $I(u) = I(v)$ for each relation $u = v$ of R. The question is to show that, for each word w over the alphabet $S \cup S^{-1}$, we have $\mathrm{eval}_G(I(w)) = 1 \Leftrightarrow I(w) \equiv_{I(R)} \varepsilon$, where $\equiv_{I(R)}$ is the congruence generated by $I(R)$ and the free group relations $I(s)I(s)^{-1} = I(s^{-1})I(s) = 1$. The letters $I(s)$ are just copies of the letters of S, so the question is also the question of proving

$$\mathrm{eval}_G(I(w)) = 1 \quad \Leftrightarrow \quad w \equiv_R \varepsilon \tag{1.35}$$

for each word w in $(S \cup S^{-1})^*$, where $\equiv_R$ is the congruence generated by R and the free group relations $ss^{-1} = s^{-1}s = 1$.

Claim 12. *The relation $w \equiv_R \varepsilon$ implies* $\mathrm{eval}_G(I(w)) = 1$.

Indeed, if $u = v$ is a relation of R, we have $\mathrm{eval}_M(u) = \mathrm{eval}_M(v)$ by hypothesis, hence $\mathrm{eval}_G(I(u)) = I(\mathrm{eval}_M(u)) = I(\mathrm{eval}_M(v) = \mathrm{eval}_G(I(v))$. On the other hand, $\mathrm{eval}_G(I(s)I(s)^{-1}) = \mathrm{eval}_G(I(s)^{-1}I(s)) = 1 = \mathrm{eval}_G(\varepsilon)$ holds for each s in S. As the previous pairs generate $\equiv_R$, the result follows.

Claim 13. *Every word w in $(S \cup S^{-1})^*$ is $\equiv_R$-equivalent to some word uv^{-1} with u, v in S^*.*

Indeed, w can always be expressed as $u_1 v_1^{-1} u_2 v_2^{-1} ... u_d v_d^{-1}$, where $u_1, ..., v_d$ belong to S^*. We use induction on d. For $d = 1$, there is nothing to prove. Assume $d \geqslant 2$. By induction hypothesis, there exist u', v' in S^* satisfying $u_1 v_1^{-1} u_2 v_2^{-1} ... u_{d-1} v_{d-1}^{-1} \equiv_R u'v'^{-1}$. In the monoid M, the elements represented by v' and u_d have a common right-multiple. This means that there exist two words u'', v'' in S^* satisfying $u_d v'' \equiv_R v'u''$ (and even $u_d v'' \equiv_R^+ v'u''$, where $\equiv_R^+$ is the positive equivalence generated by R). Using the free groupu relations we deduce $v'^{-1}u_d \equiv_R u''v''^{-1}$, and, from there, $w \equiv_R u'v'^{-1}u_d v_d^{-1} \equiv_R u'u''v''^{-1}v_d^{-1}$, as expected.

Claim 14. *The relation* $\mathrm{eval}_G(I(w)) = 1$ *implies $w \equiv_R \varepsilon$.*

Indeed, assume $\mathrm{eval}_G(I(w)) = 1$. By Claim 10, there exist words u, v in S^* satisfying $w \equiv_R uv^{-1}$. Then $\mathrm{eval}_G(I(w)) = 1$ implies $\mathrm{eval}_G I(u) = \mathrm{eval}_G I(v)$, which is also $I(\mathrm{eval}_M(u)) = I(\mathrm{eval}_M(v))$. As I is injective, this implies $\mathrm{eval}_M(u) = \mathrm{eval}_M(v)$, hence $u \equiv_R^+ v$ since $\langle S \mid R\rangle^+$ is a presentation of M. A fortiori, we have $u \equiv_R v$, hence $uv^{-1} \equiv_R \varepsilon$, and, finally, $w \equiv_R \varepsilon$ by transitivity.

The proof of Ore's Theorem is complete. □

Corollary 1.36.— *Under the hypotheses of Proposition 1.28 (Ore's Theorem), the monoid M is isomorphic to the submonoid of G generated by $I(S)$.*

PROOF. As I is injective, it is an isomorphism of M onto its image in G. By hypothesis, M is generated by S, so its image is generated by $I(S)$, hence it is the submonoid of G that is generated by $I(S)$. □

Exercise 1.37.— Prove that the sufficient conditions of Ore's Theorem are also sufficient: If there exists an injective homomorphism I of a monoid M into a group G such that every element of G has an expression of the form $I(a)I(b)^{-1}$ with a, b in M, then M is cancellative and any two elements of M admit a common right-multiple.

Exercise 1.38.— Let M be the monoid $\langle a, b, c, d, a', b', c', d' \mid ac = a'c', ad = a'd', bc = b'c' \rangle^+$. Prove that M is cancellative, but M embeds in no group. [Hint: Prove that $bd = b'd'$ fails in M, but holds in every group that satisfies the three defining relations of M.]

2. Garside theory

At this point, we proved the following results:

- by Corollary **??**, the Braid Isotopy Problem reduces to the Word Problem of B_n with respect to Artin's generators σ_i;

- by Corollary 1.22, the Word Problem of the group B_n with respect to Artin's generators σ_i reduces to the Word Problem of the monoid $\overline{B_n^+}$ with respect to Artin's generators σ_i;

- by Proposition 1.26, the Word Problem of the monoid B_n^+ with respect to Artin's generators σ_i is solvable;

- by Corollary 1.36 (Ore's Theorem), if the monoid B_n^+ is cancellative and admits common right-multiples, then it is isomorphic to $\overline{B_n^+}$.

So, *if* the monoid B_n^+ is cancellative and admits common right-multiples, we shall have solved the Braid Isotopy Problem (at last).

2.1. Common multiples in B_n^+. So, the strategy is clear: it remains to prove that the monoid B_n^+, *i.e.*, the monoid defined by the braid relations of $(*)$, is cancellative and admits common right-multiples. We begin with the latter point, which is easy.

Proposition 2.1.— *Any two elements of the monoid B_n^+ admit a common right-multiple.*

We begin with an auxiliary result.

Lemma 2.2.— *Assume that u is a positive n-strand braid word of length at most ℓ. Then there exists a positive braid word v satisfying $uv \equiv^+ \Delta_n^\ell$.*

PROOF. We use induction on $\ell \geqslant 0$. The result is obvious for $\ell = 0$. Assume $\ell \geqslant 1$. The result is obvious if u is empty. Otherwise, we have $u = u'\sigma_i$ for some u' of length at most $\ell - 1$. By induction hypothesis, there exists a word v' satisfying $u'v' \equiv^+ \Delta_n^{\ell-1}$. Let $\phi_n(v')$ denote the word obtained from v' by exchanging the letters σ_i and σ_{n-i} everywhere. By Lemma 1.15, we have $\Delta_n\, \phi_n(v') \equiv^+ v'\Delta_n$. On the other hand, by Lemma 1.15, there exists a positive word $w_{i,n}$ satisfying $\sigma_i\, w_{i,n} \equiv^+ \Delta_n$. Put $v = w_{i,n}\phi_n(v')$. Then we obtain

$$uv = u'\,\sigma_i\, w_{i,n}\phi_n(v') \equiv^+ u'\,\Delta_n\,\phi_n(v') \equiv^+ u'v'\Delta_n \equiv^+ \Delta_n^{\ell-1}\Delta_n = \Delta_n^\ell. \qquad \square$$

PROOF OF PROPOSITION 2.1. Let a, b be any two elements of B_n^+. By definition, there exist positive n-strand braid words u, v representing a and b. Let d be any number satisfying $d \geqslant |u|$ and $d \geqslant |v|$. By Lemma 2.2, the element of B_n^+ represented by the word Δ_n^d is a common right-multiple of a and b in B_n^+. □

2.2. Complemented presentations. So we are left with the question of proving that the monoid B_n^+ is cancellative. The result is true, but this is a *difficult* question, and no very simple proof is known. The first proof was due to F.A. Garside in his 1969 paper [**11**]. The proof we shall give now is a variant of Garside's proof that relies on a combinatorial method called *word reversing*. It uses the specific form of the braid relations.

Definition 2.3 (complement, complemented presentation).— Let S be an alphabet (nonempty set). A *complement* on S is a mapping $C : S \times S \to S^*$ satisfying $C(s, s) = \varepsilon$ for each s in S. We say that a monoid M is *associated with* C if M admits the presentation $\langle S \mid R_C \rangle^+$ where R_C is the family of all relations $sC(s, t) = tC(t, s)$ for $s \neq t$ in S.

Lemma 2.4.— *The monoid B_n^+ is associated with the complement C on $\{\sigma_1, ..., \sigma_{n-1}\}$ defined by*

$$C(\sigma_i, \sigma_j) = \begin{cases} \varepsilon & \text{for } i = j, \\ \sigma_j\sigma_i & \text{for } |i-j| = 1, \\ \sigma_j & \text{for } |i-j| \geqslant 2. \end{cases} \tag{2.5}$$

Proof. Check it. □

Complemented presentations are rather special, and we shall establish a specific criterion for recognizing whether a monoid with a complemented presentation is cancellative.

2.3. Subword reversing. We introduce a rewrite system, *i.e.*, a collection of rules that transform words into new words. We are interested in the special case of braids, but it will be more simple to state a general definition.

Definition 2.6 (alphabet duplication).— Assume that C is a complement on the alphabet S. We consider a duplicated alphabet $S^\pm$ obtained by adding to S a copy denoted s^{-1} for each letter s. The letters of S are called *positive*, those of the form s^{-1} are called *negative*. A word on $S^\pm$ is called an $S^\pm$-word. If w is a $S^\pm$-word, we denote by w^{-1} the word obtained from w by reversing the order of the letters and exchanging s and s^{-1} everywhere.

In examples like $S = \{\mathtt{a}, \mathtt{b}\}$, it is convenient to use $\mathtt{A}$ for $\mathtt{a}^{-1}$, $\mathtt{B}$ for $\mathtt{b}^{-1}$, etc. So, for $w = \mathtt{abaA}$, we find $w^{-1} = \mathtt{aABA}$. Remember that we work with words, and not with elements of a group. So, for instance, $\mathtt{aA}$ is not the same as the empty word. In any case, we are now considering monoids, and there are no inverses.

Definition 2.7 (subword reversing).— Assume that w, w' are $S^\pm$-words. We say that w is *reversible to w' in one step*, denoted $w \curvearrowright_C^1 w'$ or, simply, $w \curvearrowright^1 w'$, if w' either by deleting some length two subword $s^{-1}s$, or by replacing a length two subword $s^{-1}t$ with $s \neq t$ by the corresponding word $C(s,t)C(t,s)^{-1}$. We say that $(w_0, ..., w_N)$ is a *reversing sequence* if $w_{k-1} \curvearrowright w_k$ holds for each k, and that w is reversible to w', denoted $w \curvearrowright w'$, if there is a (finite) reversing sequence that starts with w and finishes with w'.

Reversing uses the complement C as a recipe to push the negative letters to the right and the positive letters to the left by iteratively reversing $-+$-subwords into $+-$-subword (whence the name). Note that deleting $s^{-1}s$ enters the general scheme as we assume that, for every letter s in S, we have $C(s,s) = \varepsilon$.

Example 2.8.— Consider the complement associated with the braid relations $(*)$. Put $w = \sigma_3^{-1}\sigma_1\sigma_2^{-1}\sigma_1$. Then w contains two $-+$-subwords, namely $\sigma_3^{-1}\sigma_1$ and $\sigma_2^{-1}\sigma_1$. So there are two ways of starting a right reversing from w: replacing $\sigma_3^{-1}\sigma_1$ with $\sigma_1\sigma_3^{-1}$, or replacing $\sigma_2^{-1}\sigma_1$ with $\sigma_1\sigma_2\sigma_1^{-1}\sigma_2^{-1}$. A typical reversing sequence is (we use $\mathtt{a}$ for σ_1, etc.

$$\mathtt{CaBa} \curvearrowright^1 \mathtt{aCBa} \curvearrowright^1 \mathtt{aCabAB} \curvearrowright^1 \mathtt{aaCbAB} \curvearrowright^1 \mathtt{aabcBCAB}. \tag{2.9}$$

We cannot continue, since the latter word contains no $-+$ subword.

Exercise 2.10.— Let C be the complement on $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ defined by $C(\mathtt{a}, \mathtt{b}) = \mathtt{ba}$, $C(\mathtt{b}, \mathtt{a}) = \mathtt{ab}$, $C(\mathtt{b}, \mathtt{c}) = \mathtt{cb}$, $C(\mathtt{c}, \mathtt{b}) = \mathtt{bc}$, $C(\mathtt{a}, \mathtt{c}) = \mathtt{ca}$, $C(\mathtt{c}, \mathtt{a}) = \mathtt{ac}$. Show that there exist arbitrarily long reversing sequences starting from $\mathtt{Bac}$.

2.4. Reversing diagrams. It is useful to visualize the reversing process by associating with every reversing sequence a certain planar diagram (which is connected with what is called a *van Kampen diagram* in general).

Assume that C is a complement on S, and $(w_0, ..., w_N)$ is a reversing sequence for C. The associated reversing diagram will contain right-oriented horizontal and down-oriented vertical edges labeled by letters s of S, plus (possibly) ε-labeled arcs.

First, we draw a connected path indexed by the successive letters of w_0 by attaching a horizontal arrow $\xrightarrow{s}$ with each letter s, and a vertical arrow $\downarrow^s$ with each letter s^{-1}. Then, we inductively complete the diagram as follows. Assume that one goes from w_{k-1} to w_k by reversing some subword $s^{-1}t$. By induction hypothesis, the latter subword $s^{-1}t$ corresponds to an open pattern $s \downarrow \quad \xrightarrow{t}$ in the diagram.

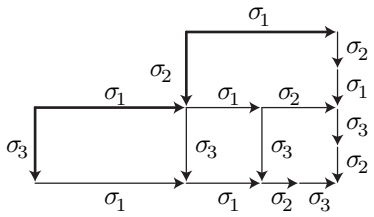
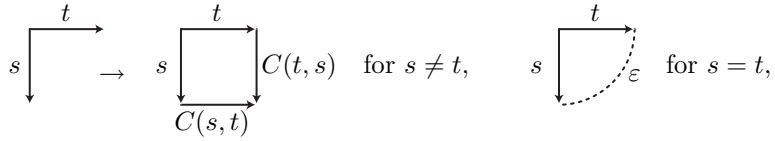


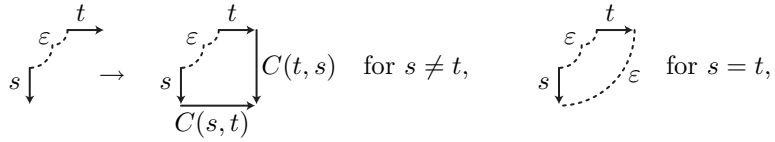
FIGURE 3. Reversing diagram starting from $\sigma_3^{-1}\sigma_1\sigma_2^{-1}\sigma_1$: one starts with a staircase labelled $\sigma_3^{-1}\sigma_1\sigma_2^{-1}\sigma_1$ by drawing a vertical σ_i-labelled arrow for each letter σ_i^{-1}, and an horizontal σ_i-labelled arrow for each positive letter σ_i. Then, when $\sigma_i^{-1}\sigma_j$ is reversed into $C(\sigma_i,\sigma_j)C(\sigma_j,\sigma_i)^{-1}$, we complete the open pattern corresponding to $\sigma_i^{-1}\sigma_j$ into a square by adding horizontal arrows labeled $C(\sigma_i,\sigma_j)$ and vertical arrows labeled $C(\sigma_j,\sigma_i)$.

Then we complete that pattern with new arrows, according to the rule



with the convention that ε-labeled dotted arcs are subsequently ignored. For instance, the reversing diagram associated with the reversing sequence (2.9) is displayed in Figure 3 (and a more complex example is given in Figure 5 below).

Remark 2.11.— Because of the ε-labelled arcs, the above patterns are not the most general ones appearing in a reversing diagram. The general patterns are actually



In this way, we associated with every reversing sequence a reversing diagram. Conversely, it is easy to see that, starting with a diagram as above, we can recover a (not necessarily unique) reversing sequence by reading the labels of the various paths going from the bottom-left corner to the top-right corner, and using the convention that a vertical σ_i-labeled edge contributes σ_i^{-1}.

The following result should be clear:

Lemma 2.12.— *For each $S^{\pm}$-word w, there is a unique maximal reversing diagram $\mathcal{D}_w$ starting from w (up to changing the lengths of the edges).*

Note that the diagram $\mathcal{D}_w$ need not be finite in general, *cf.* Exercise 2.10.

Definition 2.13 (extended complement).— Assume that C is a complement on an alphabet S. We define a (possibly partial) mapping $C^* : S^* \times S^* \to S^*$ as follows: $C^*(u,v) = v'$ holds if and only if there exists a word u' on S such that $u^{-1}v$ reverses to $v'u'^{-1}$.

Lemma 2.14.— (*i*) *If u and v have length one, i.e., if they are letters in S, then $C^*(u,v)$ exists and is equal to $C(u,v)$.*

(*ii*) *The words $C^*(u,v)$ and $C^*(v,u)$ exist if and only if the reversing diagram $\mathcal{D}_{u^{-1}v}$ is finite, and, in this case, we have*

$$u\,C^*(u,v) \equiv^+ vC^*(v,u), \tag{2.15}$$

where $\equiv^+$ is the congruence on S^* generated by the relations R_C.

PROOF. Point (*i*) is clear. As for (*ii*), we observe that, if u, v are any two positive words labelling two paths in a reversing diagram with the same origin and the same end, then $u \equiv^+ v$ holds. This follows from an induction on the number of tiles lying between the two considered paths. When the number is one, the only possibility is that the words have the form $sC(s,t)$, $tC(t,s)$ for some letters s, t, and the result is true by definition of the relations R_C. □

2.5. The completeness property. In general, not much can be said about monoids that are associated with a complement. In particular, it is impossible to recognize in general whether they are

cancellative or not. However, there is one good case where this is possible, namely when the considered complement turns out to be *complete*.

Definition 2.16 (completeness).— Assume that C is a complement on S. We say that C is *complete* if the operation C^* is compatible with the congruence $\equiv^+$ generated by R_C, in the following sense: for all words u, v, u', v' in S^*, if $C^*(u, v)$ is defined and if $u' \equiv^+ u$ and $v' \equiv^+ v$ hold, then $C^*(u', v')$ is defined as well and we have $C^*(u', v') \equiv^+ C^*(u, v)$.

Lemma 2.17.— *Assume that C is a complete complement. Then, for all words u, v in S^*, we have*

$$u \equiv^+ v \quad \Leftrightarrow \quad C^*(u,v){=}C^*(v,u){=}\varepsilon \quad \Leftrightarrow \quad u^{-1}v \curvearrowright \varepsilon. \tag{2.18}$$

Proof. The right equivalence is just the definition of C^*.

Assume $u \equiv^+ v$. For each complement C and each word u on S, we have $u^{-1}u \curvearrowright \varepsilon$, hence $C^*(u, u) = \varepsilon$. If, in addition, C is complete, we deduce $C^*(u, v) \equiv^+ \varepsilon$ and $C^*(v, u) \equiv^+ \varepsilon$ by stusbtituting u with v in one of the two arguments of $C^*(u, u) = \varepsilon$. Then, we observe that, by construction, all relations of R_C have nonempty left and right hand terms. So $w \equiv^+ \varepsilon$ implies $w = \varepsilon$. So, $u \equiv^+ v$ implies $C^*(u, v){=}C^*(v, u){=}\varepsilon$.

Conversely, if we have $C^*(u, v) = C^*(v, u) = \varepsilon$, then (2.15) directly gives $u \equiv^+ v$ (without using any completeness assumption). □

Completeness of the complement directly implies a cancellability result.

Proposition 2.19.— *Assume that C is a complete complement. Then the monoid associated with C is left-cancellative.*

Proof. Assume that C is a complement on the alphabet S, and M is the monoid associated with C. Proving that M is left-cancellative amounts to proving that, for all words u, v on S, and for each letter s in S, the relation $su \equiv^+ sv$ implies $u \equiv^+ v$ (where $\equiv^+$ is the congruence generated by R_C).

Assume $su \equiv^+ sv$. By Lemma 2.17, we deduce $(su)^{-1}(sv) \curvearrowright \varepsilon$, *i.e.*, $u^{-1}s^{-1}sv \curvearrowright \varepsilon$. Now, by definition, the first step in reversing the latter word is $u^{-1}s^{-1}sv \curvearrowright u^{-1}v$ (there is no other possibility). So we necessarily have $u^{-1}v \curvearrowright \varepsilon$, hence, by Lemma 2.17 again, $u \equiv^+ v$. □

So the strategy is clear: in order to prove that the monoid B_n^+ is left-cancellative, it suffices to prove that the braid complement of (2.5) is complete. As there are infinitely many braid words, this is in principle difficult. However, we shall see now that, instead of having to verify the compatibility of C^* and $\equiv^+$ for all words, it is sufficient to verify it for a small family of words only.

Lemma 2.20.— *Assume that C is a complement on S satisfying the following two properties:*

(i) For all s, t in S, we have $|C(s, t)| = |C(t, s)|$;

(ii) For all r, s, t in S, we have

$$C^*(r, sC(s,t)) \equiv^+ C^*(r, tC(t,s)) \quad \text{and} \quad C^*(sC(s,t), r) \equiv^+ C^*(tC(t,s), r), \tag{2.21}$$

this meaning, as usual, that neither side of the relation is defined, or that both are defined and they are equivalent. Then C is complete.

The meaning of (2.21) is clear: this is an instance of the compatibility of C^* and $\equiv^+$, namely the case when one of the words is just a letter r, and the other is a word involved in the basic relations of R_C, *i.e.*, the simplest possible type of word equivalence. So what Lemma 2.20 says is that, provided the relations of R_C preserve the length (condition (i)), then the most simple case of compatibility is enough to guarantee the general case.

Proof of Lemma 2.20. We shall prove using induction on ℓ the implication:

$$(\mathcal{P}_\ell) \qquad \begin{array}{l} \text{If } C^*(u,v) \text{ is defined and } |uC^*(u,v)| \leqslant \ell \text{ holds,} \\ \qquad \text{then } (u' \equiv^+ u \ \&\ v' \equiv^+ v) \text{ implies } (C^*(u',v') \equiv^+ C^*(u,v) \ \&\ C^*(v',u') \equiv^+ C^*(v,u)). \end{array}$$

First, $(\mathcal{P}_0)$ is obvious. As the first and the second argument play symmetric roles, $(\mathcal{P}_\ell)$ follows from

$$(\mathcal{P}'_\ell) \qquad \begin{array}{l} \text{If } C^*(u,v) \text{ is defined and } |uC^*(u,v)| \leqslant \ell \text{ holds,} \\ \qquad \text{then } v' \equiv^+ v \text{ implies } (C^*(u,v') \equiv^+ C^*(u,v) \ \&\ C^*(v',u) \equiv^+ C^*(v,u)). \end{array}$$

Now saying that v' is $\equiv^+$-equivalent to v means that there exists an R_C-derivation of v' from v, *i.e.*, v' is obtained from v by appying a sequence of relations of R_C. For an obvious induction, it is enough that we prove $(\mathcal{P}_\ell)$ when v' is obtained from v by applying one relation of R_C. This means that there exist s, t

in S and v_1, v_2 in S^* satisfying $v = v_1 sC(s,t)v_2$ and $v' = v_1 tC(t,s)v_2$. So, in order to prove $(\mathcal{P}_\ell)$, it is enough that we assume $(\mathcal{P}_{\ell-1})$ and prove

$(\mathcal{P}''_\ell)$ Assume $v = v_1 sC(s,t)v_2$, $v' = v_1 tC(t,s)v_2$ and $|uC^*(u,v)| \leqslant \ell$.
Then we have $(C^*(u,v') \equiv^+ C^*(u,v)$ & $C^*(v',u) \equiv^+ C^*(v,u))$.

We consider the reversing diagram for $u^{-1}v$, which is assumed to be finite, and compare it with the reversing diagram for $u^{-1}v'$, which a priori need not be finite. Now see Figure 4. The hypothesis that $C^*(u,v)$ exists implies that there exist a letter r and words $u_0, ..., u_4, v_0, ..., v_6$ as indicated on the left diagram[2]—or that $u^{-1}v_1 \curvearrowright v_0$ holds, in which case there is no letter r, and everything is obvious. The hypothesis (2.21) implies that u'_1 and v'_3 exist, and that one has $u'_1 \equiv^+ u_1$ and $v'_3 \equiv^+ v_3$. Then we look at the reversing of $u_0^{-1}v_3$ and $u_0^{-1}v'_3$.

The point is as follows: by hypothesis, $C^*(u_0, v_3)$ exists (this is the word w_4) and, by construction, the length of $u_0 C^*(u_0, v_3)$, *i.e.*, of $u_0 v_4$, is at most $\ell - 1$, because we have

$$|uC^*(u,v)| = |v_1 r u_0 v_4 v_6| \leqslant \ell,$$

and r has length one. So the induction hypothesis $\mathcal{P}_{\ell-1}$ implies that $C^*(u_0, v'_3)$ exists and is equivalent to $C^*(u, v_3)$, so v'_4 exists, and we have $v'_4 \equiv^+ v_4$ and, similarly, $u'_2 \equiv^+ u_2$.

The same argument applies to the reversings of $u_1^{-1}v_2$ and $u'_1 v'_2$ as, now, the letter s forces $|u_1 C^*(u_1, v_2)| \leqslant \ell - 1$. So u'_3 and v'_5 exist and satisfy $u'_3 \equiv^+ u_3$ and $v'_5 \equiv^+ v_5$.

Finally, the same argument applies to u_2 and v_5, proving the existence of u'_4 and v'_6 that satisfy $u'_4 \equiv^+ u_4$ and $v'_6 \equiv^+ v_6$. We deduce that the reversing of $u^{-1}v'$ terminates in finitely many steps with the word $v_0 v'_4 v'_6 {u'_4}^{-1} {u'_3}^{-1}$, and we find $C^*(u,v') = v_0 v'_4 v'_6 \equiv^+ v_0 v_4 v_6 = C^*(u,v)$ and $C^*(v',u) = u'_3 u'_4 \equiv^+ u_3 u_4 = C^*(u,v)$. This proves $\mathcal{P}_\ell$, and completes the induction. □

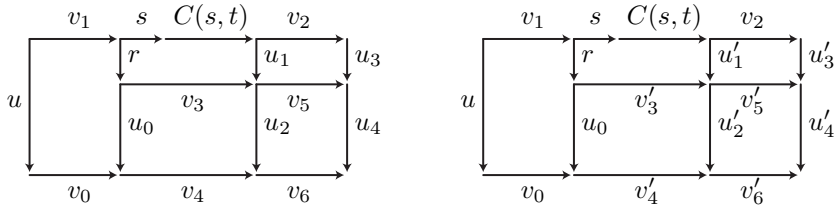


FIGURE 4. Proof of Lemma 2.20: Comparing the reversing diagrams for $u^{-1}v$ and $u^{-1}v'$.

We are nearly done.

Lemma 2.22.— *The braid complement C is complete.*

PROOF. It suffices to check that the braid complement (2.5) satisfies the hypotheses of Lemma 2.20. Now, as for (i), we see that $|C(\sigma_i, \sigma_j)| = |C(\sigma_j, \sigma_i)|$ is true for all i, j.

As for (ii), we have to consider all triples $\{\sigma_i, \sigma_j, \sigma_k\}$. It should be clear that only the relative distances, 0, 1, or $\geqslant 2$, matter. The result is straightforward when at least two indices coincides. So, essentially, it remains the three cases $(1,3,5)$, $(1,2,4)$, and $(1,2,3)$ and their cyclic permutations. Here is a typical verification for the case $1,2,3$ (there are two more cyclic permutations to consider).

$$\sigma_1^{-1}\sigma_2\sigma_3\sigma_2 \curvearrowright \sigma_2\sigma_1\sigma_3\sigma_2\sigma_1{\sigma_3}^{-1}{\sigma_2}^{-1}{\sigma_1}^{-1} \quad \text{and} \quad \sigma_1^{-1}\sigma_3\sigma_2\sigma_3 \curvearrowright \sigma_3\sigma_2\sigma_1\sigma_3\sigma_2{\sigma_3}^{-1}{\sigma_2}^{-1}{\sigma_1}^{-1}$$

whence

$$C^*(\sigma_1, \sigma_2\sigma_3\sigma_2) = \sigma_2\sigma_1\sigma_3\sigma_2\sigma_1 \equiv^+ \sigma_2\sigma_3\sigma_1\sigma_2\sigma_1 \equiv^+ \sigma_2\sigma_3\sigma_2\sigma_1\sigma_2 \equiv^+ \sigma_3\sigma_2\sigma_3\sigma_1\sigma_2 \equiv^+ \sigma_3\sigma_2\sigma_1\sigma_3\sigma_2 = C^*(\sigma_1, \sigma_3\sigma_2\sigma_3),$$

and

$C^*(\sigma_2\sigma_3\sigma_2, \sigma_1) = \sigma_1\sigma_2\sigma_3 = C^*(\sigma_3\sigma_2\sigma_3, \sigma_1)$. The other cases are similar. □

Putting the pieces together we deduce

Proposition 2.23.— *For each n, the monoid B_n^+ is cancellative.*

PROOF. The criterion of Proposition 2.19 applies, and we deduce that B_n^+ is left-cancellative.

As for right-cancellation, we observe that, due to the symmetry of the braid relations, there exists an involutive antiautomorphism ι of B_n^+ that, for each i, maps σ_i to itself: ι is the antihomomorphism that reverses the order of letters—so, for the corresponding braid diagrams, it corresponds to taking a mirror-image with respect to a mirror orthogonal to the main direction of the strands. Now assume

[2] or that u_0 is empty, in which case the argument is similar and simpler

$ab = a'b$ in B_n^+. Then we have $\iota(b)\iota(a) = \iota(b)\iota(a')$, hence $\iota(a) = \iota(a')$ since B_n^+ is left-cancellative, hence $a = a'$ since ι is involutive (applying ι twice is the identity). □

Corollary 2.24.— (*i*) *For each n, the monoids* $\overline{B_n^+}$ *and* B_n^+ *are isomorphic, i.e., the monoid* $\overline{B_n^+}$ *admits the presentation* (1.2).

(*ii*) *For all positive braid words* u, v, *we have*

$$u \equiv v \quad \Leftrightarrow \quad u \equiv^+ v. \tag{2.25}$$

Proof. (*i*) The monoid B_n^+ is cancellative and admits common right-multiples, hence it is eligible for Ore's Theorem. Hence, by Corollary 1.36, B_n^+ is isomorphic to the submonoid of the group B_n generated by $\sigma_1, ..., \sigma_{n-1}$, which, by definition, is $\overline{B_n^+}$.

(*ii*) We know that $u \equiv^+ v$ always implies $u \equiv v$. Conversely, $u \equiv v$ means that u and v represent the same element of $\overline{B_n^+}$, hence by (*i*) the same element of B_n^+, hence that $u \equiv^+ v$ holds. □

WE ARE DONE:

Theorem 2.26.— *The Braid Isotopy Problem is decidable.*

Proof. The proof scheme planned at the beginning of this section has been completed. □

Exercise 2.27.— Show that the conclusion of Lemma 2.20 remains valid when Condition (*i*) is relaxed to: There exists a map: $\lambda : S^* \to \mathbb{N}$ such that $u' \equiv^+ u$ implies $\lambda(u') = \lambda(u)$ and $\lambda(su) > \lambda(u)$ holds for each letter s of S.

Exercise 2.28.— Show that Condition (*ii*) of Lemma 2.20 is also true when S has two elements.

3. Algorithms

We just have seen that the Braid Isotopy Problem is decidable, *i.e.*, there exist algorithms that solve it. We first summarize the algorithm that comes from the above argument. Then we observe that, owing to the auxiliary results that have been proved in the meanwhile, we can describe better algorithms.

3.1. The stupid monoid algorithm. The algorithm that directly comes from Section 2 is as follows.

Algorithm 3.1.— *Input: Two n-strand braid diagrams* D, D'.
Ouput: YES if D *and* D' *are isotopic, NO otherwise.*
Method: - Encode $\widetilde{D}D'$ *into a braid word* w;
- Run Algorithm 1.21 on w *to obtain* d *and* w_1;
- Run Algorithm 1.27 on Δ_n^{2d} *and* w_1, *and return its answer.*

Proof. First, D and D' are isotopic if and only if $w \equiv \varepsilon$ holds. Then, by construction, we have$\Delta_n^{2d} w \equiv^+ w_1$, hence $w \equiv \varepsilon$ is equivalent to $\Delta_n^{2d} \equiv w_1$. By Corollary 2.25(*ii*), the latter is equivalent to $\Delta_n^{2d} \equiv^+ w_1$, and this is what Algorithm 1.27 tests. □

Example 3.2.— Let D, D' be the resisting diagrams of Chapter I. We saw that $\widetilde{D}D'$ is encoded in $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$, and found in Example 1.23 the values

$$d = 4, \qquad w_1 = \texttt{ababaababababaababaababbaa}.$$

It remains to run Algorithm 1.27 on the words $(\texttt{aba})^8$ and w_1 above, *i.e.*, to enumerate all positive words equivalent to $(\texttt{aba})^8$. *If* we were doing it, we would never see w_1 in the list, and we *would* conclude that D is not trivial—but we shall not do it, because the list of words equivalent to $(\texttt{aba})^8$ is very long.

Remark 3.3.— Initially, our aim was to decide whether the braids represented by $\sigma_1^2\sigma_2^2$ and $\sigma_2^2\sigma_1^2$ are isotopic. The latter braid words are positive, and forming the quotient is stupid. Actually, we can immediately see, in this very special case, that $\sigma_1^2\sigma_2^2$ and $\sigma_2^2\sigma_1^2$ are not equivalent: indeed, $\sigma_1^2\sigma_2^2 \equiv^+ \sigma_2^2\sigma_1^2$ is impossible, as no braid relation can be applied to $\sigma_1^2\sigma_2^2$ or $\sigma_2^2\sigma_1^2$, so this words are alone in their equivalence class. So, in this way, we did prove that the braids represented by $\sigma_1^2\sigma_2^2$ and $\sigma_2^2\sigma_1^2$ are not isotopic.

3.2. Using reversing. A much better algorithm can be obtained by replacing the systematic enumeration of Algorithm 1.27 with subword reversing.

Proposition 3.4.— *For all positive braid words u, v, reversing $u^{-1}v$ terminates in a finite number of steps. In other words, $C^*(u, v)$ is always defined.*

PROOF. Let u, v be arbitrary positive words. By Proposition 2.1, the braids represented by u and v admit a common right-multiple in $\overline{B_n^+}$, *i.e.*, there exist positive braid words u', v' satisfying $uv \equiv^+ vu'$. By Lemma 2.17, this implies $(uv')^{-1}(vu') \curvearrowright \varepsilon$. Hence the reversing diagram for $(uv')^{-1}(vu')$, *i.e.*, from $v'^{-1}u^{-1}vu'$, is finite. But then the reversing diagram for $u^{-1}v$, which is included in the latter, is finite as well. □

We deduce a new algorithm for solving the Word Problem of $\overline{B_n^+}$ with respect to $\{\sigma_1, ..., \sigma_{n-1}\}$.

Algorithm 3.5.—
Input: Two positive braid words u, v.
Ouput: YES, if u and v are positively equivalent, NO otherwise;
Method: - Reverse $u^{-1}v$;
- Return YES if the final word is empty, and NO otherwise.

Note that it is crucial to know that reversing always terminates: otherwise, we would not solve the Word Problem, as we would risk to never conclude in the case when the initial words are not equivalent.

Example 3.6.— Assume $u = \sigma_1^2\sigma_2^2$ and $v = \sigma_2^2\sigma_1^2$. Reversing $u^{-1}v$ leads to $\sigma_1\sigma_2^3\sigma_1\sigma_2^{-1}\sigma_1^{-3}\sigma_2^{-1}$, a nonempty word, so u and v are not equivalent (see Figure 5).

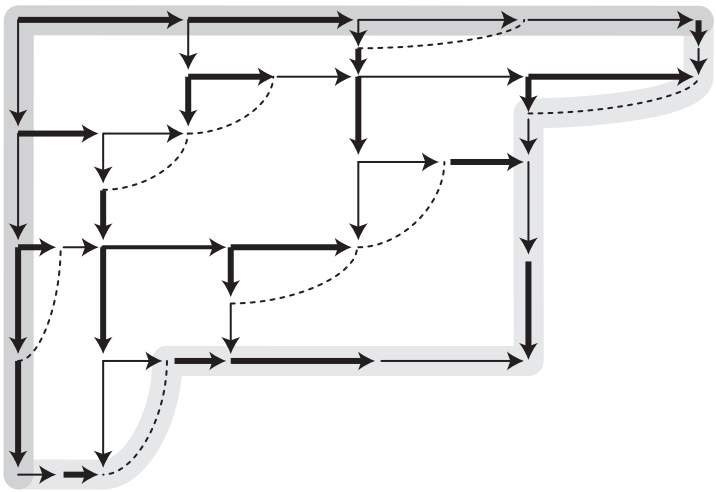


FIGURE 5. Reversing of the braid word $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$. Thin arrows correspon to σ_1, thick arrows correspond to σ_2. One starts with the dark grey path, and finishes with the light grey path, which is terminal as it contains no $\sigma_i^{-1}\sigma_j$.

Inserting Algorithm 3.5 in Algorithm 3.1 gives a more realistic solution to the Braid Triviality Problem.

Algorithm 3.7.— *Input: Two n-strand braid diagrams D, D'.*
Ouput: YES if D and D' are isotopic, NO otherwise.
Method: - Encode $\widetilde{D}D'$ into a braid word w;
- Run Algorithm 1.21 on w to obtain d and w_1;
- Run Algorithm 3.5 on Δ_n^{2d} and w_1, and return its answer.

3.3. More reversing. Instead of using the word Δ_n to eliminate negative letters, we can use reversing.

Proposition 3.8.— *For every braid word w, reversing w terminates in a finite number of steps.*

PROOF. Using induction on p, we show the result for w of the form $u_1^{-1}v_1u_2^{-1}v_2...u_p^{-1}v_p$ with $u_1, ..., v_p$ positive. For $p = 1$, this is Proposition 3.4. Assume $p \geqslant 2$. By Proposition 3.4, there exist positive

words u'_1, v'_1 satisfying $u_1^{-1}v_1 \curvearrowright v'_1 u'_1{}^{-1}$. By induction hypothesis, there exist positive words u'_2, v'_2 satisfying $u_2^{-1}v_2...u_p^{-1}v_p \curvearrowright v'_2 u'_2{}^{-1}$. By Proposition 3.4 again, there exist positive words u'_3, v'_3 satisfying $u'_1{}^{-1}v'_2 \curvearrowright v'_3 u'_3{}^{-1}$. Then we find

$$w \curvearrowright v'_1 u'_1{}^{-1} u_2^{-1} v_2 ... u_p^{-1} v_p \curvearrowright v'_1 u'_1{}^{-1} v'_2 u'_2{}^{-1} \curvearrowright v'_1 v'_3 u'_3{}^{-1} u'_2{}^{-1},$$

hence $w \curvearrowright (v'_1 v'_3)(u'_2 u'_3)^{-1}$, as expected. □

We deduce a new solution to the Braid Isotopy Problem using two reversings.

Algorithm 3.9.—
Input: Two n-strand braid diagrams D, D'.
Ouput: YES if D and D' are isotopic, NO otherwise.
Method: - Encode $\widetilde{D}D'$ into a braid word w;
- Reverse w into vu^{-1} with u, v positive;
- Run Algorithm 3.5 on u and v, i.e., reverse $u^{-1}v$;
- Return YES if the final word is empty, NO otherwise.

Example 3.10.— Starting with D, D' as in Example 1.23, we encode $\widetilde{D}D'$ into $(\mathtt{a}^2\mathtt{b}^2)^{-1}(\mathtt{b}^2\mathtt{a}^2)$. The first reversing leads to $(\mathtt{ab}^3\mathtt{a})(\mathtt{ba}^3\mathtt{b})^{-1}$, see Figure 5. We exchange the numerator and the denominator, and reverse $(\mathtt{ba}^3\mathtt{b})^{-1}(\mathtt{ab}^3\mathtt{a})$. We find $(\mathtt{a}^2\mathtt{b}^2)(\mathtt{b}^2\mathtt{a}^2)^{-1}$ (the initial word); this is not the empty word, we deduce that D and D' are not isotopic.

CHAPTER IV

The Artin representation

We turn to a completely different solution to the Braid Isotopy Problem, namely one based on algebraic topology. This solution is the original one used by Emil Artin. It relies on viewing a braid as the isotopy class of a homeomorphism of a punctured disk and deducing an action on the fundamental group of a disk with holes, which is a free group.

1. The braid group as a mapping class group

For every surface (2-dimensional manifold) one introduces a certain group called the *mapping class group* of the surface. When the surface is a disk with n marked points, the mapping class group is precisely the braid group B_n.

1.1. The principle. A geometric braid (or the braid diagram that is its projection) can be seen as a picture movie of the danse of n points in a disk: see Figure 1.

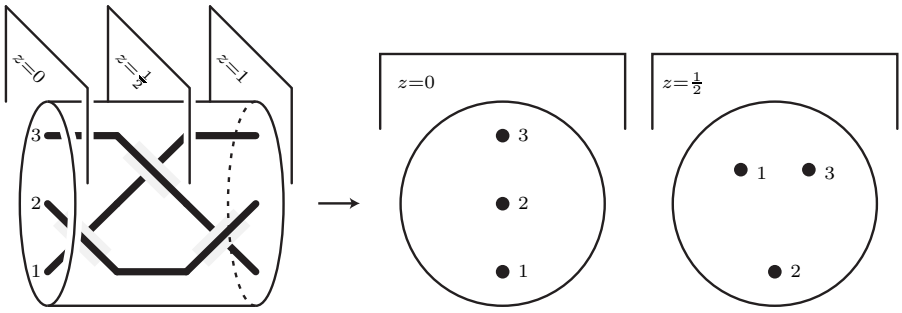


FIGURE 1. A geometric braid (drawn in a cylinder) viewed as the danse of n points in a disk (here $n = 3$): the coordinate z is the time that goes from 0 to 1; the intersection of each plane $z = z_0$ with the n strands of the braid consists of n points that continuously move in the disk.

Starting with this observation, one obtains a one-to-one correspondence between isotopy classes of geometric braids, *i.e.*, braids, and isotopy classes of homeomorphisms of the punctured disk.

1.2. The mapping class group of a surface with marked points. For each surface and each choice of finitely many points on this surface, one introduces a certain group called the mapping class group.

Definition 1.1 (mapping class group).— Let Σ be an oriented compact surface, possibly with boundary, and let $\{P_1, ..., P_n\}$ be a finite set of distinguished interior points of Σ. The *mapping class group* $\mathcal{MCG}(\Sigma, \{P_1, ..., P_n\})$ of the surface Σ relative to $\{P_1, ..., P_n\}$ is the group of all isotopy classes of orientation-preserving self-homeomorphisms of Σ that fix the boundary $\partial\Sigma$ pointwise and preserve $\{P_1, ..., P_n\}$ globally.

Mapping class groups play an important role in low-dimensional topology (study of surfaces and 3-dimensional manifolds). A deep theorem of Epstein states that two homeomorphisms of a compact surface are homotopic if and only if they are isotopic. We recall from Chapter **??** that two homeomorphisms φ, φ' of Σ into itself are called *homotopic* (*resp.isotopic*) if there exists a continuous map $F : \Sigma \times [0, 1] \to \Sigma$ satisfying $F(-, 0) = \varphi$ and $F(-, 1) = \varphi'$ (*resp.*this and, in addition, $F(-, t)$ is a homeomorphism for each t). In the case of Definition 1.1, one requires moreover that $F(-, t)$ fixes the boundary pointwise, and the set of punctures globally.

Exercise 1.2.— Show that isotopic homeomorphisms necessarily induce the same permutation of the punctures. [Hint: Use the fact that the set of all permutations of n points is finite, hence discrete.]

In the sequel we shall be interested in one particular case, namely that of a disk, and we fix some notation.

Notation 1.3.— We denote by D_n the disk of $\mathbb{R}^2$ with center $(0,(n+1)/2)$ and radius $n/2+1$, with the n marked points $(0,1)$, ..., $(0,n)$. We denote by D_n^- the complement of $\{P_1,...,P_n\}$ in D_n.

So D_n^- is a disk with n holes.

As announced above, there is a simple connection between braids and the mapping class group of D_n.

Proposition 1.4.— *The braid group B_n is isomorphic to the mapping class group $\mathcal{MCG}(D_n)$.*

Proof (Sketch). Let Γ be an n-strand geometric braid, sitting in the cylinder $D^2\times[0,1]$. By construction, the n strands are starting at the puncture points of $D_n\times\{0\}$ and ending at the puncture points of $D_n\times\{1\}$. As explained in Figure 1, Γ may be considered to be the graph of the motion, as time goes from 0 to 1, of n points moving in the disk, starting and ending at the puncture points. It can be proved that this motion extends to a continuous family of homeomorphisms of the disk, starting with the identity and fixed on the boundary at all times: think that the disk is filled with jelly: the danse of the punctures causes the jelly around them to move; if we insist that nothing moves in a neighbour of the boundary disk, then the extension is unique up to isotopy. The end map of this isotopy is the corresponding homeomorphism $\varphi: D_n\to D_n$, which is well defined up to isotopy fixing the punctures globally and the boundary pointwise.

Conversely, given a homeomorphism $\varphi: D_n\to D_n$, representing some element of the mapping class group, we want to get an n-strand geometric braid. By a trick of Alexander, every homeomorphism of a disk that fixes the boundary is isotopic to the identity, through homeomorphisms fixing the boundary. The corresponding braid is then the graph of the restriction of such an isotopy to the marked points. □

So, from now on, we can view an n-strand braid as an isotopy class of homeomorphisms of D_n that leave ∂D_n, and preserve the n punctures globally.

2. The fundamental group

One associates with every topological space various *homotopy invariants*, *i.e.*, objects (numbers, functions, groups, algebras) that only depend on the homeomorphism type of the considered space and that are constructed by means of homotopy classes. One of the main such homotopy invariant is the *fundamental group*.

2.1. Loops. In the sequel, a curve γ in the plane, or more generally in any surface, is identified with a parametrization, *i.e.*, with a continuous map of the interval $[0,1]$ of $\mathbb{R}$ to the considered surface, of the form $t\mapsto(x(t),y(t))$.

Definition 2.1 (loop).— Let X be a topological space, and P_0 be a point in X. A *loop* in X with basepoint P_0 is a continuous map $\gamma:[0,1]\to X$ satisfying $\gamma(0)=\gamma(1)=P_0$. The set of all loops in X with basepoint P_0 is denoted $\Omega_1(X;P_0)$.

Then, as above, we have the natural notion of *homotopic* loops. Two loops γ,γ' with basepoint P_0 in X are called homotopic, denoted $\gamma\sim\gamma'$ if there exists $F:[0,1]\times[0,1]\to X$ satisfying $F(-,0)=\gamma$, $F(-,1)=\gamma'$, and $F(0,t)=F(1,t)=P_0$ for every t. So two loops are homotopic if one can continuously deform one to the other.

Notation 2.2.— The quotient-set $\Omega_1(X;P_0)/\sim$ is denoted $\pi_1(X;P_0)$. For f in $\Omega_1(X;P_0)$, the homotopy class of a loop γ is denoted $[\gamma]$.

As we did in Chapter **??** with geometric braids, we can define a binary operation on loops.

Definition 2.3 (product).— For γ_1,γ_2 in $\Omega_1(X;P_0)$, the *product* of γ_1 and γ_2 is defined by

$$\gamma_1\gamma_2(t)=\begin{cases}\gamma_1(2t) & \text{for } 0\leqslant t\leqslant 1/2,\\ \gamma_2(2t-1) & \text{for } 1/2\leqslant t\leqslant 1.\end{cases}\tag{2.4}$$

The product is well defined since the values are coherent at $1/2$.

Lemma 2.5.— *The product of loops is compatible with homotopy, and it induces a group structure on $\pi_1(X;P_0)$.*

PROOF. Do it. □

Lemma 2.6.— *Assume that X is a path-connected space, i.e., for any two points P, P' in X, there exists at least one continuous map $\pi : [0,1] \to X$ satisfying $\pi(0) = P$ and $\pi(1) = P'$ (called a* path *from P to P'). Then, for all P_0, P_0' in X, the groups $\pi(X; P_0)$ and $\pi(X; P_0')$ are isomorphic.*

PROOF. Let π be a fixed path connecting P_0 to P_0' in X. Then the map $\gamma \mapsto \pi^{-1}\gamma\pi$ induces a well defined morphism of $\pi(X; P_0)$ and $\pi(X; P_0')$ (why?), and the map $\gamma \mapsto \pi\gamma\pi^{-1}$ induces a well defined morphism of $\pi(X; P_0')$ and $\pi(X; P_0)$ which is the inverse of the previous one. Hence both are isomorphisms. □

(Above and as in the case of braids, we always denote by fg the product "f then g", *i.e.*, $g \circ f$.)

Owing to Lemma 2.6, we can forget about basepoints, and put:

Definition 2.7 (fundamental group).— Assume that X is a path-connected space. The *fundamental group* $\pi_1(X)$ of X is the (isomorphism class) of $\pi_1(X; P_0)$, where P_0 is any point of X.

Exercise 2.8.— Show that the fundamental group is a homeomorphism invariant, *i.e.*, that homeomorphic spaces have the same fundamental group (up to isomorphism).

Exercise 2.9.— Show that the fundamental group of the plane is the trivial group. [Hint: For each loop γ with basepoint P_0, define a homotopy that contracts γ to the constant loop of value P_0.]

2.2. The fundamental group of a disk with one hole. Our aim now is to compute the fundamental group of a disk with n holes. We begin with $n = 1$.

Proposition 2.10.— *The fundamental group of a circle is $\mathbb{Z}$.*

PROOF. Let S^1 be the circle $x^2 + y^2 = 1$ in $\mathbb{R}^2$, and P_0 be the point $(1, 0)$. For each integer n, let γ_n be the curve $t \mapsto (\cos(2\pi nt), \sin(2\pi nt))$. Then γ_n is drawn inside S^1, and both the initial and final ends are the point P_0. So γ_n is a loop based on P_0 in S^1, *i.e.*, it belongs to $\Omega_1(S^1; P_0)$. We recall that $[\gamma_n]$ denotes the homotopy class of γ_n. We shall see that $I : n \mapsto [\gamma_n]$ defines an isomorphism of $\mathbb{Z}$ to $\pi_1(S^1)$.

First, I is a homomorphism: there is a homotopy from $\gamma_n\gamma_m$ to γ_{n+m} (write it explicitly).

We have to prove that I is injective, and surjective. To do that, we use the covering of S^1 by a helix—hence by $\mathbb{R}$—illustrated in Figure 2. So let H be the circular helix $t \mapsto (\cos t, \sin t, t)$ with $t \in \mathbb{R}$, and let pr denote the restriction of the projection $\mathrm{pr} : (x, y, z) \mapsto (x, y)$ to H. Then pr is continuous, surjective. It is not injective globally, but, locally, it is: there exists a finite covering of S^1 by open subsets $U_1, ..., U_p$ (for instance, any two arcs of length ℓ satisfying $\pi < \ell < 2\pi$) such that, for each point M in U_i and any point $\widetilde{M}$ in H, there is an open neighbourhood $\widetilde{U}$ of $\widetilde{M}$ in H such that pr induces a homeomophism of $\widetilde{U}$ onto U_i.

For each integer n, let $\widetilde{P}_n$ be the point $(1, 0, n)$. The points $\widetilde{P}_n$ are the various preimages of P_0 in H. Let γ be a loop in S^1 with basepoint P_0. Then we can *lift* γ (*i.e.*, find a path that projects on γ) into a unique path $\widetilde{\gamma}$ in H that starts with $\widetilde{P}_0$, : the uniqueness follows from the property that pr is a homeomorphism on each arc U_i. We claim that γ is homotopic to γ_n if and only if $\widetilde{\gamma}$ finishes at $\widetilde{P}_n$. Since $\widetilde{\gamma_n}$ finishes at $\widetilde{P}_n$, this will show that I is bijective.

The reason why the claim is true is that, if F is a homotopy between two loops γ, γ' in S^1, then F lifts into a unique homotopy between the paths $\widetilde{\gamma}$ and $\widetilde{\gamma'}$, always because pr is locally a homeomorphism. Then it is impossible that the liftings of two homotopic loops finishes at different endpoints $\widetilde{P}_n$ because the restriction of the homotopy to the final point is a continuous map of $[0, 1]$ into a discrete set, hence it is constant. So I is injective.

On the other hand, if γ is any loop in S^1 such that $\widetilde{\gamma}$ finishes at $\widetilde{P}_n$, then there is the path $\widetilde{\gamma} \cdot \widetilde{\gamma_n}^{-1}$ is a loop in H. The fundamental group of H is trivial, hence there is a homotopy of that loop to the constant loop, and projecting this homotopy gives a homotopy of $\gamma\gamma_n^{-1}$ to the constant loop of S^1. So I is surjective. □

Corollary 2.11.— *The fundamental group of D_1^- is $\mathbb{Z}$.*

PROOF. A disk with one hole is just a thickened version of a circle. Every loop in the circle is a loop in D_1^-. Conversely, using radial projection, we associate with every loop in D_1^- a homotopic loop in S^1. □

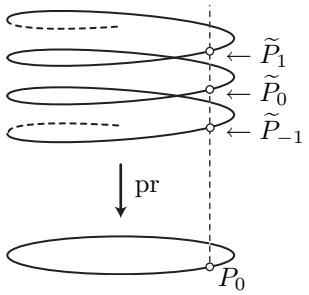


FIGURE 2. Every loop in the circle can be lifted into a path in the helix that covers it; what matters is the final point of that path.

2.3. The fundamental group of a disk with n holes. Up to a homeomorphism, a disk with n holes can be obtained by gluing one besides the other n disks with one hole. So we need a method for determining the fundamental group of a space obtained by gluing two (or any finite number of) spaces starting with the fundamental groups of the spaces. This is what the Van Kampen Theorem does.

Proposition 2.12 (van Kampen Theorem, special case**).**— *Assume that X is a topological space, and X_1, X_2 are open, path-connected subspaces of X that cover X (i.e., X is $X_1 \cup X_2$). Assume that $\pi_1(X_i)$ has the presentation $\langle S_i \mid R_i\rangle$ for $i = 1, 2$, and that $X_1 \cap X_2$ is path-connected and has a trivial fundamental group. Then $\pi_1(X)$ admits the presentation $\langle S_1 \cup S_2 \mid R_1 \cup R_2\rangle$.*

PROOF (SKETCH). The hypothesis that X_1 and X_2 are path-connected implies that X is path-connected. Choose the basepoint P_0 in the intersection $X_1 \cap X_2$. Then each loop in X decomposes into a finite sequence of paths γ_k alternately inside X_1 and X_2, with endpoints in $X_1 \cap X_2$. As $X_1 \cap X_2$ is path-connected, we can assume that the ends of each γ_k is P_0. Hence each γ_k is a loop, hence a finite product of loops representing elements of S_i and their inverses. Therefore, $S_1 \cup S_2$ generates $\pi_1(X)$.

It is clear that the relations of R_1 and R_2 are true in $\pi_1(X)$. So the point is to prove that, conversely, if some loop in X is homotopic to the constant loop, then we can deform it to the constant loop using only relations from $R_1 \cup R_2$. If the decomposition of γ into loops inside X_1 and loops inside X_2 were unique, we could follow the fragments one by one and say that a trivial loop inside X_1 can be homotoped to the constant loop using relations of R_1, and similarly for the fragments in X_2 nd R_2. Some care is needed as, a priori, a homotopy from γ to the constant loop need not induce homotopies from each of the X_i-subloops to the constant loops: for instance, additional fragments may appear in the process. However, as $[0, 1]$ is compact, there is a number N such that, for every t, there are at most N fragments involved in the intermediate loop $F(-, t)$. □

The general van Kampen Theorem deals with the case when the intersection $X_1 \cap X_2$ is not assumed to have a trivial fundamental group. In this case, the generators and relations of $\pi_1(X_1 \cap X_2)$ enter the picture: roughly speaking, one has to make sure that the nontrivial loops inside $X_1 \cap X_2$ are not counted twice, once for X_1 and once for X_2. The general form is not needed here.

Proposition 2.13.— *The fundamental group of D_n^- is a free group with n generators.*

PROOF. We use induction on n. For $n = 1$, this is Corollary 2.11. Assume $n \geqslant 2$. Then D_n^- admits a covering by (a space homeomorphic to) D_{n-1}^- plus (a space homeomorphic to) D_1^-, intersecting in a disk with no hole, see Figure 3. The fundamental group of the latter is trivial. By induction hypothesis, the fundamental group of D_{n-1}^- has presentation $\langle g_1, ..., g_{n-1} \mid -\rangle$, whereas, by Corollary 2.11, that of D_1^-, which is $\mathbb{Z}$, has presentation $\langle g_n \mid -\rangle$. The van Kampen Theorem implies that the fundamental group of D_n^- admits the presentation $\langle g_1, ..., g_n \mid -\rangle$. □

In addition to the result, the van Kampen Theorem gives natural generators for the fundamental group of a disk with n holes. Indeed, first, by Corollary 2.11, the fundamental group of a disk with one hole is generated by the class of a loop that turns once around the hole. By the van Kampen Theorem, the fundamental group of a disk with n holes is generated by the classes of n loops that turn around each of the n holes, see Figure 4.

Notation 2.14.— Hereafter, we denote by γ_k the loop of D_n^- that turns once around the kth hole, and by g_n the homotopy class of γ_n, called the *standard* generators of $\pi_1(D_n^-)$.

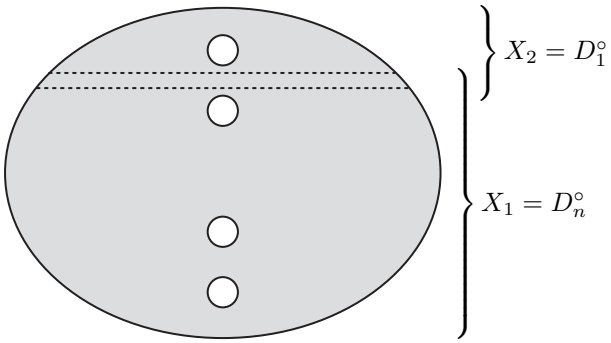


FIGURE 3. Decomposition of D_n^- into a space homeomorphic to D_{n-1}^- and a space homeomoprhic to D_1^-, with an intersection that has trivial fundamental group.

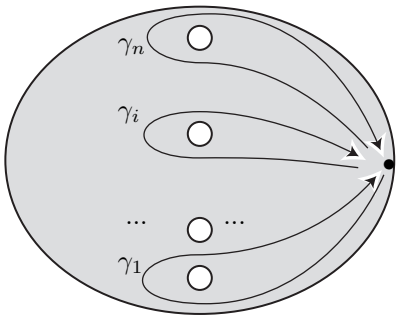


FIGURE 4. Standard generators of the fundamental group of a disk with n holes: the loop γ_k turns once around the kth hole clockwise.

3. The Artin representation

We are ready to define a mapping of the braid group B_n into the automorphisms of a free group, that will provide a new solution to the Braid Isotopy Problem.

3.1. Action of homeomorphisms on the fundamental group. For each topological space X, we construct a natural action of the isotopy classes of homeomorphisms of X into itself on the group $\pi_1(X)$.

Notation 3.1.— Hereafter we write $\mathrm{Homeo}(X)$ for the group of all homeomorphisms of a space X into itself, and $\mathrm{Homeo}(X)/\sim$ for the group of isotopy classes of homeomorphisms of X.

(We recall that γ is a map of $[0,1]$ to X.)

Proposition 3.2.— *For φ in* $\mathrm{Homeo}(X)$ *and γ a loop in X, putting*

$$\gamma \bullet \varphi(t) = \varphi(\gamma(t)) \tag{3.3}$$

induces a well defined map $\widehat{\varphi} : [\gamma] \mapsto [\gamma \bullet \varphi]$ that is an automorphism of $\pi_1(X)$, and $\rho : \varphi \mapsto \widehat{\varphi}$ is a homomorphism of $\mathrm{Homeo}(X)/\sim$ *into* $\mathrm{Aut}(\pi_1(X))$.

PROOF. First, we prove that the homotopy class of $\gamma \bullet \varphi$ depends only on the homotopy class of γ and on the isotopy class of φ. First, if F is a homotopy from γ to γ', hence is a map from $[0,1] \times [0,1]$ to X, then, as φ is a homeomorphism, $F \bullet \varphi$, defined by $(t,t') \mapsto \varphi(F(t,t'))$ is a homotopy from $\gamma \bullet \varphi$ to $\gamma' \bullet \varphi$.

Similarly, if F' is an isotopy of φ to φ', hence a map from $X \times [0,1]$ to X, then, assuming that γ is $t \mapsto (x(t), y(t))$, the map $(t,t') \mapsto F'((x(t),y(t)),t')$ is a homotopy from $\gamma \bullet \varphi$ to $\gamma \bullet \varphi'$.

So, for each homeomorphism φ of X, we have a well defined map $\widehat{\varphi}$ of $\pi_1(X)$ into itself. By construction, if γ_1, γ_2 are two loops in X, we have

$$(\gamma_1\gamma_2) \bullet \varphi = (\gamma_1 \bullet \varphi)(\gamma_2 \bullet \varphi),$$

which means that $\widehat{\varphi}$ is an endomorphism of $\pi_1(X)$. Moreover, φ has an inverse, and we obtain

$$(\gamma \bullet \varphi) \bullet \varphi^{-1} = \gamma = (\gamma \bullet \varphi^{-1}) \bullet \varphi,$$

which shows that $\widehat{\varphi^{-1}}$ is an inverse for $\widehat{\varphi}$. So $\widehat{\varphi}$ is an automorphism of $\pi_1(X)$. So, we have a map $\rho : \varphi \mapsto \widehat{\varphi}$ of $\mathrm{Homeo}(X)/\sim$ to the group $\mathrm{Aut}(\pi_1(X))$.

It remains to check that ρ is itself a homomorphism, *i.e.*, that we have $\widehat{\varphi_1\varphi_2} = \widehat{\varphi_1}\,\widehat{\varphi_2}$. That follows from (3.3) directly. □

3.2. The case of braids. Applying Proposition 3.4, we deduce:

Proposition 3.4.— *The action* (3.3) *induces a homomorphism* ρ *of* $\mathrm{Homeo}(D_n^-)/\!\sim$ *into* $\mathrm{Aut}(\pi_1(D_n^-))$.

The homomorphism ρ restricts to every group that is isomorphic to a subgroup of $\mathrm{Homeo}(D_n^-)/\!\sim$. Among such subgroups is the mapping class group of D_n, by the following observation:

Lemma 3.5.— *Restriction induces an injective homomorphism of* $\mathcal{MCG}(D_n)$ *into* $\mathrm{Homeo}(D_n^-)/\!\sim$.

Proof. Let φ be any homeomorphism of D_n that globally preserves the marked points $P_1, ..., P_n$. Then φ maps D_n^- into itself, hence its restriction to D_n^- is a homeomorphism of D_n^-. Moreover, isotopic homeomorphisms have isotopic restrictions. Hence the mapping $\varphi \mapsto \varphi|_{D_n^-}$ induces an injective homomorphism of $\mathcal{MCG}(D_n)$ into $\mathrm{Homeo}(D_n^-)/\!\sim$. □

So we deduce a homomorphism of $\mathcal{MCG}(D_n)$ into $\mathrm{Aut}(\pi_1(D_n^-)$. Now, we saw in Section 1 that $\mathcal{MCG}(D_n)$ is isomorphic to the braid group B_n, and in Section 2 that $\pi_1(D_n^-)$ is a free group of rank n. We deduce:

Corollary 3.6.— *The action* (3.3) *induces a homomorphism* ρ *of* B_n *into* $\mathrm{Aut}(F_n)$.

The homomorphism ρ is called the *Artin representation*. The nice point is that it is easy to describe the above homomorphism explicitly. Indeed, by Proposition 2.13, a generating family of $\pi_1(D_n^-)$ consist of the classes $g_1, ..., g_n$ of the loops $\gamma_1, ..., \gamma_n$ of Figure 4. On the other hand, a generating family of B_n consists of the braids $\sigma_1, ..., \sigma_{n-1}$. So, in order to specify ρ, it suffices to describe the images of the loops γ_k under the braids σ_i. This is done in Figure 5.

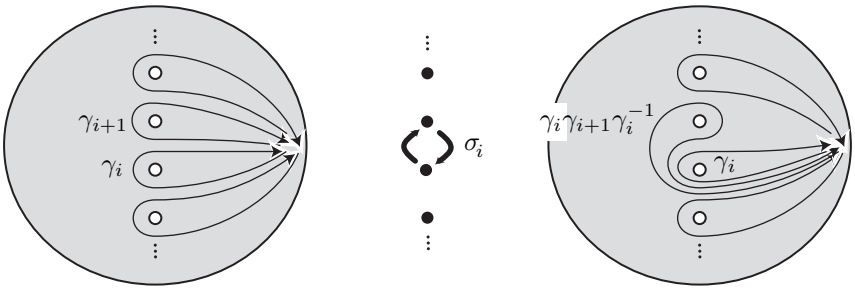


Figure 5. Action of the braid σ_i on the standard generators of $\pi_1(D_n^-)$.

Proposition 3.7.— *For each* i*, the action of the braid* σ_i *on the generators* g_k *of* $\pi_1(D_n^-)$ *is given by*

$$\rho(\sigma_i)(g_k) = g_k \bullet \sigma_i = \begin{cases} g_i g_{i+1} g_i^{-1} & \text{for } k = i, \\ g_i & \text{for } k = i+1, \\ g_k & \text{for } k \neq i, i+1. \end{cases} \tag{3.8}$$

Proof. Look at Figure 5. □

Remark 3.9.— Once the defining formulas of (3.8) are given, we may forget about their topological origin, and directly check that they define a representation of braid groups. To this end, for each braid word w, say $w = \sigma_{i_1}^{e_1} ... \sigma_{i_(\ell)}^{e_\ell}$, we define $\rho(w)$ to be the automorphism $\rho(\sigma_{i_1}^{e_1}) ... \rho(\sigma_{i_(\ell)}^{e_\ell})$. Then, in order to check that ρ induces a well defined homomorphism on B_n, it suffices to show the equalities $\rho(\sigma_i\sigma_j) = \rho(\sigma_j\sigma_i)$ for $|i-j| \geqslant 2$ and $\rho(\sigma_i\sigma_j\sigma_i) = \rho(\sigma_j\sigma_i\sigma_j)$ for $|i-j| = 1$. This is easy.

3.3. Injectivity of the Artin representation.

Theorem 3.10 (Artin, 1947)**.**— *The homomorphism* ρ *is injective.*

We shall prove Theorem 3.10 using an auxiliary result that we shall not prove, because this would require too long developments. We state this auxiliary result as a lemma.

Lemma 3.11.— *Every nontrivial braid has an expression by a braid word in which the letter σ_i with minimal index i occurs only positively (no σ_i^{-1}) or negatively (no σ_i).*

Owing to Lemma 3.11, it is enough to prove that, if w is a braid word in which the letter σ_1 occurs but σ_1^{-1} does not occur, then $\rho(w)$ is not the identity automorphism of F_n. To this end, we will show that the last letter in the freely reduced word $\rho(w)(g_1)$ is g_1^{-1}.

Notation 3.12.— For g a letter g_i or g_i^{-1}, we denote by $W(g)$ the subset of F_∞ consisting of all freely reduced words that end with the letter g.

We shall investigate the image of the set $W(g_1^{-1})$ under the automorphism $\widehat{\sigma_i}^{\pm1}$.

Lemma 3.13.— *Every automorphism $\widehat{\sigma_i}^{\pm1}$ with $i \geqslant 2$ maps $W(g_1^{-1})$ into itself.*

Proof. Consider an arbitrary element of $W(g_1^{-1})$, say vg_1^{-1} with $v \notin W(g_1)$. By construction, we have $\widehat{\sigma_i}(vg_1^{-1}) = \mathrm{red}\big(\widehat{\sigma_i}(v)\, g_1^{-1}\big)$. Assume that $\widehat{\sigma_i}(vg_1^{-1})$ does not belong to $W(g_1^{-1})$. Then the final letter g_1^{-1} in $\widehat{\sigma_i}(v)g_1^{-1}$ is cancelled by some letter g_1 occurring in $\widehat{\sigma_i}(v)$. Such a letter g_1 in $\widehat{\sigma_i}(v)$ must come from a letter g_1 in v. So there exists a decomposition $v = v'g_1v''$ satisfying $\widehat{\sigma_i}(v'') = 1$. As $\widehat{\sigma_i}$ is injective, the latter condition implies $v'' = 1$, hence $v \in W(g_1)$, which contradicts the hypothesis. The argument is the same for $\widehat{\sigma_i}^{-1}$. □

Lemma 3.14.— *The automorphism $\widehat{\sigma_1}$ maps both $W(g_1)$ and $W(g_1^{-1})$ into $W(g_1^{-1})$.*

Proof. Let us consider an arbitrary element of $W(g_1) \cup W(g_1^{-1})$, say vg_1^e with $e = \pm1$ and $v \notin W(g_1^{-e})$. Then we have $\widehat{\sigma_1}(vg_1^e) = \mathrm{red}\big(\widehat{\sigma_1}(v)g_1g_2^eg_1^{-1}\big)$. Assume $\widehat{\sigma_1}(vg_1^e) \notin W(g_1^{-1})$. This means that the final g_1^{-1} in $\widehat{\sigma_1}(vg_1^e)$ is cancelled by some letter g_1 in $\widehat{\sigma_1}(v)$. This letter comes either from some g_2 or from some $g_1^{e'}$ in v.

In the first case, we display the letter g_2 involved in the cancellation by writing $v = v'g_2v''$, where v'' is a reduced word. We find

$$\widehat{\sigma_1}(vg_1^e) = \mathrm{red}\big(\widehat{\sigma_1}(v')g_1\widehat{\sigma_1}(v'')g_1g_2^eg_1^{-1}\big). \tag{3.15}$$

By hypothesis, we have $\mathrm{red}\big(\widehat{\sigma_1}(v'')g_1g_2^e\big) = \varepsilon$, where we recall ε denotes the empty word. Hence (3.15) implies $\widehat{\sigma_1}(v'') = g_2^{-e}g_1^{-1} = \widehat{\sigma_1}(g_2^{-1}g_1^{-e})$. We deduce $v'' = g_2^{-1}g_1^{-e}$, which contradicts $v \notin W(g_1^{-e})$.

In the second case, we write similarly $u = v'g_1^{e'}v''$ with $e' = \pm1$. So we have

$$\widehat{\sigma_1}(vg_1^e) = \mathrm{red}\big(\widehat{\sigma_1}(v')g_1g_2^{e'}g_1^{-1}\widehat{\sigma_1}(v'')g_1g_2^eg_1^{-1}\big),$$

and the hypothesis is $\mathrm{red}\big(g_2^{e'}g_1^{-1}\widehat{\sigma_1}(v'')g_1g_2^e\big) = \varepsilon$. This implies $\mathrm{red}\big(\widehat{\sigma_1}(v'')\big) = g_1g_2^{-e-e'}g_1^{-1} = \widehat{\sigma_1}(g_1^{-e-e'})$, hence $v'' = g_1^{-e-e'}$. For $e = +1$, we obtain either $v'' = g_1^{-2}$ (for $e' = +1$) or $v'' = \varepsilon$ (for $e' = -1$), and, in both cases, $v \in W(g_1^{-e})$, a contradiction. Similarly, for $e = -1$, we obtain either $v'' = \varepsilon$ (for $e' = +1$) or $v'' = g_1^2$ (for $e' = -1$), and, in both cases, $v \in W(g_1^{-e})$, again a contradiction. □

We can now complete the proof of Theorem 3.10.

Proof of Theorem 3.10. Assume that β is a nontrivial braid. By Lemma 3.11, β admits an expression w in which either σ_1 or σ_1^{-1} does not appear. Assume first that σ_1 appears in w and σ_1^{-1} does not. Then w has the form $w_0\,\sigma_1\,w_1\,\sigma_1\,...\,\sigma w_p$ where the words w_k contain neither σ_1 nor σ_1^{-1}. We claim that the freely reduced word $\rho(w)(g_1)$ belongs to $W(g_1^{-1})$, hence it cannot be g_1, and $\rho(w)$, which is $\rho(\beta)$ by hypothesis, is not the identity automorphism of F_n. To prove the claim, we write

$$g_1 \bullet w = (...((g_1 \bullet w_0) \bullet \sigma_1) \bullet (w_1\,\sigma_1\,...\,w_p).$$

By construction, $g_1 \bullet w_0$ is equal to g_1. The image of the latter is $g_1g_2g_1^{-1}$, hence an element of $W(g_1^{-1})$. From there, by Lemmas 3.13 and 3.14, the successive images remain words in $W(g_1)$.

The argument is symmetric when w contains at least one letter σ_1^{-1} and no letter σ_1. Finally, if w contains neither σ_1 nor σ_1^{-1}, we similarly consider σ_2 and σ_2^{-1}, *i.e.*, we appeal to an induction on the braid index. □

3.4. Another solution to the Braid Isotopy Problem.

Corollary 3.16.— *The map ρ defines a complete braid isotopy invariant: two braid words w, w' represent isotopic braid diagrams if and only if the automorphisms $\rho(w)$ and $\rho(w')$ coincide.*

As an automorphism on a group is entirely determined by the images of the elements of a generating family, we deduce a new method for solving the Braid Isotopy Problem.

Algorithm 3.17.— *Input: Two n-strand braid diagrams D, D';*
Ouput: YES if D and D' are isotopic, NO otherwise.
Method: - Encode $\widetilde{D}D'$ into a braid word w;
- Compute the values of $\rho(w)(g_1)$, ..., $\rho(w)(g_n)$ as freely reduced words in the letters $g_k^{\pm 1}$;
- If $\rho(w)(g_k) = g_k$ holds for each k, return YES, otherwise return NO.

Example 3.18.— For $w = \sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$ as in Example III.1.23, we find

$$\rho(w)(g_1) = g_1g_2g_1^{-1}g_3g_1g_2^{-1}g_1^{-1},$$

and we deduce $w \not\equiv \varepsilon$, as the above freely reduced word is not g_1.

The algorithmic complexity of Algorithm 3.17 is poor: for w of length ℓ, the length of the words $\rho(w)(g_k)$ may be exponential in ℓ.

3.5. The Burau representation. The Artin representation gives a representation (= homomorphism) of n-strand braids in the group $\mathrm{Aut}(F_n)$. It is not hard to derive from this representation a *linear* representation, *i.e.*, a representation in a group of matrices. As this approach does not lead to a solution of the Braid Isotopy Problem, we just mention the result without details.

Proposition 3.19.— *For $1 \leqslant i \leqslant n-1$, define the $n \times n$-matrix $\rho_B(\sigma_i)$ (with entries in $\mathbb{Z}[t, t^{-1}]$) by*

$$\rho_B(\sigma_i) = I_{i-1} \otimes \begin{pmatrix} 1-t & t \\ 1 & 0 \end{pmatrix} \otimes I_{n-i-1}. \tag{3.20}$$

Then ρ_B induces a well defined linear representation of B_n into $\mathrm{GL}_n(\mathbb{Z}[t, t^{-1}])$.

Proof. As said above, one can deduce ρ_B from the Artin representation using what is called the Fox free differential calculus. However, one can also check by hand that the matrices defined in (3.20) satisfy the braid relations. □

The linear representation of Proposition 3.19 is called the *Burau representation*. It is not directly useful for the Braid Isotopy Problem, because, at least for $n \geqslant 5$, it is not a complete braid isotopy invariant: different braids may have the same image under the Burau representation.

Let us mention that there exist other linear representations of braid groups, among which some are faithful (*i.e.*, injective), hence eligible for solving the Braid Isotopy Problem, namely the so-called Lawrence–Krammer representation, which takes values in $\mathrm{GL}_{n(n-1)/2}(\mathbb{Z}[t, t^{-1}, q, q^{-1}])$.

CHAPTER V

The Dynnikov formulas

Here we present still another solution to the Braid Isotopy Problem. This solution relies on an intuition of geometry, namely using triangulations of a surface, and it involves strange formulas constructed on the operations max and + (the so-called tropical operations). Contrary to the solution based on the Artin representation, it is quite efficient from the algorithmic viewpoint. The solution was proposed by I. Dynnikov in 2000.

1. The formulas

What we shall do here is to describe the solution (which is very simple) first, and give the explanation (which is not so simple) afterwards.

1.1. The Dynnikov coordinates. For x in $\mathbb{Z}$, we put $x^+ = \max(x, 0)$ and $x^- = \min(x, 0)$.

Definition 1.1 (Dynnikov coordinates).— First we introduce deux functions F^+ and F^- of $\mathbb{Z}^4$ to $\mathbb{Z}^4$ by $F^+ = (F_1^+, ..., F_4^+)$, $F^- = (F_1^-, ..., F_4^-)$ with

$$\begin{aligned}
F_1^+(x_1, y_1, x_2, y_2) &= x_1 + y_1^+ + (y_2^+ - z_1)^+, & F_2^+(x_1, y_1, x_2, y_2) &= y_2 - z_1^+,\\
F_3^+(x_1, y_1, x_2, y_2) &= x_2 + y_2^- + (y_1^- + z_1)^-, & F_4^+(x_1, y_1, x_2, y_2) &= y_1 + z_1^+,\\
F_1^-(x_1, y_1, x_2, y_2) &= x_1 - y_1^+ - (y_2^+ + z_2)^+, & F_2^-(x_1, y_1, x_2, y_2) &= y_2 + z_2^-,\\
F_3^-(x_1, y_1, x_2, y_2) &= x_2 - y_2^- - (y_1^- - z_2)^-, & F_4^-(x_1, y_1, x_2, y_2) &= y_1 - z_2^-,
\end{aligned}$$

where we put $z_1 = x_1 - y_1^- - x_2 + y_2^+$ and $z_2 = x_1 + y_1^- - x_2 - y_2^+$.

Then one defines an action of n-strand braids on $\mathbb{Z}^{2n}$ by

$$(a_1, b_1, ..., a_n, b_n) \bullet \sigma_i^e = (a'_1, b'_1, ..., a'_n, b'_n)$$

with $a'_k = a_k$ et $b'_k = b_k$ for $k \neq i, i+1$, and

$$(a'_i, b'_i, a'_{i+1}, b'_{i+1}) = \begin{cases} F^+(a_i, b_i, a_{i+1}, b_{i+1}) & \text{for } e = +1,\\ F^-(a_i, b_i, a_{i+1}, b_{i+1}) & \text{for } e = -1. \end{cases}$$

The *Dynnikov coordinates* of an n-strand braid word w are defined to be the sequence $(0, 1, 0, 1, ..., 0, 1)\bullet w$.

Example 1.2.— Put $w = \sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$ once more. Then the Dynnikov coordinates of w turn out to be $(1, -19, -12, 9, 0, 13, 0, 1)$.

1.2. The main result. The above formulas look complicated, but they can be implemented very easily, and, then, the computation is fast. The main result is

Theorem 1.3 (Dynnikov, 2000**).**— *The coordinates of a braid word w only depend on the braid represented by w, and they characterize the latter.*

In other words, the Dynnikov coordinates make a complete isotopy invariant. We immediately deduce a new algorithm for solving the Braid Isotopy Problem.

Algorithm 1.4.— *Input: Two n-strand braid diagrams D, D';*
Ouput: YES if D and D' are isotopic, NO otherwise.
Method: - Encode $\widetilde{D}D'$ into a braid word w;
- Compute the Dynnikov coordinates of w;
- If the latter are $(0, 1, 0, 1, ..., 0, 1)$, return YES, otherwise return NO.

Example 1.5.— We saw above that the coordinates of $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$ are $(1, -19, -12, 9, 0, 13, 0, 1)$. They are not $(0, 1, 0, 1, 0, 1, 0, 1)$, so the braid $\sigma_2^{-2}\sigma_1^{-2}\sigma_2^2\sigma_1^2$ is not trivial.

The integers that appear in Dynnikov coordinates may be very large. However, adding one more $\sigma_i^{\pm 1}$ cannot do more than adding three binary digits, because it involves only max operations, which do not increase the size, and at most three additions, which in the worst case, increase the size by one digit (that would be different if multiplication were involved). It follows that the global space complexity of the method is linear, whereas the time complexity is quadratic, independently of the braid index n. (The subword reversing method of Chapter III also has a quadratic complexity, but only for each fixed value of the braid index n.)

2. Explanation

We shall now explain where do the strange formulas for the Dynnikov coordinates come from. As a preliminary remark, we observe that, as in the case of the Artin representation of Chapter IV, once we guessed the formulas for the action of σ_i, we can always check by hand that one obtains a representation of braids, *i.e.*, that the images of $\sigma_i\sigma_{i+1}\sigma_i$ and $\sigma_{i+1}\sigma_i\sigma_{i+1}$ coincide (and the other cases too).

2.1. Laminations. As in Chapter IV, one starts with the isomorphism of the braid group B_n and the mapping class group of the disk with n marked points, *i.e.*, we see a braid as an isotopy class of homeomorphisms of the disk that preserve the boundary circle pointwise and the marked points globally.

The new point here is that, instead of considering the action of the braid (actually, of the homeomorphism) on the loops that are the standard generators of $\pi_1(D_n^-)$, we look at its action on the specific family of curves L_* displayed in Figure 1. Such families of disjoint curves are generically called *laminations*.

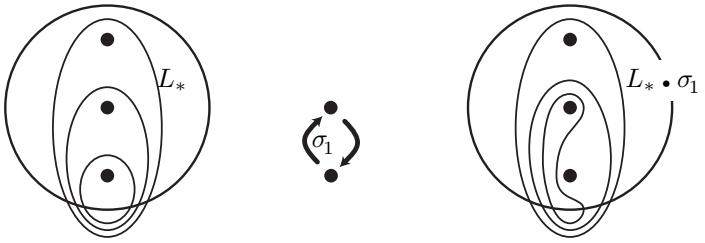


FIGURE 1. The lamination L_*, a collection of n curves surrounding the marked points of the disk D_n (here with $n = 3$), and its image under the braid (= homeomorphism) σ_1. Note that L_* is not drawn inside the disk, but rather inside a larger surface that includes it, a sphere in the current case.

By doing this, we obtain a new lamination denoted $\beta(L_*)$—or better $L_* \bullet \beta$ as we think of homeomorphisms as acting on the right (to obtain the expected order for the terms in a product).

2.2. Triangulations. The main idea is to describe laminations by counting their intersections with a fixed triangulation of D_n—or rather of a 2-sphere in which D_n is embedded once for all.

A triangulation consists of a finite number of adjacent triangles that cover the considered surface, here a 2-sphere, and are such that the intersection of any two triangles either is empty, or consists of one edge. In the sequel, we use a specific triangulation, namely the triangulation T_* displayed in Figure 2. This triangulation T_* has $n+3$ vertices, namely the n marked points of D_n, plus three points outside D_n, one of which is considered to be at infinity, viewing the sphere S^2 as a plane plus one point at infinity; T_* has $3n + 3$ edges.

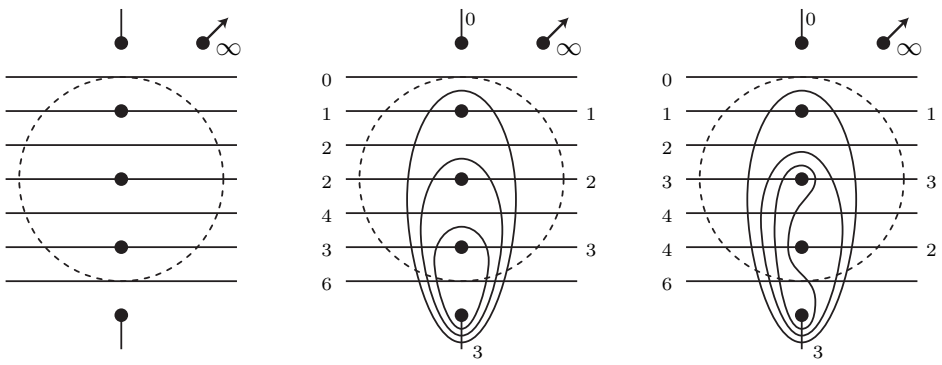


FIGURE 2. The triangulation T_*. The shape of the triangles looks unusual, because one vertex is at infinity, and some of them have two coinciding vertices (degenerate triangles). On the rigtht, we count the intersections of the edges of T_* with the curves of L_* and of $L_* \bullet \sigma_1$.

Definition 2.1 (T_*-coordinates).— Fix a numbering $e_1, ..., e_{3n+3}$ of the edges of T_*. Then the *T_*-coordinates* of a lamination is the length $3n+3$ sequence of natural numbers whose kth entry is the number of intersections between the curves of the lamination and the edge e_k.

This definition is not well posed. To obtain a number that is intrinsic and only depends on the homotopy type of the lamination, we must assume that the curves of the lamination are transversal to the edges of T_*, and there is no digon, which are those domains that arise when a curve has two adjacent intersections with the same edge. When this is done carefully, one obtains number that characterize the lamination up to homotopy.

Dynnikov's idea is to define coordinates for a braid β by comparing the T_*-coordinates of the laminations L_* and $L_* \bullet \beta$. For an induction, the problem is to express, for each lamination L (not only L_*), the T_*-coordinates of $L \bullet \sigma_i$ in terms of the T_*-coordinates of L. Here comes Dynnikov's trick.

Lemma 2.2.— *For each lamination L, the T_*-coordinates of $L \bullet \sigma_i$ are the $(T_* \bullet \sigma_i^{-1})$-coordinates of L.*

Proof. As σ_i is a homeomorphism, it is in particular a bijection, so, for every curve γ and every edge e of a tringulation T, the number of intersections of $\gamma \bullet \sigma_i$ and $e \bullet \sigma_i$ is the same as the number of intersections of γ and e. So the $(T_* \bullet \sigma_i^{-1})$-coordinates of L are the $((T_* \bullet \sigma_i^{-1}) \bullet \sigma_i)$-coordinates of $L \bullet \sigma_i$, *i.e.*, the T_*-coordinates of $L \bullet \sigma_i$. □

2.3. Flips. Hence the problem becomes that of comparing the coordinates of a given lamination L with respect to the two triangulations T_* and $T_* \bullet \sigma_i^{-1}$. This is easy. Indeed, it is known that one can always go from one triangulation to another one using a finite sequence of *flips*.

Definition 2.3 (flip).— Assume that T, T' are triangulations of some surface. One says that T' is obtained by *flipping* the edge e in T if e is the common edge of two triangles t_1, t_2 of T and T' is obtained by removing e and adding the other diagonal e' of the quadrilateral made by the triangles t_1 and t_2, see Figure 3).

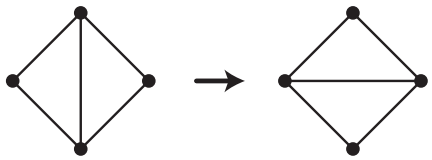


Figure 3. A flip: exchanging the diagonals in the quadrilateral made by two adjacent triangles.

Hence, on can certainly go from T_* to $T_* \bullet \sigma_i^{-1}$ by a finite sequence of flips.

Lemma 2.4.— *One goesfrom T_* to $T_* \bullet \sigma_i^{-1}$ using the four flips of Figure 4.*

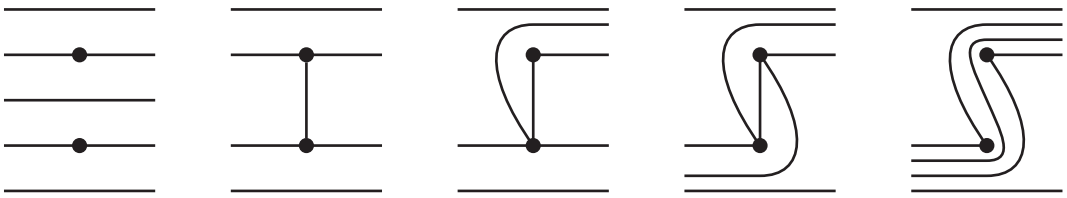


Figure 4. Going from T_* to $T_* \bullet \sigma_i^{-1}$ by a sequence of four flips; because of the point at infinity and of the degenerate triangles, it is not obvious at first that the steps are flips, but they are?

At this point, it only remains to investigate the action of one flip on the T_*-coordinates of a lamination. This is where the strange formulas involving tropical operations arise.

Lemma 2.5.— *Assume that C is a family of non-intersecting curves drawn on a triangulated surface, and $x_1, ..., x_4$, x, x' are the intersection numbers of C with the edges $e_1, ..., e_4, e, e'$ of Figure 5. Then one has*

$$x + x' = \max(x_1 + x_3, x_2 + x_4).$$

PROOF. Decompose the numbers x_k so as to count how many curves enter through the edge e_i and exit through the edge e_j. The point is that the curves are non-intersecting, so, if there is a curve entering through e_2 and exiting through e_4, so curve may enter through e_1 and exit e_3. □

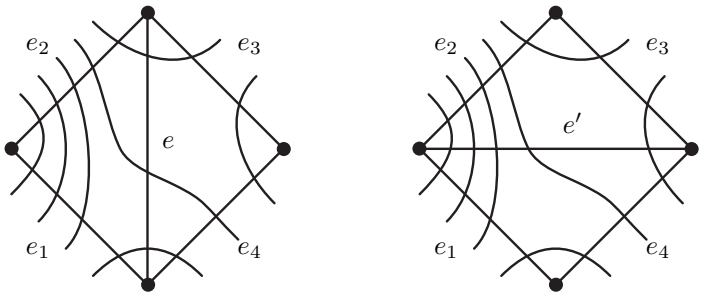


FIGURE 5. Relation between the intersection numbers of a family of non-intersecting curves with the edges of a triangulation when one flip is performed. Here we have $x_1 = 4$, $x_2 = 5$, $x_3 = 2$, $x_4 = 3$, $x = 3$, and $x' = 5$, and we find $3 + 5 = \max(4 + 2, 5 + 3)$.

Building on Lemma 2.5, one expresses the $(T_* \bullet \sigma_i^{-1})$-coordinates of L in terms of its T_*-coordinates. By Lemma 2.2, one deduces a formula expressing the $3n + 3$ T_*-coordinates of $L \bullet \sigma_i$ in terms of the T_*-coordinates of L. These are not yet the Dynnikov formulas, but almost. By choosing convenient differences, one defines from the $3n + 3$ coordinates $2n$ integers called the *reduced T_*-coordinates*, and, then, the formulas for the reduced coordinates are exactly those of Definition 1.1. We skip details.

2.4. Completeness of Dynnikov's formulas. We conclude with an argument showing that the Dynnikov coordinate do characterize the involved braid, *i.e.*, that Algorithm 1.4 actually solves the Braid Isotopy Problem.

Proposition 2.6.— *Every nontrivial braid has nontrivial Dynnikov coordinates.*

PROOF. As in the case of the Artin representation, we use as a black box the result that every nontrivial braid admits an expression by a braid word in which the generator σ_i with minimal index occurs only positively (no σ_i^{-1}) or negatively (no σ_i). So the point is to prove that, if w is a braid word that contains at least one σ_1 and no σ_1^{-1}, then the Dynnikov coordinates of w are not $(0, 1, ..., 0, 1)$. We prove the more precise result $a_1 > 0$.

Write w as $w_0\sigma_1 w_1\sigma_1 ...\sigma_1 w_p$ where there is no σ_1^{-1} in the words w_k. Let us follow the first coordinate a_1 when the successive letters of w are taken into account. By definition, we start with $a_1 = 1$. As long as we are inside w_0, the first two coordinates do not change, so they remain $0, 1$. When we pass the first letter σ_1, the coordinate a_1 changes according to a formula of the form

$$a_1 + (b_1 + c^+)^+ = 0 + (1 + c^+)^+ \geqslant 1,$$

so the new value is positive. Then the explicit formulas show that the value of a_1 cannot decrease when we apply further letters σ_1 or $\sigma_i^{\pm 1}$ with $i \geqslant 2$. □

CHAPTER VI

Handle reduction

We describe a new solution to the Braid Isotopy Problem called handle reduction. This method has a very simple combinatorial description, but it relies on the geometric properties of the so-called Cayley graph associated with the braid groups, and with some underlying order properties. The specificity of the method is that it is extremely efficient in practice—but, at the moment, there is no theoretical proof of that efficiency: the only proved upper bound for the complexity is exponential with respect to the length of the initial braid word, whereas extensive computer experiments suggest a much lower value.

1. The main result

For each n, the identity mapping on $\{\sigma_1, ..., \sigma_{n-1}\}$ induces an embedding of B_n into B_{n+1}, so that the groups B_n naturally arrange into an inductive system of groups, and the limit is denoted by B_∞: this is just the group generated by an infinite family $\sigma_1, \sigma_2, ...$ subject to the relations (II.3.10).

The elements of the group B_∞ are represented by words in the letters $\sigma_i^{\pm 1}$, which will be called *braid words*. In the sequel, we mainly deal with braid words (not braids). If w is a braid word, we denote by $\overline{w}$ the braid represented by w. Two braid words w, w' representing the same braid are called equivalent, written $w \equiv w'$. A braid word w of length ℓ is viewed as a length ℓ sequence of letters. For $1 \leqslant p \leqslant q \leqslant \ell$, the word obtained from w by deleting all letters before position p and after position q is called the (p, q)-subword of w. A prefix of w is a $(1, q)$-subword of q, *i.e.*, a subword that starts at the first letter of w.

Definition 1.1.— Assume that w is a nonempty braid word. We say that σ_m is the *main* letter of w if $\sigma_m^{\pm 1}$ occurs in w, but no $\sigma_i^{\pm 1}$ with $i > m$ does. We say that w is *σ-positive* (*resp.σ-negative*) if the main letter σ_m of w occurs only positively (*resp.*negatively) in w, *i.e.*, σ_m occurs in w but σ_m^{-1} does not.

Our aim is to prove

Proposition 1.2.— [**?, ?**] *Every braid word is equivalent to a word that is either empty, or σ-positive, or σ-negative.*

The proof given below relies on the following notion.

Definition 1.3.— We say that a braid word v is a *σ_i-handle* of *sign* $+$ (*resp.*$-$) if v is $\sigma_i u \sigma_i^{-1}$ (*resp.*$\sigma_i^{-1} u \sigma_i$) with u containing no letter $\sigma_j^{\pm 1}$ with $j \geqslant i$; we say that v is a *good* σ_i-handle if, in addition, at least one of the letters σ_{i-1}, σ_{i-1}^{-1} does not occur in u, *i.e.*, no subword of v is a σ_{i-1}-handle.

Thus Proposition 1.2 claims that every braid word w with main letter σ_m is equivalent to a braid word w' containing no σ_m-handle, this meaning that no subword of w' is a σ_m-handle.

A premilinary remark is that each word containing a handle contains a good handle.

Definition 1.4.— Let w be a braid word. We say that v is the *first handle in w* if v is a handle, there exist p, q such that v is the (p, q)-subword of w, and there exist no p', q' with $q' < q$ such that the (p', q')-subword of w is a handle.

Thus the first handle in a word w that contains a handle is the one that is first completed when one starts reading w from the left.

Lemma 1.5.— *Assume that w is a braid word containing at least one handle. Then the first handle in w is good.*

Proof. Let q be minimal such that the length q prefix w' of w contains a handle. By hypothesis, there exists p such that the (p, q)-subword of w is a handle, say $\sigma_i^e u \sigma_i^{-e}$, and, by construction, this handle is the first handle in w. We claim that this handle is good. Indeed, the contrary would mean that there

exist $p', q' < q$ such that the (p', q')-subword of w is a σ_{i-1}-handle, which implies that the length q' prefix of w contains a handle and contradicts the choice of q. □

Thus, in order to prove Proposition 1.2, it is sufficient to prove that every braid word is equivalent to a braid word that contains no good handle.

2. Handle reduction

Our task is to get rid of good handles. We do that using an iterative process, called handle reduction, that gets rid of the first handle and is repeated until no handle is left.

Definition 2.1.— (*i*) Assume that v is a good σ_i-handle, say $v = \sigma_i^e u \sigma_i^{-e}$. The *reduct* of v is defined to be the word obtained from u by replacing each letter σ_{i-1} with $\sigma_{i-1}^{-e}\sigma_i\sigma_{i-1}^e$, and each letter σ_{i-1}^{-1} with $\sigma_{i-1}^{-e}\sigma_i^{-1}\sigma_{i-1}^e$.

(*ii*) Assume that w is a braid word that contains at least one handle. Then $\mathrm{red}(w)$ denotes the word obtained from w by replacing the first handle by its reduct.

We write $\mathrm{red}^k(w)$ for $\mathrm{red}(\mathrm{red}(...(\mathrm{red}(w))...))$, red repeated k times, when the latter word exists; each word of the form $\mathrm{red}^k(w)$ is said to be obtained from w by *first handle reduction*.

Remark 2.2.— (*i*) One can introduce a similar reduction process for an arbitrary good handle, not necessarily the first one. All results established below extend to this general handle reduction. The only difference is that the latter is not deterministic in general, *i.e.*, there may be more than one way to reduce a given initial word.

(*ii*) Each braid word $\sigma_i\sigma_i^{-1}$ and $\sigma_i^{-1}\sigma_i$ is a good handle, and its reduct is the empty word ε. Thus handle reduction extends free group reduction.

The first, obvious result about handle reduction is:

Lemma 2.3.— *Each good handle is equivalent to its reduct.*

Proof. Make a picture. □

Hence, Proposition 1.2 follows from the convergence (or termination) of first handle reduction as stated in

Proposition 2.4.— *For every braid word w, there exists k such that* $\mathrm{red}^k(w)$ *contains no handle.*

Indeed, assume that w is a braid word and $\mathrm{red}^k(w)$ contains no handle. Then, by Lemma 1.5, the word $\mathrm{red}^k(w)$ is either empty, or σ-positive, or σ-negative, and, by Lemma 2.3, the words w and $\mathrm{red}^k(w)$ are equivalent.

Our task from now will be to prove Proposition 2.4, *i.e.*, to prove the convergence of first handle reduction. The proof relies on three auxiliary results, called Main Lemmas A, B, and C.

3. Main Lemma A

The key notion is the notion of a braid word *drawn in* some subset of the braid group.

Definition 3.1.— Assume $X \subseteq B_\infty$, and $a \in X$. We say that a braid word w is *drawn from a in X* if, for each prefix u of w, the braid $a\overline{u}$ belongs to X.

It is useful to think of X as the subgraph of the Cayley graph of the group B_∞ obtained by restricting the vertices to the elements of X and keeping those edges that connect two vertices in X. Then saying that w is drawn from a in X means that, starting from the vertex a, there exists inside X a path labeled by w. When X is the whole Cayley graph of B_∞, then every word is drawn from every vertex in X, but, when X is a proper subgraph, the condition of being drawn becomes nontrivial. Observe that, even if X is finite, arbitrary long words may be drawn in X: for instance, if X consists of 1 and σ_1, then, for every k, the word $(\sigma_1\sigma_1^{-1})^k$ is drawn from 1 in X.

As usual, B_∞^+ denotes the submonoid of B_∞ generated by the elements σ_i. An element of B_∞^+ is called a positive braid.

Definition 3.2.— If a, b are braids, we say that a is a *left divisor* of b, denoted $a \prec b$, if $b = ax$ holds for some x in B_∞^+. For b in B_∞^+, we denote by $\mathrm{Div}(b)$ the family of all left divisors of b in B_∞^+, *i.e.*, the set of all braids x satisfying $1 \prec x \prec b$.

Garside's theory shows that the relation $\prec$ is a partial ordering on B_∞ and that any two elements of B_∞ admit a lower bound (greatest common left divisor) and an upper bound (least common right multiple) with respect to $\prec$.

Main Lemma A.— For each braid word w, there exist two positive braids a, b such that every word of the form $\mathrm{red}^k(w)$ is drawn from a in $\mathrm{Div}(b)$.

Main Lemma A follows from two results:

Lemma 3.3.— *For each braid word w, there exist two positive braids a, b such that w is drawn from a in* $\mathrm{Div}(b)$.

Lemma 3.4.— *Assume that w is drawn from a in* $\mathrm{Div}(b)$. *Then so is* $\mathrm{red}(w)$, *when it exists.*

Proof of Lemma 3.3. Assume that w has length ℓ and main letter σ_m. For $p \leqslant \ell$, let w_p be the length p prefix of w. Garside's theory implies that, for each p, there exist integers $d_p, e_p \geqslant 0$ satisfying $1 \prec \Delta_{m+1}^{d_p} \overline{w_p} \prec \Delta_{m+1}^{d_p+e_p}$. Let $d := \max\{d_1, \cdots, d_p\}$ and $e := \max\{e_1, \cdots, e_p\}$. Then, for each p, we have $1 \prec \Delta_{m+1}^{d} \overline{w_p} \prec \Delta_{m+1}^{d+e}$, which means that w is drawn from Δ_{m+1}^d in $\mathrm{Div}(\Delta_{m+1}^{d+e})$. □

The proof of Lemma 3.4 consists in decomposing handle reduction into more elementary transformations and showing that the words drawn from a in $\mathrm{Div}(b)$ are closed under these elementary transformations.

Definition 3.5.— Let w, w' be braid words. We say that w' is obtained from w by a *type 1, 2, 3, or 4 transformation* if w' is obtained from w by replacing some subword of the following type by the associated one:

- type 1: $\sigma_i\sigma_j \mapsto \sigma_j\sigma_i$ with $|i-j| \geqslant 2$;
- type 2: $\sigma_i^{-1}\sigma_j^{-1} \mapsto \sigma_j^{-1}\sigma_i^{-1}$ with $|i-j| \geqslant 2$;
- type 3: $\sigma_i^{-1}\sigma_j \mapsto \sigma_j\sigma_i^{-1}$ with $|i-j| \geqslant 2$,
  or $\sigma_i^{-1}\sigma_j \mapsto \sigma_j\sigma_i\sigma_j^{-1}\sigma_i^{-1}$ with $|i-j| = 1$,
  or $\sigma_i^{-1}\sigma_i \mapsto \varepsilon$;
- type 4: $\sigma_i\sigma_j^{-1} \mapsto \sigma_j^{-1}\sigma_i$ with $|i-j| \geqslant 2$,
  or $\sigma_i\sigma_j^{-1} \mapsto \sigma_j^{-1}\sigma_i^{-1}\sigma_j\sigma_i$ with $|i-j| = 1$,
  or $\sigma_i\sigma_i^{-1} \mapsto \varepsilon$.

Then Lemma 3.4 follows from the next two results:

Lemma 3.6.— *From each braid word w such that* $\mathrm{red}(w)$ *exists, one can go from w to* $\mathrm{red}(w)$ *by a finite sequence of type 1–4 transformations.*

Lemma 3.7.— *Assume that w is drawn from a in* $\mathrm{Div}(b)$, *and w' is obtained from w by a transformation of type 1–4. Then w' is drawn from a in* $\mathrm{Div}(b)$.

Proof of Lemma 3.6. The point is to prove that, if v is a good handle, and v' is its reduct, then one can go from v to v' by composing types 1–4 transformations. By definition, there exist exponents $e, d = \pm 1$ such that v has the form

$$v = \sigma_i^e \quad u_0 \quad \sigma_{i-1}^d \quad u_1 \quad \cdots \quad u_{r-1} \quad \sigma_{i-1}^d \quad u_r \quad \sigma_i^{-e}, \tag{3.8}$$

where $u_0, \cdots, u_r$ contain only letters $\sigma_j^{\pm 1}$ with $j \leqslant i-2$, and we have then

$$v' = \quad u_0 \quad \sigma_{i-1}^{-e}\sigma_i^d\sigma_{i-1}^e \quad u_1 \quad \cdots \quad u_{r-1} \quad \sigma_{i-1}^{-e}\sigma_i^d\sigma_{i-1}^e \quad u_r. \tag{3.9}$$

Assume first $d = 1$, $e = -1$. The involved words are

$$\begin{array}{lccccccccc} v = & \sigma_i^{-1} & u_0 & \sigma_{i-1} & u_1 & \cdots & u_{r-1} & \sigma_{i-1} & u_r & \sigma_i, \\ v' = & & u_0 & \sigma_{i-1}\sigma_i\sigma_{i-1}^{-1} & u_1 & \cdots & u_{r-1} & \sigma_{i-1}\sigma_i\sigma_{i-1}^{-1} & u_r & . \end{array}$$

The principle is to use type 2 and 3 transformations to let the initial letter σ_i^{-1} in v migrate to the right until it reaches to the final letter σ_i. First, σ_i^{-1} crosses u_0 using type 3 transformations for the positive letters in u_0, and type 2 transformations for the negative ones. In this way, we reach the word

$$u_0 \quad \sigma_i^{-1}\sigma_{i-1} \quad u_1 \quad \cdots \quad u_{r-1} \quad \sigma_{i-1} \quad u_r \quad \sigma_i.$$

One more type 3 transformation lets σ_i^{-1} cross σ_{i-1}, resulting in the word

$$u_0 \quad \sigma_{i-1}\sigma_i\sigma_{i-1}^{-1}\sigma_i^{-1} \quad u_1 \quad \cdots \quad u_{r-1} \quad \sigma_{i-1} \quad u_r \quad \sigma_i.$$

The same process lets σ_i^{-1} cross u_1, and the next σ_{i-1}, and, after r such steps, we reach the word

$$u_0 \quad \sigma_{i-1}\sigma_i\sigma_{i-1}^{-1} \quad u_1 \quad \cdots \quad u_{r-1} \quad \sigma_{i-1}\sigma_i\sigma_{i-1}^{-1} \quad u_r \quad \sigma_i^{-1}\sigma_i,$$

and a final type 3 transformation leads to the expected word v'.

The argument for the case $d = -1$, $e = +1$ is similar, with transformations of type 1 and 4 instead of 2 and 3.

For the case $d = 1$, $e = 1$, the argument is symmetric, *i.e.*, we start with the final letter σ_i^{-1} and let it migrate to the left, using transformations of type 2 and 4.

Finally, the case $d = e = -1$ is similar, with transformations of type 1 and 3 instead of 2 and 4. □

Proof of Lemma 3.7. We assume that w is drawn from a in $\mathrm{Div}(b)$, and that w' is obtained from w by one type 1 transformation. This means that there exist words w_1, w_2 and letters σ_i, σ_j with $|i-j| \geqslant 2$ satisfying

$$w = w_1\ \sigma_i\sigma_j\ w_2 \quad \text{and} \quad w' = w_1\ \sigma_j\sigma_i\ w_2.$$

Our task is to show that, for every prefix u of w', the braid $a\overline{u}$ belongs to $\mathrm{Div}(b)$. By construction, all prefixes of w' are prefixes of w, except $u_1 = w_1\sigma_j$. The question is to show $1 \prec a\overline{u_1} \prec b$. Let $c = a\overline{w_1}$ and $d = a\overline{w_1}\sigma_i\sigma_j$. By construction, we have $c \prec a\overline{u_1} \prec d$, and it is sufficient to show $1 \prec c$ and $d \prec b$. Now the latter relations directly follow from the hypothesis that w is drawn from a in $\mathrm{Div}(b)$, as w_1 and $w_1\sigma_i\sigma_j$ are prefixes of w. So w' is drawn from a in $\mathrm{Div}(b)$.

Consider now a type 2 transformation. By definition, we have

$$w = w_1\ \sigma_i^{-1}\sigma_j^{-1}\ w_2 \quad \text{and} \quad w' = w_1\ \sigma_j^{-1}\sigma_i^{-1}\ w_2,$$

again with $|i-j| \geqslant 2$. The only prefix of w' that is not a prefix of w is $u_1 = w_1\sigma_j^{-1}$. Let $c = a\overline{w_1}\sigma_i^{-1}\sigma_j^{-1}$, and $d = a\overline{w_1}$. By construction, we have $c \prec a\overline{u_1} \prec d$, and, once again, it is sufficient to show $1 \prec c$ and $d \prec b$. The latter relations follow from the hypothesis that w is drawn from a in $\mathrm{Div}(b)$, as $w_1\sigma_i^{-1}\sigma_j^{-1}$ and w_1 are prefixes of w. So w' is drawn from a in $\mathrm{Div}(b)$.

We turn to type 3, and consider the case

$$w = w_1\ \sigma_i^{-1}\sigma_j\ w_2 \quad \text{and} \quad w' = w_1\ \sigma_j\sigma_i\sigma_j^{-1}\sigma_i^{-1}\ w_2$$

with $|i-j| = 1$. The other two cases, namely $|i-j| \geqslant 2$ and $i = j$, are similar and easier. Three prefixes of w' are not prefixes of w, namely $u_1 = w_1\sigma_j$, $u_2 = w_1\sigma_j\sigma_i$, and $u_3 = w_1\sigma_j\sigma_i\sigma_j^{-1}$. Let $c = a\overline{w_1}\sigma_i^{-1}$, and $d = a\overline{w_1}\sigma_j\sigma_i$. By construction, we have $c \prec a\overline{u_k} \prec d$ for $k = 1, 2, 3$, and, here again, it suffices to prove $1 \prec c$ and $d \prec b$. Now $1 \prec c$ follows from the hypothesis that w is drawn from a in $\mathrm{Div}(b)$, as $w_1\sigma_i^{-1}$ is a prefix of w. On the other hand, the hypothesis that both $c\sigma_i$ and $c\sigma_j$ are left divisors of b implies that their least common multiple, which is d, is also a divisor of b. So w' is drawn from a in $\mathrm{Div}(b)$.

Finally, consider type 4. We consider the case of

$$w = w_1\ \sigma_i\sigma_j^{-1}\ w_2 \quad \text{and} \quad w' = w_1\ \sigma_j^{-1}\sigma_i^{-1}\sigma_j\sigma_i\ w_2$$

with $|i-j| = 1$. Three prefixes of w' fail to be prefixes of w, namely $u_1 = w_1\sigma_j^{-1}$, $u_2 = w_1\sigma_j^{-1}\sigma_i^{-1}$, and $u_3 = w_1\sigma_j^{-1}\sigma_i^{-1}\sigma_j$. Let $c = a\overline{w_1}\sigma_j^{-1}\sigma_i^{-1}$, and $d = a\overline{w_1}\sigma_i$. By construction, we have $c \prec a\overline{u_k} \prec d$ for $k = 1, 2, 3$. So the point again is to check the relations $1 \prec c$ and $d \prec b$. The latter directly follows from the hypothesis that w is drawn from a in $\mathrm{Div}(b)$ since $w_1\sigma_i$ is a prefix of w. On the other hand, w_1 and $w_1\sigma_i\sigma_j^{-1}$ are prefixes of w, hence the hypothesis that w is drawn from a in $\mathrm{Div}(b)$ implies that 1 is a left divisor both of $d\sigma_i^{-1}$ and $d\sigma_j^{-1}$, hence it is left divisor of their greatest common left divisor, which is c. Once again, w' is drawn from a in $\mathrm{Div}(b)$. □

Thus the proof of Main Lemma A is complete.

4. Main Lemma B

Main Lemma B enables one to convert the geometric boundedness result of Main Lemma A (all words obtained by handle reduction remain drawn in some finite subset of the braid monoid B_∞^+) into an actual finiteness result.

Main Lemma B.— A σ-positive word is not equivalent to the empty word.

Proof. The result has been proved in Chapter IV, in the course of establishing Theorem IV.**??**. Indeed, what we proved is that, if a braid word w contains at least one letter σ_1 and no letter σ_1^{-1}, then the associated Artin automorphism is not the identity and, therefore, the word w cannot represent the unit braid. As mentioned above, using the flip aitomrphism Φ_n, we deduce that, symmetrically, the Artin automorphism associated with a σ-positive word is not the identity, hence that a σ-positive word cannot be equivalent to the empty word. □

Corollary 4.1.— *Assume that a, b are positive braids and w is a σ-positive braid word drawn from a in* Div(b). *Then the number of occurrences of the main letter of w is at most the cardinality of* Div(b).

Proof. Assume that the main letter σ_m of w occurs r times in w. Let $u_1, \dots, u_r$ be the prefixes of w such that u_j finishes just before the jth letter σ_m in w. By hypothesis, all braids $a\overline{u_j}$ belong to Div(b). Now $j < j'$ implies $a\overline{u_j} \neq a\overline{u_{j'}}$: indeed, by construction, we have $u_{j'} = u_j v$, where v contains at least one letter σ_m, and no letter σ_m^{-1}, so, by Main Lemma B, the braid $\overline{v}$ is not 1. Hence $a\overline{u_1}, \dots, a\overline{u_r}$ are pairwise distincts elements of Div(b), and, therefore, we have $r \leqslant \mathrm{card}(\mathrm{Div}(b))$. □

5. Main Lemma C

The last ingredient is a monotonicity result actually showing that some parameter either always increases or always decreases when first handle reductions are performed. Here we give the argument without mentioning the order phenomenon explicitly.

Definition 5.1.— Assume that w is a braid word with main letter σ_i. We denote by $h(w)$ the number of σ_i-handles in w, and, assuming $h(w) \geqslant 1$, we denote by $e(w)$ the sign of the first σ_i-handle in w and by $P(w)$ the prefix of w that finishes with the first letter of the first σ_i-handle of w.

Main Lemma C.— Assume that w is a braid word drawn from a in Div(b) containing at least one handle, that the main letter of w is σ_m and that the first handle in w is a σ_i-handle. Let w' be obtained from w by reducing the first handle of w. Then three cases are possible:

Case 1: $h(w') = h(w) = 0$;
Case 2: $h(w') < h(w)$;
Case 3: $h(w') = h(w) \geqslant 1$.

Moreover, in Case 3, we have $e(w') = e(w)$, and there exists a word $\gamma(w)$ satisfying
(a) the word $\gamma(w)$ is drawn from $a\overline{P(w)}$ in Div(b),
(b) we have $P(w') \equiv P(w)\gamma(w)$,
(c) if $i < m$ holds, then $\gamma(w)$ is empty,
(d) if $i = m$ holds, then $\gamma(w)$ contains one letter $\sigma_i^{-e(w)}$ and no letter $\sigma_i^{e(w)}$.

Proof. Let w^* be the word obtained from w by deleting all letters $\sigma_i^{\pm 1}$ with $i < m$. Then w^* consists of an alternating sequence of blocks of σ_m and σ_m^{-1}. We define the *profile* $\Pi(w)$ of w to be the finite sequence made by the sizes of these blocks. For instance, for $w = \sigma_2\sigma_1\sigma_2\sigma_1^{-1}\sigma_2^{-1}\sigma_2\sigma_2\sigma_2\sigma_1$, the main letter of w is σ_2, we have $w^* = \sigma_2\sigma_2\sigma_2^{-1}\sigma_2\sigma_2\sigma_2$, and $\Pi(w) = (2, 1, 3)$ as w^* consists of two σ_2's, followed by one σ_2^{-1}, followed by three σ_2's. The σ_m-handles in w correspond to the sign alternations in the exponents of the letters σ_m and, therefore, $\Pi(w)$ is a sequence of length $h(w) + 1$.

If w contains no σ_m-handle, *i.e.*, if $\Pi(w)$ is a length 1 sequence, then one goes from w to w' by reducing some σ_i-handle wit $i < m$, and w' contains no σ_m-handle either. So we are in Case 1.

From now on, we assume $h(w) \geqslant 1$. Then $\Pi(w)$ is some sequence $(r, s, \dots)$ of length $\geqslant 2$, and the generic form of w is

$$w = v_0\ \sigma_m e\ v_1\ \sigma_m e\ \cdots\ v_{r-2}\ \sigma_m e\ v_{r-1}\ \underline{\sigma_m e\ v_r\ \sigma_m {-e}}\ v_{r+1}\ \sigma_m {-e}\ \cdots, \tag{5.2}$$

where the v words contain no $\sigma_m^{\pm 1}$ and the underlined subword is the first σ_m-handle in w. With this notation, we have $P(w) = v_0\sigma_m e \cdots v_{r-1}\sigma_m e$.

Assume first $i < m$, *i.e.*, the first handle in w is not the underlined σ_m-handle. Then the reduction from w to w' occurs inside one of the words $v_0, \dots, v_r$, *i.e.*, it consists in replacing some subword v_j with the corresponding word $\mathrm{red}(v_j)$. In this case, we have $\Pi(w') = \Pi(w)$, and, therefore, $h(w') = h(w)$ and $e(w') = e(w)$. Moreover, $P(w')$ is either equal to $P(w)$ (case $j = r$), or obtained from $P(w)$ by replacing the subword v_j with $\mathrm{red}(v_j)$ (case $j < r$). In all cases, $P(w') \equiv P(w)$ holds, and all requirements of Case 3 are fulfilled with $\gamma(w) = \varepsilon$ (the empty word).

Assume now $i = m$, *i.e.*, w' is obtained from w by reducing the underlined σ_m-handle of (5.2). We compare the profiles of w' and w according to the letters $\sigma_{m-1}^{\pm1}$ possibly occurring in v_r. The hypothesis that the word $\sigma_m e v_r \sigma_m{-}e$ is a good handle implies that σ_{m-1} and σ_{m-1}^{-1} do not simultaneously occur in v_r, and, therefore, the latter can be written as

$$u_0\ \sigma_{m-1}d\ u_2\ \sigma_{m-1}d\ \cdots\ u_{t-1}\ \sigma_{m-1}d\ u_t$$

for some $t \geqslant 0$, $d = \pm1$, and the u words containing no $\sigma_m^{\pm1}$ or $\sigma_{m-1}^{\pm1}$.

Assume first $t = 0$, *i.e.*, v_r contains no $\sigma_{m-1}^{\pm1}$. Then the reduct of $\sigma_m e v_r \sigma_m{-}e$ is v_r, so here reduction amounts to deleting the underlined letters $\sigma_m e$ and $\sigma_m{-}e$ of (5.2). Hence, $\Pi(w')$ is the sequence obtained from $(r-1, s-1, \cdots)$ by possibly regrouping entries if some zero value appears. Therefore, in all cases, we have $h(w') \leqslant h(w)$, and equality holds if and only if we have $r \geqslant 2$ and $s \geqslant 2$. The latter case corresponds to

$$w' = v_0\ \sigma_m e\ v_1\ \sigma_m e\ \cdots\ v_{r-2}\ \underline{\sigma_m e\ v_{r-1}\quad v_r\quad v_{r+1}\ \sigma_m{-}e}\ \cdots, \tag{5.3}$$

in which the new first σ_m-handle is underlined. We read on (5.3) the relations $e(w') = e(w) = e$ and $P(w') = v_0\sigma_m e \cdots v_{r-2}\sigma_m e$, and, therefore,

$$P(w) = P(w')v_{r-1}\sigma_m e.$$

We deduce $P(w') \equiv P(w)\ \sigma_m{-}ev_{r-1}^{-1}$, which gives the expected properties for $\gamma(w) = \sigma_m{-}ev_{r-1}^{-1}$, as, by construction, the word $\gamma(w)$ is drawn from $a\overline{P(w)}$ in $\mathrm{Div}(b)$ since $v_{r-1}\sigma_m e$ is a suffix of $P(w)$, which by hypothesis is drawn from a in $\mathrm{Div}(b)$.

Assume now $t \geqslant 1$ with $d = -e$, *i.e.*, the letter $\sigma_{m-1}{-}e$ occurs in the handle v_r. Then each letter $\sigma_{m-1}{-}e$ in v_r gives rise to a letter σ_m^{-e} in the reduct of v_r, hence in w'. Hence $\Pi(w')$ is the sequence obtained from $(r-1, s-1+t, \cdots)$ by possibly regrouping entries if some zero value appears. Therefore, in all cases, we have $h(w') \leqslant h(w)$, and equality holds if and only if we have $r \geqslant 2$. The latter case corresponds to

$$w' = v_0\ \sigma_m e\ v_1\ \sigma_m e\ \cdots\ v_{r-2}\ \underline{\sigma_m e\ v_{r-1}\quad u_0\ \sigma_{m-1}{-}e\sigma_m{-}e\sigma_{m-1}e\ u_1}\ \cdots, \tag{5.4}$$

in which the new first σ_m-handle is underlined. We read on (5.4) the relations $e(w') = e(w) = e$ and $P(w') = v_0\sigma_m e \cdots v_{r-2}\sigma_m e$, hence $P(w) = P(w')v_{r-1}\sigma_m e$ as above, and we conclude exactly as in the previous case.

Finally, assume $t \geqslant 1$ with $d = e$, *i.e.*, the letter $\sigma_{m-1}e$ occurs in the handle v_r. Each letter $\sigma_{m-1}e$ in v_r gives rise to a letter σ_m^{-e} in the reduct of v_r, hence in w'. It follows that the profile of w' is the sequence obtained from $(r-1+t, s-1, \cdots)$ by possibly regrouping entries if some zero value appears. Therefore, in all cases, we have $h(w') \leqslant h(w)$, and equality holds if and only if we have $s \geqslant 2$. Writing v for $v_0\sigma_m e \cdots v_{r-1}$, the latter case corresponds to

$$w' = v\ u_0\ \sigma_{m-1}{-}e\sigma_m e\sigma_{m-1}e\ u_1\ \cdots\ u_{t-1}\ \sigma_{m-1}{-}e\underline{\sigma_m e\sigma_{m-1}e\ u_t\quad v_{r+1}\ \sigma_m{-}e}\ \cdots \tag{5.5}$$

in which the new first σ_m-handle is underlined. We read on (5.5) the relation $e(w') = e(w) = e$. Moreover, with our notations, we have $P(w) = v\sigma_m e$, and (5.5) gives

$$P(w)v_r\sigma_m{-}e \equiv P(w')\sigma_{m-1}eu_t.$$

We deduce $P(w') \equiv P(w)\ v_r\sigma_m{-}eu_t^{-1}\sigma_{m-1}{-}e$, which gives the expected properties for $\gamma(w) = v_r\sigma_m{-}eu_t^{-1}\sigma_{m-1}{-}e$, as the word $\gamma(w)$ is drawn from $a\overline{P(w)}$ in $\mathrm{Div}(b)$. Indeed, w is drawn from a in $\mathrm{Div}(b)$ by hypothesis and $P(w)v_r\sigma_m{-}e$ is a prefix of w, hence $v_r\sigma_m{-}e$ is drawn from $a\overline{P(w)}$ in $\mathrm{Div}(b)$; on the other hand, by Main Lemma A, w' is drawn from a in $\mathrm{Div}(b)$ too, and $P(w')\sigma_{m-1}eu_t$ is a prefix of w', hence $u_t^{-1}\sigma_{m-1}{-}e$ is drawn from $a\overline{P(w')}\sigma_{m-1}e\overline{u_t}$, which is also $a\overline{P(w)v_r\sigma_m}{-}e$, in $\mathrm{Div}(b)$. So $\gamma(w)$ is drawn from $a\overline{P(w)}$ in $\mathrm{Div}(b)$, and the proof is complete. □

We are now ready to conclude, *i.e.*, to prove Proposition 2.4.

Proof of Proposition 2.4. We prove the following result using induction on $m \geqslant 1$:

> For every braid word w with main letter σ_m, there exists k such that $\mathrm{red}^k(w)$ contains no handle (and therefore $\mathrm{red}^{k+1}(w)$ does not exist).

For $m = 1$, the only possible letters in w are σ_1 and σ_1^{-1}, handle reduction is a free group reduction, and the result is clear, with k at most the half of the length of w.

Assume $m \geqslant 2$, and assume for a contradiction that w is a braid word with main letter σ_m such that $\mathrm{red}^k(w)$ exists for every k. We write w_k for $\mathrm{red}^k(w)$.

By Main Lemma C, the numbers $h(w_k)$ make a nonincreasing sequence, hence the latter must be eventually constant. So, at the expense of possibly deleting the first w_k's, we can assume that there exists h such that $h(w_k) = h$ holds for every k.

By hypothesis, w_{k+1} is obtained from w_k by reducing its first handle, which is either a σ_m-handle, or a σ_i-handle for some $i < m$. Let K be the set of all k's such that the first handle in w_k is a σ_m-handle.

Firstly, we claim that K is infinite. Indeed, let k be any nonnegative integer. Then we can write

$$w_k = v_0\ \sigma_m e\ v_1\ \sigma_m e\ v_2\ \cdots\ v_{r-1}\ \sigma_m e\ v_r\ v$$

where v either begins with $\sigma_m{-}e$ (case $h > 0$) or is empty (case $h = 0$). By construction, the main letter of each of the words v_j is $\sigma_{m'}$ with $m' < m$. Hence, by induction hypothesis, there exists for each j an integer k_j such that $\mathrm{red}^{k_j}(v_j)$ contains no handle. Let $k' = k + k_0 + \cdots + k_r$. Then, by construction, we have

$$w_{k'} = \mathrm{red}^{k_0}(v_0)\ \sigma_m e\ \mathrm{red}^{k_1}(v_1)\ \sigma_m e\ v_2\ \cdots\ \mathrm{red}^{k_{r-1}}(v_{r-1})\ \sigma_m e\ \mathrm{red}^{k_r}(v_r)\ v.$$

If v were empty, $w_{k'}$ would contain no handle, contradicting our hypothesis that the sequence $(w_k)_{k\geqslant 0}$ is infinite. Hence v begins with $\sigma_m{-}e$, and the first handle in $w_{k'}$ is a σ_m-handle. Thus we found an element k' of K which is $\geqslant k$, and K is infinite.

On the other hand, we claim that K is finite, thus getting the expected contradiction. Indeed, let a, b be positive braids such that w, hence, by Main Lemma A, all words w_k are drawn from a in $\mathrm{Div}(b)$. We apply Main Lemma C to w_k. By hypothesis, we always are in Case 3. Let e be the common value of $e(w_k)$ for all k, and let γ be the (infinite) word $\gamma(w_0)\gamma(w_1)\ldots$. By construction, the word γ is drawn from $a\overline{P(w)}$ in $\mathrm{Div}(b)$, it contains no letter $\sigma_m e$, and it contains exactly one letter $\sigma_m{-}e$ for each k in K. By Main Lemma B, the number of such letters, and therefore the cardinal of K, is at most the cardinal of $\mathrm{Div}(b)$. In particular K is finite.

Hence the existence of a word w with main letter σ_m such that $\mathrm{red}^k(w)$ exists for every k is a contradictory assumption, and the proof is complete. □

CHAPTER VII

The greedy normal form

We now consider a solution to the Braid Isotopy Problem of a completely if and only iferent type, namely a solution based on a normal form approach: a braid is an equivalence class of braid words and we shall choose in each equivalence class a distinguished word called normal. Then, by construction, a braid word represents the trivial braid if and only if it is *equal* to the unique normal word that represents the trivial braid (usually the empty word, but this is not necessary). Equivalently, two braid words w, w' are equivalent if and only if the (unique) normal words $\widetilde{w}, \widetilde{w'}$ that are equivalent to w and w' coincide. Of course, this gives an algorithmic solution only if the procedure that chooses the distinguished element in each equivalence class is itself effective.

In the case of braid groups, several normal forms have been constructed. Here we consider one that relies on Garside's results of Chapter III and was discovered by several researchers about at the same time (Adjan, Thurston, ElRifai and Morton), called the *greedy normal form*. Its specific interest is that it enjoys many nice combinatorial properties, in particular those involved for a bi-automatic structure.

1. Summary of previous results

1.1. Braid groups. We recall that the n-strand braid group B_n is defined for $n \geqslant 1$ by the presentation

$$B_n = \left\langle \sigma_1, ..., \sigma_{n-1} \;\middle|\; \begin{array}{ll} \sigma_i\sigma_j = \sigma_j\sigma_i & \text{for } |i-j| \geqslant 2 \\ \sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j & \text{for } |i-j| = 1 \end{array} \right\rangle. \tag{1.1}$$

So, B_1 is a trivial group $\{1\}$, while B_2 is the free group generated by σ_1. The elements of B_n are called n strand braids, or simply *n-braids*. We use B_∞ for the group generated by an infinite sequence of σ_i's subject to the relations of (1.1), *i.e.*, the direct limit of all B_n's under the inclusion of B_n into B_{n+1}.

By definition, every n-braid x admits (infinitely many) expressions in terms of the generators σ_i and their inverses. Such a expression is called an n strand *braid word*. Two braid words w, w' representing the same braid are said to be *equivalent*, denoted $w \equiv w'$; the braid represented by a braid word w is denoted $\overline{w}$. By definition, two words w, w' are equivalent if and only if one can go from w to w' by a finite sequence of elementary transformations of the following types:

- replacing a subword $\sigma_i\sigma_j$ with $|i-j| \geqslant 2$ with the corresponding word $\sigma_j\sigma_i$,
- replacing a subword $\sigma_i\sigma_j\sigma_i$ with $|i-j| = 1$ with the corresponding word $\sigma_j\sigma_i\sigma_j$,
- deleting a subword $\sigma_i\sigma_i^{-1}$ or $\sigma_i^{-1}\sigma_i$,
- inserting a word of the form $\sigma_i\sigma_i^{-1}$ or $\sigma_i^{-1}\sigma_i$.

By definition, *solving the Word Problem* for the group presentation (1.1)—one sometimes simply say "solving the Word Problem for B_n"—means describing an algorithm that, given an arbitrary braid word w, decides whether w represents 1 in the group B_n.

1.2. Braid diagrams. One associates with every n strand braid word w an n strand braid *diagram* by stacking elementary diagrams associated with the successive letters according to the rules

Then two braid words are equivalent if and only if the diagrams they encode are the projections of ambient isotopic figures in R^3, *i.e.*, one can deform one diagram into the other without allowing the strands to cross or moving the endpoints (see details in Chapter II).

It follows that the Word Problem for the presentation (1.1) of B_n is equivalent to solving the n-strand Braid Isotopy Problem, *i.e.*, the problem of recognizing whether a given n-strand braid diagram is isotopic to the trivial diagram (the one with no crossing) or not.

1.3. Braid monoids. For each n, one introduces the monoid[1] B_n^+ that admits, as a monoid, the presentation (1.1). The elements of B_n^+ are called *positive* n-braids.

By construction, the elements of B_n^+ are represented by braid words that contain no letter σ_i^{-1}: such words are called *positive braid words*. Two positive braid words w, w' represent the same element of the monoid B_n^+, denoted $w \equiv^+ w'$, if and only if one can go from w to w' by a finite sequence of elementary transformations of the following types:

- replacing a subword $\sigma_i\sigma_j$ with $|i-j| \geqslant 2$ with the corresponding word $\sigma_j\sigma_i$,
- replacing a subword $\sigma_i\sigma_j\sigma_i$ with $|i-j| = 1$ with the corresponding word $\sigma_j\sigma_i\sigma_j$,

i.e., the same transformations as for braid word equivalence, but without using the inverses of the generators σ_i. It is clear that, if w, w' are positive braid words, $w \equiv^+ w'$ implies $w \equiv w'$, but the contrary is not clear at all: one might be able to go from w to w' by introducing some auxoliary pairs $\sigma_i\sigma_i^{-1}$ or $\sigma_i^{-1}\sigma_i$ that subsequently vanish but be unable to do it without introducing such pairs. This does *not* happen: we proved in Chapter III (Corollary III.2.24)

Proposition 1.2.— *The relation $\equiv^+$ is the restriction of the relation $\equiv$ to positive braid words: for all positive braid words w, w' one has*

$$w \equiv^+ w' \quad \Leftrightarrow \quad w \equiv w'. \tag{1.3}$$

The proof is delicate and requires long developments.

By contrast, a trivial but useful fact is that, because the relations in (1.1) have the property that, in each case, both sides are words with the same length, then $w \equiv^+ w'$ always implies that the length of the word w is equal to the length of the word w', denoted $|w| = |w'|$. It follows that the length function induces a well defined function on the braid monoid B_n^+: for x a positive braid, we denote by $|x|$ the length of every positive braid word that represents x, and call it the *length* of x. The following results are then easy.

Proposition 1.4.— (*i*) *The length function is a homomorphism of B_n^+ to $\mathbb{N}$ that takes σ_i to 1 for each i.*

(*ii*) *The trivial braid 1 is the only positive braid with length zero.*

(*iii*) *For each n, and for each number ℓ, there exists only finitely many positive braids x in B_n^+ satisfying $|x| \leqslant \ell$.*

Proof. Point (*i*) directly follows from the definition. For (*ii*), the empty word is the only word with length zero. As for (*iii*), such a positive braid has to be represented by at least one word of length at most ℓ on the finite alphabet $\{\sigma_1, ..., \sigma_{n-1}\}$, and the number of such words is at most $1 + (n-1) + (n-1)^2 + \cdots + (n-1)^\ell$. □

2. The lattice structure of B_n^+

2.1. The left-divisibility relation on positive braids. First we recall the notion of left- and right-divisiblity in the monoid B_n^+.

Definition 2.1 (left-divisor, right-multiple).— For x, y in B_n^+, we say that x is a *left-divisor* of y, denoted $x \preccurlyeq y$, or, equivalently, that y is a *right multiple* of x, if $y = xz$ holds for some z in B_n^+. We denote by $\mathrm{Div}(y)$ the (finite) set of all left-divisors of y in B_n^+.

Note that x is a (left) divisor of y in the sense of B_n^+ if and only if it is a (left) divisor in the sense of B_∞^+, so there is no need to specify the index n.

Lemma 2.2.— *For each n, the left-divisibility relation is an ordering*[2].

Proof. The only point that is not completely obvious (check the others!) is antisymmetry. Assume $x \preccurlyeq y$ and $y \preccurlyeq x$. By Proposition 1.4, $y = xy'$ implies $|y| = |x| + |y'|$. So, if we have $x \preccurlyeq y$, say $y = xy'$, and $|x| = |y|$, then we have $|y'| = 0$, whence $y' = 1$, and $y = x$. □

[1] a *monoid* is an algebraic structure consisting of a set equipped of a binary operation that is associative and admits a neutral element; so a group is a special type of monoid in which, in addition, all elements admit an inverse; in a general monoid, inverses need not exist.

[2] *i.e.*, a binary relation that is reflexive ($x \preccurlyeq x$ always holds), antisymmetric (the conjunction of $x \preccurlyeq y$ and $y \preccurlyeq x$ implies $x = y$), transitive (the conjunction of $x \preccurlyeq y$ and $y \preccurlyeq z$ implies $x \preccurlyeq z$)

2.2. Lcm's and gcd's. In Proposition III.2.1, we have seen that any two elements of the monoid B_n^+ admit a common right-multiple. The (easy) proof is based on the fact that every element of B_n^+ that can be expressed as a product of at most k letters σ_i is a left-divisor of Δ_n^k, where Δ_n is is inductively defined by

$$\Delta_1 = 1, \qquad \Delta_n = \sigma_1\sigma_2...\sigma_{n-1}\,\Delta_{n-1}. \tag{2.3}$$

Then we have seen that every positive braid x of B_n^+ with length at most ℓ is a left-divisor of Δ_n^k, *i.e.*, in other words, Δ_n^ℓ is a common right-multiple of all positive braids of B_n^+ with length at most ℓ.

Using the reversing technique of Section III.2, we can establish a more precise result, namely that any two elements of B_n^+ admit a *least* common right-multiple. In the sequel we shall use W^+ for the set of all positive braid words. The only result we need to know here is that there exists a function C from $W^+ \times W^+$ to W^+ such that, for all braid words u, v, u', v', the following relations hold

$$u\,C(u,v) \equiv^+ v\,C(v,u), \tag{2.4}$$

$$uv' \equiv vu' \implies \exists w(u' \equiv^+ C(v,u)\,w \text{ and } v' \equiv^+ C(u,v)\,w). \tag{2.5}$$

Definition 2.6 (least common right-multiple or **right-lcm).**— Assume that x, y, z belong to the monoid B_n^+[3]. We say that z is a *least common right-multiple*, or *right-lcm*, of x and y if z is a right-multiple of x and of y, and, for every z' that is a right-multiple of x and y, we have $z \preccurlyeq z'$.

In other words, a right-lcm is a supremum with respect to the left-divisibility relation.

Proposition 2.7 (existence of lcm).— *Any two elements of B_n^+ admit a unique right-lcm.*

Proof. Let x, y be elements of B_n^+. By construction, there exist positive braid words u, v that represent x and y. Let z be the braid represented by the words $uC(u,v)$ and $vC(v,u)$ (which, by (2.4)) are equivalent. By construction, z is a right-multiple of x and of y.

Now that z' is an arbitrary common right-multiple of x and y. This means that there exist positive braid words u', v' satisfying $uv' \equiv^+ vu'$ and such that z' is represented by uv' and vu'. By (2.5), there must exist a positive braid word w satisfying

$$u' \equiv^+ C(v,u)\,w \text{ and } v' \equiv^+ C(u,v)\,w.$$

This means that we have $z' = z \cdot \overline{w}$, *i.e.*, z' is right-multiple of z. So z is a right-lcm of x and y.

Finally, assume that z' is another right-lcm of x and y. Because z' is a common right-multiple of x and y, we have $z \preccurlyeq z'$ by the above argument. On the other hand, as z is a common right-multiple of x and y and z is a right-lcm of x and y, we must have $z' \preccurlyeq z$ as well. By Lemma 2.2, the left-divisibility relation is antisymmetric, we deduce $z' = z$. □

In the sequel, the right-lcm of two positive braids x, y is denoted by $\mathrm{lcm}_R(x,y)$. Using an induction on the cardinal, it is easy to deduce

Corollary 2.8.— *Every nonempty finite set of positive braids admits a right-lcm*[4].

Symmetrically, there is the notion of a greatest common left-divisor.

Definition 2.9 (greatest common left-divisor or **left-gcd).**— Assume that x, y, z belong to the monoid B_n^+. We say that z is a *greatest common left-divisor*, or *left-gcd*, of x and y if z is a left-divisor of x and of y, and, for every z' that is a left-divisor of x and y, we have $z' \succcurlyeq z$.

In other words, a left-gcd is an infimum with respect to the left-divisibility relation.

Proposition 2.10 (existence of gcd).— *Any two elements of B_n^+ admit a unique left-gcd.*

Proof. Let x, y be positive braids in B_n^+, and let X be the set of all common left-divisors of x and y. The set X is nonempty as it contains at least the trivail braid 1, and, by Proposition 1.4(*iii*), it is finite since $z \prec x$ implies $|z| \leqslant |x|$. By Corollary 2.8, the set X admits a right-lcm z.

First, we claim that z belongs to X, *i.e.*, z is a left-divisor of x and y. Indeed, by hypothesis, x is a right-multiple of every element of X, hence, by definition of a right-lcm, it is a right-multiple of the right-lcm of X, *i.e.*, $z \preccurlyeq x$ holds. For symmetric reasons, we have $z \preccurlyeq y$.

[3] or, similarly, to any monoid

[4] We say that z is a right-lcm for a set X if z is a right-multiple of every element of X and every element that is a right-multiple of every element of X is a right-multiple of z

Now, let z' be any common left-divisor of x and y. By definition, z' belongs to X, hence, as z is a right-lcm of X, we have $z' \preccurlyeq z$. Hence z is a left-gcd for x and y.

As for uniqueness, the argument is the same as for the uniqueness of the lcm, and it follows from Proposition 1.4(*ii*). □

Thus we proved that, for each n, the left-divisibility relation gives to the ordered set $(B_n^+, \preccurlyeq)$ the structure of a *lattice*.

Finally, we observe that, as B_n^+ is not commutative for $n \geqslant 3$, there are the symmetric notions of a right-divisor and a left multiple. It is easy to check that the right-divisibility relation enjoys the same lattice properties as the left-divisibility relations—but we shall mostly use left-divisors here.

2.3. The right-complement operation \. We introduce one more binary operation, derived from the right-lcm operation.

Definition 2.11 (right-complement).— If x, y are positive braids, the *right-complement* of x in y, denoted $x\backslash y$ ("x under y") is the unique braid z that satisfies $xz = \mathrm{lcm}_R(x, y)$.

The uniqueness of the right-complement is guaranteed by the fact that the monoid B_n^+ is left-cancellative, *i.e.*, $xz = xz'$ implies $z = z'$, as was shown in Chapter III.

Lemma 2.12.— *For all positive braids x, y, z,*
(*i*) *$x \preccurlyeq y$ is equivalent to $y\backslash x = 1$,*
(*ii*) *we have $x\backslash(yz) = x\backslash y \cdot (y\backslash x)\backslash z$ and $(yz)\backslash x = z\backslash(y\backslash x)$,*
(*iii*) *$x \preccurlyeq yz$ is equivalent to $y\backslash x \preccurlyeq z$.*

The proof is left as an exercise.

Exercise 2.13.— Show that every (finite or infinite) set of positive braids admits a left-gcd.

3. The greedy normal form, case of positive braids

We are interested in constructing a normal form for (arbitrary) braids, *i.e.*, associating with every braid a distinguished braid word that represents it. In this section, we begin with the more restricted aim of finding a distinguished representative for positive braids. The extension to general braids will be made in the next section.

3.1. Permutation braids. We introduce a special family of positive braids canonically associated with permutations. We recall from Chapter II that a permutation of $\{1, ..., n\}$ denoted $\mathrm{perm}(x)$ is associated with each braid x in B_n: $\mathrm{perm}(x)(i) = j$ holds if and only if the strand that finishes at position i in x starts from position j. With this definition, perm is a surjective homomorphism of the group B_n onto the symmetric group $\mathfrak{S}_n$[5]. We shall introduce below a distinguished section of this surjection.

We begin with some notations. First, the transposition that exchanges i and $i+1$ will be denoted s_i; thus s_i is the permutation associated by the braid σ_i.

Notation 3.1.— For $1 \leqslant i \leqslant j$, we put

$$\sigma_{i,j} = \begin{cases} 1 & \text{for } i = j, \\ \sigma_i \sigma_{i+1} ... \sigma_{j-1} & \text{for } i < j. \end{cases} \tag{3.2}$$

.

Pictorially, the braid $\sigma_{i,j}$ corresponds to the ith strand going right to position j passing under the intermediate strands. For instance, $\sigma_{i,i+1}$ is σ_i. As in the case of σ_i, we shall use $\sigma_{i,j}$ to denote both the braid word defined above and the braid it represents.

Lemma 3.3.— *For $i \leqslant j < k-1$, we have*

$$\sigma_{i,k}\, \sigma_j = \sigma_{j+1}\, \sigma_{i,k}. \tag{3.4}$$

[5]We recall that, if f, g are permutations (or, more generally, functions, fg denotes the composition of f and g, *i.e.*, g followed by f: for each i, we have $fg(i) = f(g(i))$. This is why the correspondence between braids and permutations is defined in this way.

PROOF. We find for $j \leqslant k-1$

$$\begin{aligned}\sigma_{i,k}\,\sigma_j &= \sigma_(i,j)\ \sigma_j\ \sigma_{j+1}\ \sigma_{ii+2,k}\ \sigma_j\\ &= \sigma_(i,j)\ \sigma_j\ \sigma_{j+1}\ \sigma_j\ \sigma_{ii+2,k}\\ &= \sigma_(i,j)\ \sigma_{j+1}\ \sigma_j\ \sigma_{j+1}\ \sigma_{ii+2,k}\\ &= \sigma_{j+1}\ \sigma_{1,j}\ \sigma_j\ \sigma_{j+1}\ \sigma_{ii+2,k} = \sigma_{j+1}\ \sigma_{i,k},\end{aligned}$$

which is the expected result. □

Definition 3.5 (permutation braid).— (See Figure 1.) For f a permutation of $\{1,...,n\}$, we define the positive braid word $\mathrm{bw}(f)$ by $\mathrm{bw}(\mathrm{id}) = \varepsilon$ and $\mathrm{bw}(f) = \sigma_{f(k),k}\ \mathrm{bw}(g)$, where k is the largest number moved by f, and g is the permutation of $\{1,...,n-1\}$ defined by

$$g(i) = \begin{cases} f(i) & \text{for } i < k \text{ and } f(i) < f(k),\\ f(i)-1 & \text{for } i < k \text{ and } f(i) > f(k),\\ i & \text{for } i \geqslant k.\end{cases} \tag{3.6}$$

We denote by $\mathrm{br}(f)$ the braid represented by $\mathrm{bw}(f)$, and call it the *permutation braid* associated with f.

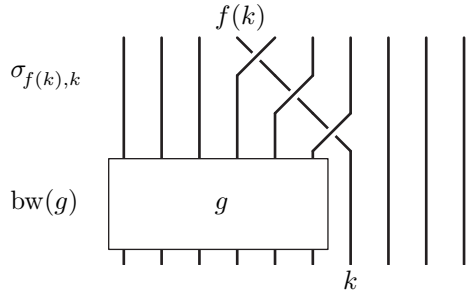


FIGURE 1. Inductive definition of the permutation braid word $\mathrm{bw}(f)$.

It is easy to check that $\mathrm{br}(s_i)$ is equal to σ_i and, more generally, to establish the following using an induction on n.

Lemma 3.7.— *For $n \geqslant 1$, the* br *mapping is a (set-theoretical) section of* perm*, i.e., for every permutation f of $\{1,...,n\}$, we have* $\mathrm{perm}(\mathrm{br}(f)) = f$.

A significant role is played by the so-called *flip* permutation ω_n that maps i to $n-i$ for each i in $\{1,...,n\}$.

Lemma 3.8.— *For $n \geqslant 1$, put $\underline{\Delta}_n = \mathrm{bw}(\omega_n)$. Then the braid represented by $\underline{\Delta}_n$ is the braid Δ_n of* (2.3).

PROOF. We already observed that the permutation associated with Δ_n is the flip ω_n. By Lemma 3.7, the same holds for the braid represented by $\underline{\Delta}_n$. But two positive braids may have the same permutation and not be equal, so that is *not* sufficient to conclude anything.

Actually, we check the result using an induction on n. The result is obvious for $n = 1$. Assume $n \geqslant 2$. Applying (3.6), we have $\underline{\Delta}_n = \sigma_{1,n} g$, where g is the permutation of $\{1,...,n-1\}$ defined for each i by $g(i) = \omega_n(i) - 1 = n - i - 1 = \omega_{n-1}(i))$. Hence we obtain $\underline{\Delta}_n = \sigma_{1,n}\,\underline{\Delta}_{n-1}$. By induction hypothesis, the braid represented by $\underline{\Delta}_{n-1}$ is Δ_{n-1}, so the braid represented by $\underline{\Delta}_n$ is $\sigma_{1,n}\Delta_{n-1}$, which is Δ_n by (2.3). □

3.2. Simple braids. We introduce a new family of positive braids. At the end, we shall see that they coincide with permutation braids, but, at first, we start from a different point of view.

Definition 3.9 (simple braid).— For f a permutation of $\{1,...,n\}$, we denote by $\mathrm{Inv}(f)$ the *inversion number* of f, *i.e.*, the number of ordered pairs (i,j), $1 \leqslant i < j \leqslant n$, satisfying $f(i) > f(j)$. We say that a braid x is *simple* if x is positive and $|x| = \mathrm{Inv}(\mathrm{perm}(x))$ holds.

Every braid σ_i is simple, for we have $|\sigma_i| = \mathrm{Inv}(\mathrm{perm}(\sigma_i)) = \mathrm{Inv}(s_i) = 1$. The unit braid 1 is simple as well, as $|1| = \mathrm{Inv}(\mathrm{id}) = 0$ holds. On the other hand, the braid σ_1^2 is not simple, as we have $|\sigma_1^2| = 2$ and $\mathrm{Inv}(\mathrm{perm}(\sigma_1^2)) = \mathrm{Inv}(\mathrm{id}) = 0$. An induction shows that a positive braid x is simple if and only if any

two strands cross at most once in any positive braid diagram that represents x. As $\mathrm{Inv}(f) \leqslant n(n+1)/2$ holds for every permutation f of n integers, the length of a simple braid in B_n^+ is bounded by $n(n+1)/2$, and, therefore, there exist only finitely many simple braids in B_n^+. The main property of simple braids for our current purpose is that they are closed under left- and right-divisors, as will be proved below.

Lemma 3.10.— *Assume that x, y are positive braids and the product xy is simple. Then both x and y are simple.*

Proof. For every permutation f and every integer i, we have

$$\mathrm{Inv}(s_i f) = \begin{cases} \mathrm{Inv}(f)+1 & \text{if } f^{-1}(i) < f^{-1}(i+1) \text{ holds,} \\ \mathrm{Inv}(f)-1 & \text{if } f^{-1}(i) > f^{-1}(i+1) \text{ holds.} \end{cases} . \tag{3.11}$$

This comes from the direct comparison, for each i, of the number of j's below i satisfying $f(j) > f(i)$ and the number of j's below i satisfying $f'(j) > f'(i)$, where f' is $s_i f$.

Formula (3.11) implies inductively that $\mathrm{Inv}(\mathrm{perm}(x)) \leqslant |x|$ holds for every positive braid x. It also implies the inequality

$$\mathrm{Inv}(\mathrm{perm}(xy)) \leqslant |x| + \mathrm{Inv}(\mathrm{perm}(y)).$$

So $\mathrm{Inv}(\mathrm{perm}(y)) < |y|$ implies $\mathrm{Inv}(\mathrm{perm}(xy)) < |xy|$, and, therefore, the hypothesis that xy is simple implies that y is simple.

The argument is symmetric for left-divisors, using the formula

$$\mathrm{Inv}(f s_i) = \begin{cases} \mathrm{Inv}(f)+1 & \text{if } f(i) < f(i+1) \text{ holds,} \\ \mathrm{Inv}(f)-1 & \text{if } f(i) > f(i+1) \text{ holds} \end{cases} .$$

which follows from applying (3.11) to f^{-1}. □

Lemma 3.12.— *Assume that f is a permutation of $\{1, ..., n\}$ and $\mathrm{br}(f)$ is a simple braid. Then, for $1 \leqslant i \leqslant n-1$, two cases are possible:*

(i) Either $f^{-1}(i) < f^{-1}(i+1)$ holds, the braid $\sigma_i \,\mathrm{br}(f)$ is simple, and we have $\sigma_i \,\mathrm{br}(f) = \mathrm{br}(s_i f)$;

(ii) Or $f^{-1}(i) > f^{-1}(i+1)$ holds, and $\sigma_i \,\mathrm{br}(f)$ is not simple.

Proof. By hypothesis, $|\mathrm{br}(f)| = \mathrm{Inv}(f)$ holds. In case (i), we have $|\sigma_i \,\mathrm{br}(f)| = \mathrm{Inv}(f)+1 = \mathrm{Inv}(s_i f)$, so the braid $\sigma_i \,\mathrm{br}(f)$ is simple. In case (ii), we have $|\sigma_i \,\mathrm{br}(f)| = \mathrm{Inv}(f)+1$ and $\mathrm{Inv}(s_i f) = \mathrm{Inv}(f)-1$, and $\sigma_i \,\mathrm{br}(f)$ is not simple.

So it remains to prove that $f^{-1}(i) < f^{-1}(i+1)$ implies $\mathrm{br}(s_i f) = \sigma_i \,\mathrm{br}(f)$. We use induction on $\mathrm{Inv}(f)$ (or, equivalently, on the largest number moved by f). The result is true for $f = \mathrm{id}$, as $\mathrm{br}(f) = 1$ holds then. So, we assume $f \neq \mathrm{id}$. Let k be the largest number moved by f. Then $f(k) < k$ necessarily holds. By definition, we have $\mathrm{br}(f) = \sigma_{f(k),k} \,\mathrm{br}(g)$, where g is as in Definition 3.9. Put $f' = s_i f$. Five cases are to be considered.

Assume first $i < f(k) - 1$. Then the largest number moved by f' is k, which is mapped to $f(k)$. Hence we have $\mathrm{br}(f') = \sigma_{f(k),k} \,\mathrm{br}(g')$, where g' is $s_i g$ (check it!). By induction hypothesis, we have $\mathrm{br}(g') = \sigma_i \,\mathrm{br}(g)$, so, as σ_i and $\sigma_{f(k),k}$ commute since $i \leqslant f(k) - 2$ holds, we find

$$\mathrm{br}(f') = \sigma_{f(k),k} \,\mathrm{br}(g') = \sigma_{f(k),k} \,\sigma_i \,\mathrm{br}(g) = \sigma_i \,\sigma_{f(k),k} \,\mathrm{br}(g) = \sigma_i \,\mathrm{br}(f).$$

Assume now $i = f(k) - 1$. This times, the largest number moved by f' is still k, which is mapped to $f(k) - 1$. With the same notation, we find

$$\mathrm{br}(f') = \sigma_{f(k)-1,k} \,\mathrm{br}(g) = \sigma_i \,\sigma_{f(k),k} \,\mathrm{br}(g) = \sigma_i \,\mathrm{br}(f).$$

Third, the case $i = f(k)$ is impossible as, by hypothesis, we have $k = f^{-1}(f(k)) > f^{-1}(f(k)+1)$.

Next, assume $f(k) + 1 \leqslant i \leqslant k - 1$. Then the largest number moved by f' is still k, which is moved to $f(k)$. In this case, one obtains $g' =$
$ss_{i-1}g'$. Applying Lemma 3.3 and the induction hypothesis, we find now

$$\mathrm{br}(f') = \sigma_{f(k),k} \,\mathrm{br}(g') = \sigma_{f(k),k} \,\sigma_{i-1} \,\mathrm{br}(g) = \sigma_i \,\sigma_{f(k),k} \,\mathrm{br}(g) = \sigma_i \,\mathrm{br}(f).$$

Finally, assume $ii \geqslant k$. Then the largest number moved by f' is $i+1$, which is mapped to i, and we have $g' = f$. Then we find directly $\mathrm{br}(f') = \sigma_i \,\mathrm{br}(f)$. □

Proposition 3.13.— *Assume $n \geqslant 2$. For every positive braid x in B_n^+, the following are equivalent:*

(i) The braid x is a permutation braid;

(ii) The braid x is simple;

(*iii*) *The braid x is a right-divisor of Δ_n;*
(*iv*) *The braid x is a left-divisor of Δ_n.*

PROOF. Assume that f is a permutation of $\{1, ..., n\}$. We prove that $\mathrm{br}(f)$ is simple using induction on the inversion number $\mathrm{Inv}(f)$. For $\mathrm{Inv}(f) = 0$, f is the identity, and the result is obvious. Otherwise, there exists at least one integer i satisfying $f^{-1}(i) > f^{-1}(i+1)$. Let $g = s_i f$. Because s_i^2 is the identity, we also have $f = s_i g$, and $\mathrm{Inv}(g) < \mathrm{Inv}(f)$. By induction hypothesis, $\mathrm{br}(g)$ is simple. By Lemma 3.12, $\mathrm{br}(f)$ is $\sigma_i \,\mathrm{br}(g)$, and it is simple. So (*i*) implies (*ii*).

Conversely, we prove using induction on $\mathrm{Inv}(\mathrm{perm}(x))$ that, if x is simple, then $x = \mathrm{br}(\mathrm{perm}(x))$ holds. For $\mathrm{Inv}(\mathrm{perm}(x)) = 0$, the hypothesis that x is simple implies $|x| = 0$, hence $x = 1$, and x is a permutation braid. Otherwise, write $x = \sigma_i\, y$. Then we have $\mathrm{perm}(x) = s_i \,\mathrm{perm}(y)$. We have $|x| = |y|+1$, and $\mathrm{Inv}(\mathrm{perm}(x)) \leqslant \mathrm{Inv}(\mathrm{perm}(y)) + 1$ by Lemma 3.12, hence y must be simple, with $\mathrm{Inv}(\mathrm{perm}(y)) = \mathrm{Inv}(\mathrm{perm}(x)) - 1$. So, by induction hypothesis, we have $y = \mathrm{br}(\mathrm{perm}(y))$. By Lemma 3.12, we deduce $x = \mathrm{br}(s_i \,\mathrm{perm}(y))$, *i.e.*, $x = \mathrm{br}(\mathrm{perm}(x))$. So (*ii*) implies (*i*).

Assume $f \neq \omega_n$. Then there exists i satisfying $f^{-1}(i) < f^{-1}(i+1)$, and we have $\mathrm{Inv}(s_i f) > \mathrm{Inv}(f)$ and $\sigma_i \mathrm{br}(f) = \mathrm{br}(s_i f)$. Applying the same argument to $s_i f$ and iterating, we find a permutation g satisfying $gf = \omega_n$, hence $\mathrm{br}(gf) = \mathrm{br}(g)\mathrm{br}(f)$. By Lemma 3.8, we have $\mathrm{br}(\omega_n) = \Delta_n$, and, therefore, Δ_n is a left-multiple of $\mathrm{br}(f)$. Hence (*i*) and (*ii*) imply (*iii*).

The previous argument shows that, for every simple braid x in B_n^+, there exists a simple braid y that satisfies $yx = \Delta_n$. Let us denote the latter braid by $\phi(x)$. Thus ϕ is a mapping of the set S consisting of all simple braids in B_n^+ into itself. As the monoid B_n^+ admits left cancellation, the mapping ϕ is injective. Hence, as S is finite, ϕ is also surjective. Hence, for every y in S, there exists x in S satisfying $\Delta_n = yx$. In particular, Δ_n is a right-multiple of c. So (*i*) and (*ii*) implie (*iv*).

Finally, assume $\Delta_n = xy$. Then x and y are simple by Lemma 3.10. □

Corollary 3.14.— *Simple braids are closed under right-lcm and right-complement.*

PROOF. Assume that x, y are simple braids lying in B_n^+. By Proposition 3.13, x and y are left-divisors of Δ_n. Hence so is their eight-lcm. By Proposition 3.13 again, it follows that the latter is simple. On the other hand, we are $x \cdot x\backslash y = \mathrm{lcm}_R(x, y)$. The simplicity of $\mathrm{lcm}_R(x, y)$ plus Lemma 3.10 imply that $x\backslash y$ is simple. □

3.3. The head of a positive braid.

Definition 3.15 (head).— For each positive n-strand braid x, the *head* of x, denoted $H(x)$, is the left-gcd of x and Δ_n.

The head of the trivial braid 1 is certainly 1, as 1 has no left-divisor except itself. The head of σ_i is σ_i, as, by Lemma III.1.17, σ_i is a left-divisor of Δ_n and it admits no left-divisor except 1 and itself. The head of Δ_n is Δ_n.

Lemma 3.16.— (*i*) *For each positive braid x, the head of x is the unique maximal simple braid that left-divides x; the relation $y = H(x)$ is true if and only if y is a simple left-divisor of x and one has*

$$\forall \text{ simple } z\,(\, z \preccurlyeq x \Rightarrow z \preccurlyeq y\,). \tag{3.17}$$

(*ii*) *The relation $H(x) = 1$ holds if and only if x is trivial.*

PROOF. (*i*) By construction, $H(x)$ is a left-divisor of Δ_n, hence, by Proposition 3.13, it is a simple braid, and it left-divides x by definition. On the other hand, let z be any simple braid that left-divides x. By Proposition 3.13 again, z left-divides Δ_n, hence it must divide the left-gcd of x and Δ_n, which is $H(x)$. So $H(x)$ is the macimal simple braid that left-divides x, and (3.18) holds.

Conversely, assume that y is a simple left-divisor of x satisfying (3.18). Then we have $y \preccurlyeq H(x)$ by definition of the head. On the other hand, (3.18) implies in particular to $z = H(x)$, in which case it gives $H(x) \preccurlyeq y$. Hence we have $y = H(x)$.

(*ii*) If x is not equal to 1, then it is left-divisible by at least one generator σ_i. Then σ_i is a left-divisor of x and of Δ_n, hence of $H(x)$. Hence $H(x)$ cannot be equal to 1. □

In this way, we have obtained, for each positive braid x, a distinguished decomposition

$$x = H(x) \cdot x', \tag{3.18}$$

where the first factor is a simple braid, *i.e.*, equivalently, a permutation braid. By iterating the process, we shall obtain a decomposition of every positive braid into a product of finitely simple braids.

Definition 3.19 (normal sequence).— A sequence $(x_1, ..., x_d)$ of simple n-strand braids is said to be *normal* if it is either empty (case $d = 0$) or, for each k, one has $x_k = H(x_k...x_d)$ and $x_d \neq 1$.

Proposition 3.20 (normal form).— *Every positive braid admits a unique normal decomposition; more precisely, for every nontrivial positive braid x, there exists a unique normal sequence $(x_1, ..., x_d)$ such that x_d is not trivial and $x = x_1...x_d$ holds.*

We naturally consider the empty sequence () as being the normal form of the trivial braid 1.

Proof. We use induction on $|x|$. The result is vacuously true for $|x| = 0$, *i.e.*, for $x = 1$. It is also obviously true for $|x| = 1$, *i.e.*, when x is one of the generators σ_i. Then x itself is simple, and the length one sequence (x) is the solution, and it is unique. So assume $|x| \geqslant 2$. By Lemma 3.16(ii), the simple braid $H(x)$ is not 1. Hence, in the decomposition of (3.18), we have $H(x) \neq 1$, whence $|x'| < |x|$. If x' is trivial, then, as in the case of σ_i, we have a length one normal decomposition $(H(x))$, and we are done? Otherwise, we apply the induction hypothesis to get a normal decomposition $(x_2, ..., x_d)$ for x'. Then $(x_1, x_2, ..., x_d)$ is a normal decomposition for x. This shows the existence.

As for uniqueness, it is clear since, by definition, the first factor of a normal decomposition of x must be the head of x. □

The unique normal sequence provided by Proposition 3.20 will naturally be called the *normal decomposition* of the braid x, or its *greedy normal form*. This terminology comes from the definition of the head: at each step, we take as much as we can of the current remainder. For future use, we note the following convenient characterization, which directly follows from (3.18) in Lemma 3.16.

Exercise 3.21.— Show that the normal decomposition of σ_1^2 is the length two sequence (σ_1, σ_1), whereas the normal decomposition of $\sigma_1\sigma_2$ is the length one sequence $(\sigma_1\sigma_2)$. What are the normal decompositions of Δ_n^k, of σ_1^k, and of $\sigma_1^2\sigma_2^2$?

Lemma 3.22.— *Assume that x_1, x_2 are simple braids. Then (x_1, x_2) is normal if and only if*

$$\forall \text{ simple } z\,(\, z \preccurlyeq x_1x_2 \Rightarrow z \preccurlyeq x_1\,). \tag{3.23}$$

3.4. Local characterization. We shall see below that the greedy normal form enjoys various good properties. They all follow from the following alternative characterization.

Lemma 3.24.— *Assume that $(x_1, ..., x_d)$ is a sequence of simple n-braids. Then the following are equivalent:*

(i) *The sequence $(x_1, ..., x_d)$ is normal;*

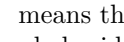
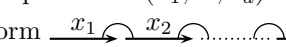
(ii) *For $1 \leqslant k < dd$, the sequence (x_k, x_{k+1}) is normal;*

Proof. Assume (i) and $k < dd$. For every sequence of simple braids $(x_k, ..., x_d)$, we have

$$x_k \preccurlyeq H(x_kx_{k+1}) \preccurlyeq H(x_k...x_d). \tag{3.25}$$

The hypothesis that $(x_1, ..., x_d)$ is normal implies $x_k = H(x_k...x_d)$. Owing to (3.25), we deduce $x_k = H(x_kx_{k+1})$, so (x_k, x_{k+1}) is normal, and (i) implies (ii).

The converse implication is the nontrivial point. We use induction on $y_* \geqslant 2$. For $y_* = 2$, there is nothing to prove. So we assume $y_* \geqslant 3$, and (x_k, x_{k+1}) is normal for each k. We aim at proving that $(x_1, ..., x_d)$ is normal. By the induction hypothesis, the sequence $(x_2, ..., x_d)$ is normal, and the only result to prove is $x_1 = H(x_1...x_d)$, *i.e.*, according to Lemma 3.16, we wish to show that each simple braid z left-dividing $x_1...x_d$ left-divides x_1. So assume $z \preccurlyeq x_1...x_d$. By Lemma 2.12(i), we have $(x_1x_2...x_d)\backslash z = 1$, hence, by Lemma 2.12($ii$), $(x_2...x_d)\backslash(x_1\backslash z) = 1$, which, by Lemma 2.12($iii$), implies $x_1\backslash z \preccurlyeq x_2...x_d$. Corollary 3.14 implies that $x_1\backslash z$ is simple, so the normality of $x_2...x_d$ implies $x_1\backslash z \preccurlyeq x_2$, which, by Lemma 2.12($iii$) again, implies $z \preccurlyeq x_1x_2$. By hypothesis, (x_1, x_2) is normal, so we deduce $z \preccurlyeq x_1$. Hence (ii) implies (i). □

In a diagram, the property that a pair (x_1, x_2) is normal will be indicated by drawing a small arc connecting the target end of the arrow associated with x_1 to the source end of the arrow associated with x_2: so  means that (x_1, x_2) is normal. It follows from the characterization of Lemma 3.24 that a sequence of simple braids $(x_1, ..., x_d)$ is normal if and only if (x_d is non-trivial and) it corresponds to a picture of the form  .

3.5. Computation of the normal form. The normal decomposition yields a distinguished expression for each positive braid. Indeed, in Definition 3.5, we have chosen a distinguished braid word representative for each simple braid, hence we obtain a distinguished word representative for every positive braid by concatenating the distinguished words associated with the successive factors of its normal decomposition. For instance, the distinguished word representing the braid Δ_3 is the word $\underline{\Delta}_3$, *i.e.*, $\sigma_1\sigma_2\sigma_1$, so the distinguished word representative of the braid Δ_3^2, whose normal decomposition is (Δ_3, Δ_3) is the word $\sigma_1\sigma_2\sigma_1\sigma_1\sigma_2\sigma_1$.

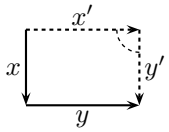
In itself, defining a normal form, *i.e.*, choosing a distinguished element in each equivalence class, has no interest unless one can (efficiently) compute this distinguished element. What makes the interest of the current normal form is the existence of very efficient algorithms for computing it. This is what we shall explain now.

First, we have a basic procedure for the case of two simple braids.

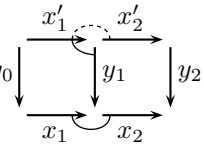
Lemma 3.26.— *For each pair of simple braids x, y, there exist simple braids x', y' satisfying $x'y' = xy$ and such that (x', y') is normal (or has length one),*

PROOF. Write $x' = H(xy)$ and $xy = x'\,y'$. By construction, we have $x \preccurlyeq x'$, say $x' = xz$. Then we have $xy = xzy'$, whence $y = zy'$. By Lemma 3.10 we deduce that y' is simple. So (assuming that at least one of x, y is not trivial) the normal decomposition of xy is either the length one sequence (x') if y' is trivial, *i.e.*, if xy is simple, or the length two sequence (x', y'). □

In other words, the length of the normal decomposition of xy is at most two. This corresponds to the possibility of completing every diagram of the following type



Then we play domino.

Lemma 3.27 (domino rule 1).— *Assume that the diagram* 

is commutative and (x_1, x_2) and (x'_1, y_1) are normal. Then (x'_1, x'_2) is normal as well.

PROOF. Assume that z is simple and left-divides $x'_1x'_2$. A fortiori we have $z \preccurlyeq x'_1x'_2y_2$, hence $z \preccurlyeq y_0x_1x_2$ using the commutativity of the diagram. By Lemma 2.12, we deduce $y_0\backslash z \preccurlyeq x_1x_2$. By Corollary 3.14, $y_0\backslash z$ is simple, and, by hypothesis, (x_1, x_2) is normal. We deduce $y_0\backslash z \preccurlyeq x_1$, whence $z \preccurlyeq y_0x_1$ by Lemma 2.12, *i.e.*, $z \preccurlyeq x'_1y_1$ by commutativity of the diagram. As z is simple and (x'_1, y_1) is normal, we deduce $z \preccurlyeq x'_1$ and, therefore, (x'_1, x'_2) is normal. □

Now we can prove the main result.

Proposition 3.28 (left-multiplication).— *Assume that $(x_1, ..., x_d)$ is the normal decomposition of a positive braid x, and s is a simple braid. Then the normal decomposition of the braid sx is $(x'_1, ..., x'_d, s_d)$—or $(x'_1, ..., x'_d)$ if s_d is trivial—where we put $s_0 = s$ and, inductively, (x'_i, s_i) is the normal decomposition of $s_{i-1}x_i$ for i increasing from 0 to $d-1$ (see Figure 2).*

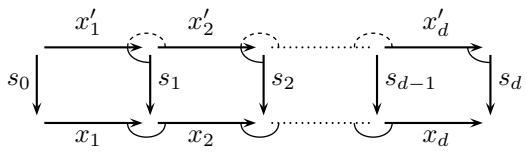


FIGURE 2. Computing the normal form of sx from that of x: starting from $s_0 = s$, take the normal decomposition (x'_1, s_1) of s_0x_1, then the normal decomposition (x'_2, s_2) of s_1x_2, and so on from left to right; the sequence $(x'_1, ..., x'_d, s_d)$ is normal.

PROOF. First, the existence of $x'_1, ..., s_d$ follows from Lemma 3.26. Then, the commutativity of the diagram of Figure 2 gives $x'_1x'_2...x'_ds_d = s_0x_1x_2...x_d = sx$, so $(x'_1, ..., x'_d, s_d)$ is a decomposition of sx. Moreover, each braid x'_i and s_d is simple by construction.

So it only remains to check that the sequence $(x'_1, ..., x'_d, s_d)$ is normal. As for the last two entries x'_d and s_d, this follows from their construction. Now assume $i \leqslant d-1$. Then (x_i, x_{i+1}) is normal by hypothesis, and so is (x'_i, s_i) by construction. Moreover, always by construction, we have $x'_i s_i = s_{i-1} x_i$ and $x'_{i+1} s_{i+1} = s_i x_{i+1}$. Then Lemma 3.27 implies that (x'_i, x'_{i+1}) is normal. □

Corollary 3.29.— *For each n, the greedy normal form of a positive braid of length ℓ in B_n^+ can be computed in time $O(\ell^2)$.*

Proof. There are only finitely many pairs of simple braids in B_n^+, namely $(n!)^2$. So, *for fixed n*, we can compute the normal form of a pair of simple braids in constant time $O(1)$. By Proposition 3.28, computing the normal form of $\sigma_i x$ from that of x requires at most $O(|x|)$ steps, as the length of the normal form of x is always bounded above by the length of x. Thus, starting from the trivial braid and applying Proposition 3.28 ℓ times, we determine the normal form of a length ℓ positive braid word in at most $O(\ell^2)$ steps. □

4. The greedy normal form, general case

We now turn to the case of arbitrary, not necessarily positive braids. We shall define two different normal forms.

4.1. The Delta-normal form. Using the powers of the braid Δ_n, it is very easy to deduce a distinguished decomposition for each braid from the greedy normal form of positive braids.

Lemma 4.1.— *For every braid z in B_n, there exists a unique pair (p, x) where p is an integer and x is a positive braid in B_n^+ that satisfy*

$$z = \Delta_n^p \cdot x \quad \text{with} \quad \Delta_n \not\preccurlyeq x. \tag{4.2}$$

Proof. We observed in Corollary III.1.20 that, if z can be expressed by a braid word containing k negative letters σ_i^{-1}, then the braid $\Delta_n^k z$ belongs to the monoid B_n^+. Let P be the set of all integers k such that $\Delta_n^{-k} z$ belongs to B_n^+. As recalled above, P is nonempty. Choose a braid word w representing z, and let r be the number of positive letters in w. Then, for $k \geqslant r$, the number of negative letters in the word $\underline{\Delta}_n^{-k} w$ is larger than the number of positive letters and, so, this word cannot represent a positive braid. So P is included in the interval $(-\infty, r)$ and, therefore, P has a maximal element p. Then we have $z = \Delta_n^p \cdot x$ for some positive braid x, and the choice of p implies $\Delta_n \not\preccurlyeq x$ (for otherwise we could write $z = \Delta_n^{p+1} x'$ for some positive x').

The choice of p and the fact that the group B_n is cancellative (as is every group) guarantees the uniqueness. □

Combining Lemma 4.1 with the greedy normal form on B_n^+ immediately provides a distinguished expression for each braid.

Proposition 4.3 (normal form).— *For each braid z in B_n, there exists a unique integer p and a unique normal sequence $(x_1, ..., x_d)$ satisfying $x_1 \neq \Delta_n$, $x_d \neq 1$ and $z = \Delta_n^p x_1 \dots x_d$.*

Proof. The only point to prove is that, if $(x_1, ..., x_d)$ is the normal decomposition of some positive braid x of B_n^+, then the condition $\Delta_n \not\preccurlyeq x$ is equivalent to $x_1 \neq \Delta_n$. This follows from the fact that Δ_n is simple, and, therefore, $\Delta_n \preccurlyeq x$ is equivalent to $\Delta_n \preccurlyeq H(x)$, which is $\Delta_n \preccurlyeq x_1$ by construction. □

Definition 4.4 (Δ-normal form).— In the context of Proposition 4.3, we say that $(\Delta_n^p \mid x_1, ..., x_d)$ is the *Δ-normal form* of z.

Recognizing that a sequence $(p \mid x_1, ..., x_d)_n$ is a Δ-normal form, *i.e.*, there exists a braid it is the Δ-normal of which, is easy: the condition is simply that x_1 is not Δ_n, (x_k, x_{k+1}) is normal for each k, and x_d is not 1. If z is a braid and we have a decomposition that satisfies the above requirements, then this is the expected normal form.

Example 4.5.— For $n \geqslant \max(i+1, 3)$, the Δ-normal form of σ_i is $(\Delta_n^0 \mid \sigma_i)$. Indeed, σ_i is a positive simple braid, and it is not left-divisible by Δ_n. By contrast, the Δ_2-normal form of σ_1 is $(\Delta_2^1 \mid 1)$, as we have $\Delta_2 = \sigma_1$.

More interestingly, for $n \geqslant \max(i+1, 3)$, the Δ-normal for of σ_i^{-1} is $(\Delta_n^{-1} \mid x)$, where x is the unique positive braid that satisfies $x\sigma_i = \Delta_n$. For instance, the Δ_3-normal form of σ_1^{-1} is $(\Delta_3^{-1} \mid \sigma_1\sigma_2)$, whereas that of σ_2^{-1} is $(\Delta_3^{-1} \mid \sigma_2\sigma_1)$.

If we insist on having really one distinguished braid word for each braid, then we can use the unique permutation braid words of Definition 3.5 for each simple braid. So, for instance, the Δ_3-normal word representing σ_1^{-1} would be the word $\underline{\Delta}_3^{-1}\sigma_1\sigma_2$, *i.e.*, $\sigma_1^{-1}\sigma_2^{-1}\sigma_1^{-1}\sigma_1\sigma_2$ (which is of course equivalent to σ_1^{-1}, but is a different word), whereas the Δ-normal word representing σ_2^{-1} would be $\sigma_1^{-1}\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_1$.

The Δ-normal form is easy to compute algorithmically.

Lemma 4.6.— *Assume that the Δ-normal form of a braid z is $(\Delta_n^p \mid x_1, ..., x_d)$. Then, for each integer e, the Δ-normal form of the braid $\Delta_n^e\, z$ is $(\Delta_n^{p+e} \mid x_1, ..., x_d)$.*

Proof. The sequence $(\Delta_n^{p+e} \mid x_1, ..., x_d)$ satisfies the requirement for being a Δ-normal form whenever $(\Delta_n^p \mid x_1, ..., x_d)$ does, and it provides a decomposition of $\Delta_n^e z$ whenever $(\Delta_n^p \mid x_1, ..., x_d)$ provides a decomposition of z. □

Lemma 4.7.— *For $n \geqslant 2$, denote by Φ_n the automorphism of B_n that maps σ_i to σ_{n-i} for $1 \leqslant i < n$. Then, for each braid x in B_n, we have*

$$x \cdot \Delta_n = \Delta_n \cdot \Phi_n(x). \tag{4.8}$$

Proof. The result is true when x is a generator σ_i (why?), and extends to arbitrary braids as Φ_n is an automorphism. □

Proposition 4.9 (computation).— *Assume that the Δ-normal form of z is $(\Delta_n^p \mid x_1, ..., x_d)$ and s is a simple braid.*

(i) The Δ-normal form of sz is computed as follows:
- *Compute the normal form $(x'_1, ..., x'_{d'})$ of $\Phi_n^p(s)x_1...x_d$ using Proposition 3.28;*
- *Let q be the largest number for which $x'_q = \Delta_n$ holds;*
- *Then the Δ-normal form of sz is $(\Delta_n^{p+q} \mid x'_{q+1}, ..., x'_{d'})$.*

(ii) The Δ-normal form of $s^{-1}z$ is computed as follows:
- *Determine s' satisfying $\Delta_n = s's$;*
- *Compute the $(x'_1, ..., x'_{d'})$ of $\Phi_n^p(s')x_1...x_d$ using Proposition 3.28;*
- *Let q be the largest number for which $x'_q = \Delta_n$ holds;*
- *Then the Δ-normal form of $s^{-1}z$ is $(p + q - 1 \mid x'_{q+1}, ..., x'_{d'})_n$.*

Proof. *(i)* By hypothesis, we have $z = \Delta_n^p x_1...x_d$, whence

$$sz = s\Delta_n^p x_1...x_d = \Delta_n^p \Phi_n^p(s)x_1...x_d.$$

The sequence $(x_1, ..., x_d)$ is normal, and $\Phi_n^p(s)$, which is s or $\Phi_n(s)$ according to the parity of p, is simple. So it makes sense to compute the normal form $(x'_1, ..., x'_{d'})$ of this positive braid using Proposition 3.28 (one then has $d' = d$ or $d' = d + 1$). Then, by construction, we have $sz = \Delta_n^p x'_1...x'_{d'}$. However, $(p \mid x'_1, ..., x'_{d'})_n$ need not be the Δ-normal form of sz because the first factors $x'_1, x'_2, ...$ may be equal to Δ_n. In this case, it suffices to remove these factors and to incorporate them in the initial power of Δ_n.

(ii) The argument is similar, owing to the fact that $y'y = \Delta_n$ implies $s^{-1} = \Delta_n^{-1}s'$. What we do is to compute the Δ-normal form of $s'z$ using the method of *(i)*, and then to use Lemma 4.6 to obtain the Δ-normal form of $\Delta_n^{-1}s'z$, *i.e.*, we remove 1 from the initial power of Δ_n. □

4.2. The symmetric normal form. The symmetric normal form has many good properties, but it is not symmetric, and it has the unpleasant property that the normal form of a braid viewed as an element of B_n need not coincide with the normal form of that braid viewed as an ellement of B_{n+1}, as was seen for σ_1^{-1} in Example 4.5. We shall now define a new normal that avoids such disadvantages.

We already observed that every braid in B_n can be expressed as the quotient of two positive braids[6]: for instance, with the symmetric normal form, we obtain such a fractionary expression in which the denominator is always a power of Δ_n (possibly a trivial one). Such an expression is never unique (we can always right-multiply the numerator and the denominator by any positive braid), but—exactly as in the case of integers—we can obtain a distinguished expression when we require in addition that the fraction is left-irreducible, *i.e.*, the numerator and the denominators have no common left-divisor.

Lemma 4.10.— *(See Figure 3.) Assume that (x, y) and (x', y') are pairs of positive braids satisfying $z = xy^{-1} = x'^{-1}y'$. Then there exist positive braids t, t', x_*, and y_* satisfying*

$$x = tx_*, \quad y = ty*, \quad x' = t'x_*, \quad \text{and} \quad y' = t' = y_*. \tag{4.11}$$

[6] one says that B_n is a group of fractions for the monoid B_n^+, see Ore's Theorem in Chapter III.

PROOF. In the monoid B_n^+, the braids x and y admit a common left-multiple, *i.e.*, there exist positive braids s, s' satisfying $sx = s'x'$. In the group B_n, we have

$$sy = sxx^{-1}y' = s'x'x'^{-1}y' = s'y'.$$

Now, let $st = s't'$ be the right-lcm of s and s'. As $sx = s'x'$ holds, sx is a right-multiple of s and s', hence it is a right-multiple of their right-lcm st, *i.e.*, there exists a positive braid x_* satisfying $sx = stx_*$, hence $x = tx_*$. As we also have $sy = s'y'$, the same argument shows that sy is a right-multiple of s and s', hence of st, and there exists a positive braid y_* satisfying $sy = sty_*$, whence $y = ty_*$. Arguing symmetrically with $s'x'$ and $s'y'$, we obtain $x' = t'x_*$ and $y' = t'y_*$. □

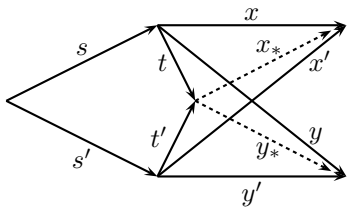


FIGURE 3. Factorizing fractionary decompositions (proof of Lemma 4.12)

Lemma 4.12.— *For each braid z in B_n, there exists a unique pair of positive braids (x, y) satisfying $z = x^{-1}y$ with $\gcd_L(x, y) = 1$.*

PROOF. First, assume that x_1, y_1 are arbitrary positive braids satisfying $z = x_1^{-1}y_1$: as recalled above, such a pair of braids certainly exists. Let z_1 be the left-gcd of x_1 and y_1, and let x, y be defined by $x_1 = z_1x$ and $y_1 = z_1y$. Then we have $z = (z_1x)^{-1}(z_1y) = x^{-1}y$. Moreover, it is easy to check, for all positive braids x, y, z_1, the equality $\gcd_L(z_1x, z_1y) = z_1 \cdot \gcd_L(x, y)$. In the current case, we deduce $z_1 = z_1\gcd_L(x, y)$, which implies $\gcd_L(x, y) = 1$. So we proved the existence of a pair (x, y) of the expected type.

We turn to uniqueness. So assume $z = xy^{-1} = x'^{-1}y'$ with $\gcd_L(x, y) = \gcd_L(x', y') = 1$. Using Lemma **??**, we obtain t, t', x_*, y_* satisfying (4.11). By construction t is a common left-divisor of x and y, so the hypothesis on the gcd implies $t = 1$. Similarly, we have $t' = 1$, whence $x = x_* = x'$ and $y = y_* = y'$, the expected uniqueness result. □

Combining Lemma 4.12 with the greedy normal form on B_n^+ provides a new distinguished expression for each braid.

Proposition 4.13 (symmetric normal form).— *For each braid z in B_n, there exists a unique pair of normal sequences $(x_1, ..., x_d)$, $(y_1, ..., y_e)$ satisfying $\gcd_L(x_1, y_1) \neq 1$ and $z = x_d^{-1}...x_1^{-1}y_1...y_e$.*

PROOF. The only point to prove is that, if $(x_1, ..., x_d)$ and $(y_1, ..., y_e)$ are the normal decompositions of two positive braids x, y of B_n^+, then the condition $\gcd_L(x, y) = 1$ is equivalent to $\gcd_L(x_1, y_1) = 1$. Now, it is obvious that, if x_1 and y_1 admit a non-trivial common left-divisor, then so do x and y. Conversely, assume that x, y admit a non-trivial common left-divisor s. Then s is left-divisible by at least one generator σ_i. As the latter is a simple braid, $\sigma_i \preccurlyeq x$ implies $\sigma_i \preccurlyeq x_1$ and, similarly, $\sigma_i \preccurlyeq y$ implies $\sigma_i \preccurlyeq y_1$. So x_1 and y_1 have a non-trivial left-gcd. □

Definition 4.14 (symmetric normal form).— In the context of Proposition 4.13, we say that the sequence $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ is the *symmetric normal form* of z.

Recognizing that a sequence $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ is a symmetric normal form, *i.e.*, there exists a braid it is the symmetric normal of which, is easy.

Proposition 4.15 (characterization).— *Assume that $(x_1, ..., x_d)$ and $(y_1, ..., y_e)$ are normal sequences. Then $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ is a symmetric normal sequence if and only if the left-gcd of x_1 and y_1 is trivial.*

PROOF. It is obvious that, if $x_1...x_d$ and $y_1...y_ee$ have no non-trivial common left-divisor, then so do x_1 and y_1. Conversely, assume that x_1 and y_1 have non non-trivial common left-divisor. Let z be a common left-divisor of $x_1...x_d$ and $y_1...y_e$. If z is not trivial, there exists i such that σ_i left-divides $x_1...x_d$ and $y_1...y_e$. As σ_i is simple and $(x_1, ..., x_d)$ is normal, this implies that σ_i left-divides the head of $x_1...x_d$, which is x_1, and, similarly, it left-divides the head of $y_1...y_e$, which is y_1. □

We shall associate to a quotient of the form $x^{-1}y$, with x, y positive, the diagram $\overleftarrow{x}\ \ \overrightarrow{y}$, and then we draw $\overleftarrow{x}\frown\overrightarrow{y}$ to indicate that the left-gcd of x and y is trivial. With such convention, a sequence of signed simple braids $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ is symmetric normal if and only if it corresponds to a diagram of the type $\overleftarrow{x_d}\frown\cdots\frown\overleftarrow{x_1}\frown\overrightarrow{y_1}\frown\cdots\frown\overrightarrow{y_e}$, with x_d non-trivial or no negative factor, and y_e non-trivial or no positive factor.

Example 4.16.— The symmetric normal form of σ_i is (σ_i). More generally, the symmetric normal form of a positive braid coincides with its normal form as constructed in Section 3.

In the other direction, the symmetric normal form of σ_i^{-1} is (σ_i^{-1}). More generally, if z is a negative braid, then its symmetric normal form is the inverse of the normal form of the positive braid z^{-1}, *i.e.*, it is $(x_d^{-1}, ..., x_1^{-1})$, where $(x_1, ..., x_d)$ is the normal of z^{-1}.

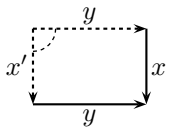
Consider now $z = \sigma_1\sigma_2^{-1}$. We claim that the symmetric form of z is the length two sequence $((\sigma_1\sigma_2)^{-1}, (\sigma_2\sigma_1))$: indeed, $\sigma_1\sigma_2$ and $\sigma_2\sigma_1$ are simple braids with no non-trivial common left-divisor and one has $z = (\sigma_1\sigma_2)^{-1}(\sigma_2\sigma_1)$.

As in the case of the Δ-normal form, the symmetric normal form is interesting only if one can compute it effectively—and the more efficiently the better. First, we have a basic procedure for the case of two simple braids with opposite signs.

Lemma 4.17.— *If x, y are simple braids, there exist a unique pair of simple braids x', y' satisfying $x'^{-1}y' = yx^{-1}$ and such that (x'^{-1}, y) is normal, i.e., the left-gcd of x' and y' is trivial.*

Proof. The existence and uniqueness of x' and y' is guaranteed by Lemma 4.12, and the only point that remains to check is that x' and y' are simple whenever x and y are. Now, by construction, we have $x'y = y'x$, and this braid is a common left-multiple z of x and y. Then $\gcd_{\scriptscriptstyle L}(x', y') = 1$ is equivalent to z being the left-lcm of x and y. As x and y are simple, z is simple, and so are its left-divisors x' and y'. □

This corresponds to the possibility of completing each diagram of the type



Next, we observe a simple connection between two forms of normality.

Notation 4.18 (duality).— For s a simple braid in B_n^+, we denote by s^* the unique simple braid such that $ss^* = \Delta_n$ holds.

(This notation is convenient, but slightly dangerous as one has to remember which n is involved.)

Lemma 4.19.— *For each simple braid s in B_n^+, one has $s^{**} = \Phi_n(s)$.*

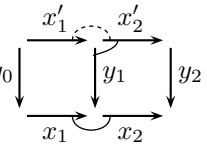
Proof. By definition, one has $s\Delta_n = s(s^*s^{**}) = (ss^*)s^{**} = \Delta_n s^{**}$, and $s\Delta_n = \Delta_n\Phi_n(s)$. □

Lemma 4.20.— *For all simple braids x_1, x_2 in B_n^+, the following are equivalent:*
(i) The sequence (x_1, x_2) is normal;
(ii) The sequence (x_1^{*-1}, x_2) is normal, i.e., the left-gcd of x_1^* and x_2 is trivial.

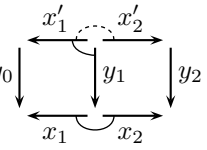
Proof. Assume (i) and let z be a common left-divisor of $x_1\backslash\Delta_n$ and x_2. Then x_1z left-divides $x_1x_1^*$, which is $\mathrm{lcm}_{\scriptscriptstyle R}(x_1, \Delta_n)$ by definition. As x_1 is simple, the latter lcm is Δ_n, which implies that x_1z is simple. On the other hand, z left-divides x_2 by hypothesis, hence x_1z left-divides x_1x_2. As (x_1, x_2) is normal, we deduce that x_1z left-divides x_1, hence that z is trivial. So (i) implies (ii).

Conversely assume (ii), and let z be a simple left-divisor of x_1x_2. As z is simple, it left-divides Δ_n, which is $x_1x_1^*$. It follows that z left-divides the left-gcd of x_1x_2 and $x_1x_1^*$, which is $x_1\gcd_{\scriptscriptstyle L}(x_2, x_1^*)$. The latter gcd is trivila, so z left-divide x_1, and (x_1, x_2) is normal. So (ii) implies (i). □

Then, once again, we shall play domino.

Lemma 4.21 (domino rule 2).— *Assume that the diagram*  *is commutative, and* (x_1, x_2) *and* (y_1^{-1}, x_2') *are normal.*[7] *Then* (x_1', x_2') *is normal as well.*

PROOF. Assume that z is simple and left-divides $x_1'x_2'$. A fortiori we have $z \preccurlyeq x_1'x_2'y_2$, hence $z \preccurlyeq y_0x_1x_2$ using the commutativity of the diagram. By Lemma 2.12, we deduce $y_0\backslash z \preccurlyeq x_1x_2$. By Corollary 3.14, $y_0\backslash z$ is simple, and, by hypothesis, (x_1, x_2) is normal. We deduce $y_0\backslash z \preccurlyeq x_1$, whence $z \preccurlyeq y_0x_1$ by Lemma 2.12, *i.e.*, $z \preccurlyeq x_1'y_1$ by commutativity of the diagram. So z left-divides both $x_1'x_2'$ and $x_1'y_1$, hence it left-divides their left-gcd, which is $x_1'\mathrm{gcd}_L(x_2', y_1)$. By hypothesis, the latter is x_1'. Therefore, (x_1', x_2') is normal. □

Lemma 4.22 (domino rule 3).— *Assume that the diagram*  *is commutative, and* (x_1^{-1}, x_2) *and* (y_1^{-1}, x_2') *are normal.*[8] *Then* $({x_1'}^{-1}, x_2')$ *is normal as well.*

PROOF. Assume that z is simple and left-divides x_1' and x_2'. A fortiori we have $z \preccurlyeq x_1'y_0$, hence $z \preccurlyeq y_1x_1$ using the commutativity of the diagram. Similarly, we have $z \preccurlyeq x_2'y_2$, hence $z \preccurlyeq y_1x_2$, whence $z \preccurlyeq y_1\mathrm{gcd}_L(x_1, x_2)$. The hypothesis $\mathrm{gcd}_L(x_1, x_2) = 1$ implies $z \preccurlyeq y_1$, whence $z \preccurlyeq \mathrm{gcd}_L(x_1', y_1)$. The hypothesis $\mathrm{gcd}_L(x_1', y_1) = 1$ implies $z = 1$, *i.e.*, the left-gcd of x_1' and x_2' is trivial, or, equivalently, $({x_1'}^{-1}, x_2)$ is normal. □

We can now put things together.

Proposition 4.23 (left-multiplication I).— *Assume that* $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ *is the symmetric normal decomposition of a braid* z*, and* s *is a simple braid. Then the normal decomposition of the braid* sz *is the sequence* $({x_d'}^{-1}, ..., {x_1'}^{-1}, y_1', ..., y_e', s_e)$ *specified in Figure 4.*

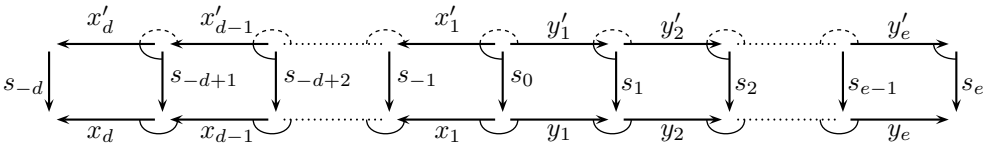


FIGURE 4. Computing the normal form of sz from that of z: starting from $s_{-d} = s$, take the normal decomposition (x_1', y_1) of y_0x_1, then the normal decomposition (x_2', y_2) of y_1x_2, and so on from left to right.

PROOF. Start from the left and use Lemmas 4.17 and 3.26 to fill the diagram of Figure 4. The top sequence is then normal by domino rules 1, 2, and 3. □

We are not yet completely done: it remains to treat the left-multiplication by a letter σ_i^{-1} or, more generally, by the inverse of a simple braid. As in the case of the Δ-normal form, it is enough to treat the case of left-multiplying by Δ_n^{-1} (why?).

Proposition 4.24 (left-multiplication II).— *Assume that* $(x_d^{-1}, ..., x_1^{-1}, y_1, ..., y_e)$ *is the symmetric normal decomposition of a braid* z *of* B_n*,. Then the normal decomposition of the braid* $\Delta_n^{-1}z$ *is the sequence* $(\Phi_n(x_d)^{-1}, ..., \Phi_n(x_1)^{-1}, {y_1^*}^{-1}, y_2, ..., y_e)$ *specified in Figure 5.*

PROOF. By construction the diagram is commutative, and it suffices to check normality at each step. As for $(\Phi_n(x_k), \Phi_n(x_{k+1}))$, its normality follows from that of (x_k, x_{k+1}) and the fact that Φ_n is an automorphism. So there only remains the case of $(y_1^*, \Phi_n(x_1))$ and of $({y_1^*}^{-1}, y_2)$.

As for the latter, (y_1, y_2) is normal by hypothesis, hence by Lemma 4.20 (direction (i) implies (ii)) so is $({y_1^*}^{-1}, y_2)$.

[7] *i.e.*, in the latter case, the left-gcd of x_2' and y_1 is trivial

[8] *i.e.*, the left-gcd of x_1 and x_2, and that of x_2' and y_1 are trivial

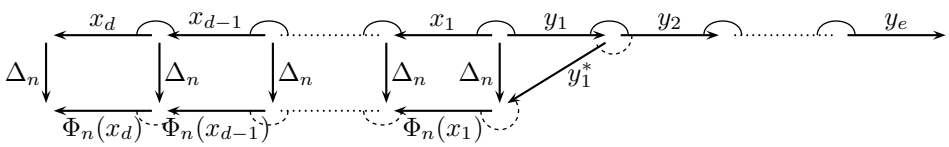


FIGURE 5. Computing the normal form of $\Delta_n^{-1}z$ from that of z.

As for the former, (y_1^{-1}, x_1) is normal by hypothesis[9], hence so is $(\Phi_n(y_1)^{-1}, \Phi_n(x_1)$ as Φ_n is an automorphism. By Lemma 4.19, $\Phi_n(y_1)$ is y_1^{**}, so the latter result states that $(y_1^{**-1}, \Phi_n(x_1))$ is normal. By Lemma 4.20 (direction (ii) implies (i)) we deduce that $(y_1^*, \Phi_n(x_1))$ is normal, which completes the proof. □

As in the case of positive braids, the existence of the incremental rule for computing the symmetric normal form implies

Corollary 4.25.— *For each n, the greedy normal form of a positive braid of length ℓ in B_n^+ can be computed in time $O(\ell^2)$.*

The existence of a symmetric normal form obeying computation rules of the above type is one of the aspects of the existence of what is called an *automatic structure* on braid groups. This is the subject of another course...

[9]remember that this just means that the left-gcd of x_1 and y_1 is trivial, so x_1 and y_1 play symmetric roles, and the normality of (y_1^{-1}, x_1) is equivalent to that of (x_1^{-1}, y_1)

Bibliography

[1] E. Artin, *Theorie der Zöpfe*, Abh. Math. Sem. Univ. Hamburg **4** (1925) 47–72.
[2] E. Artin, *Theory of Braids*, Ann. of Math. **48** (1947) 101–126.
[3] S. Bigelow, *Braid groups are linear*, J. Amer. Math. Soc. **14** (2001) 471–486.
[4] J. Birman, *Braids, Links, and Mapping Class Groups*, Annals of Math. Studies **82** Princeton Univ. Press (1975).
[5] P. Dehornoy, *Groups with a complemented presentation*, J. Pure Appl. Algebra **116** (1997) 115–137.
[6] P. Dehornoy, *A fast method for comparing braids*, Advances in Math. **125** (1997) 200–235.
[7] P. Dehornoy, *Braid-based cryptography*, Contemp. Math. **360** (2004) 5–33.
[8] P. Dehornoy, with I. Dynnikov, D. Rolfsen, B. Wiest, *Ordering Braids*, Mathematical Surveys and Monographs vol. 148, Amer. Math. Soc., (2008).
[9] E. A. ElRifai & H. R. Morton, *Algorithms for positive braids*, Quart. J. Math. Oxford **45-2** (1994) 479–497.
[10] D. Epstein, J. Cannon, D. Holt, S. Levy, M. Paterson & W. Thurston, *Word Processing in Groups*, Jones & Bartlett Publ. (1992).
[11] F. A. Garside, *The braid group and other groups*, Quart. J. Math. Oxford **20-78** (1969) 235–254.
[12] C. Kassel & V. Turaev, *Braid groups*, Grad. Texts in Math., Springer (2008).
[13] D. Krammer, *The braid group B_4 is linear*, Invent. Math. **142** (2000) 451–486.
[14] D. Krammer, *Braid groups are linear*, Ann. Math. **151-1** (2002) 131–156.
[15] http://www.math.unicaen.fr/∼tressapp/index.html.