A Fast Method for Comparing Braids

Patrick DEHORNOY

ABSTRACT. We describe a new method for comparing braid words which relies both on the automatic structure of the braid groups and on the existence of a linear ordering on braids. This syntactical algorithm is a direct generalization of the classical word reduction used in the description of free groups, and is more efficient in practice than all previously known methods.

We consider in this paper the classical braid isotopy problem, *i.e.*, the question of deciding if a given two-dimensional diagram made of a series of mutually crossing strands can be transformed into another one by moving strands but not allowing one to pass through another one. As is well-known, this problem became a question of algebra after E. Artin in the 20's has rephrased it as the word problem for a family of effectively presented groups, Artin's braid groups B_n.

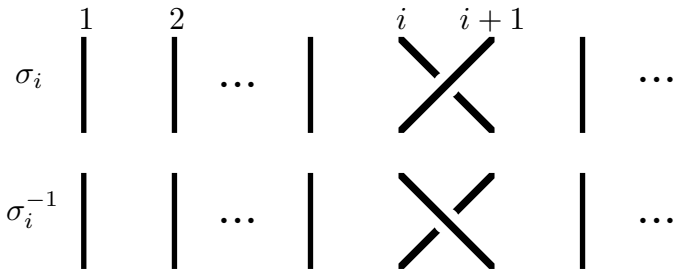
Many solutions have been described, beginning with Artin's original construction that uses the geometric idea of combing the braids to obtain a normal form for braid words and a decomposition of the groups B_n as semidirect products of free groups ([1]). The starting point for modern braid comparison method is the purely algebraic result by Garside [10] that every braid can be decomposed into a quotient of two positive braids, *i.e.*, of braids where all crossings have the same orientation. Several algorithms have been constructed: [10] itself, then [17] (*cf.* [9]), [8], [6], [16], [15]. These methods take advantage of the special form of the relations in the standard presentation of the groups B_n, mainly in terms of the geometry of the associated Cayley graph. In particular the existence of a (bi)automatic structure on B_n guarantees the existence of a quadratic isoperimetric inequality, and explains the efficiency of the practical algorithms deduced from this approach: they have a polynomial complexity with respect to the length of the braid words, even a quadratic complexity when the number of strands is fixed.

The aim of this paper is to present a new method for solving the braid isotopy problem. This method appeals to a completely new ingredient, namely the existence of an *linearly ordered* structure on the groups B_n. Such a structure was introduced recently in [5] in connection with results in the study of self-distributive operations (and, ultimately, with a problematic of set theory, *cf.* [7]). The use of the braid order will prove to be crucial in order to "pilot" our algorithm, and it explains heuristically its efficiency. Indeed the new method proves to be more efficient than the previously known ones, specially for braids with many strands: typically it enables us to compare on a microcomputer random braids that involve say 1000 crossings and any number of strands in less than one second, which seems to be (far) beyond the practical capabilities of the former methods.

The paper is organized as follows. Section 1 describes the method, which consists in iterating some reduction operation on braid words until some special form is obtained. Sections 2 and 3 establish that such reductions must always terminate and use two complementary arguments, respectively a boundedness result that follows from *algebraic* considerations (about the geometry of the Cayley graph of B_∞), and an acyclicity result that directly appeals to *order* considerations. Section 4 discusses the algorithmic aspects of the method, which in turn suggest further conjectures.

1. The geometric principle of handle reduction

As usual we shall use σ_i and σ_i^{-1} to denote the elementary braid diagrams where the stands at positions i and $(i+1)$ cross as below



Then every braid diagram is described by a (finite) concatenation of σ_i's and σ_i^{-1}'s, *i.e.*, by a braid *word* involving the letters $\sigma_i^{\pm 1}$. E. Artin has shown that two such braid words represent isotopic braid diagrams if and only if they are equivalent with respect to the least congruence $\equiv$ that satisfies, for all integers i, j,

$$\begin{cases} \sigma_i\sigma_{i+1}\sigma_i \equiv \sigma_{i+1}\sigma_i\sigma_{i+1} \\ \sigma_i\sigma_j \equiv \sigma_j\sigma_i \qquad \text{for } |i-j| \geq 2 \end{cases} \tag{1.1}$$

and $\sigma_i\sigma_i^{-1} \equiv \sigma_i^{-1}\sigma_i \equiv \varepsilon$, where ε is the nullstring. In other words, if B_n denotes the group generated by a sequence of generators σ_1, σ_2, ..., σ_{n-1} subject to the relations (1.1), then the isotopy problem for n strand braids becomes the *word problem* for the presentation (1.1) of the group B_n.

In this text it will be convenient to work with braids that involve an unbounded number of strands. This amounts to consider the direct limit B_∞ of the groups B_n (with respect to inclusion), *i.e.*, the group generated by an infinite sequence σ_1, σ_2, ..., subject to (1.1). So braid words will be arbitrary finite sequences of letters $\sigma_k^{\pm 1}$. The *braid* associated with a braid word w, *i.e.*, its class in the group B_∞, is denoted by $\overline{w}$. If the braid β is the class of the word w, we say also that w is a decomposition for β. If w is a braid word, the *length* of w is the number of letters occurring in w (hence the number of crossings in the associated braid diagram), and the *width* of w is the width of the smallest domain that includes the non-trivial part of w: formally the width of w is n if $n-2$ is the difference between the lowest and the highest indices of letters occurring in w (for instance the width $\sigma_3\sigma_6^{-1}\sigma_3^{-1}$ is 5, since this braid word involves the strands numbered 3 to 7; in particular the width of an n strand braid word is always at most n). Finally we say that a braid word is *freely reduced* if it includes no subword of the form $\sigma_i\sigma_i^{-1}$ or $\sigma_i^{-1}\sigma_i$, and we speak of *free reduction* of an arbitrary word to refer to the operation of iteratively deleting all such pairs.

Our construction will relie upon the existence of braid decompositions with a particular syntactical form. For any integer j, we shall say that the letters σ_j in a word w are *positive* occurrences of σ_j, and that the letters σ_j^{-1} are *negative* occurrences of σ_j.

Definition. The braid word w is *reduced* either if w is the nullstring, or if the *main* generator of w, defined as the generator with lower index occurring in w, occurs only positively, or only negatively.

For instance the word $\sigma_3^{-1}\sigma_2\sigma_4^{-1}\sigma_2$ is reduced, since the main generator, here σ_2, occurs only positively (no occurrence of σ_2^{-1}). On the other hand $\sigma_1\sigma_2\sigma_1^{-1}$ is not reduced, since the main generator σ_1 occurs both positively and negatively. The interest of considering reduced braid words in connection with the word problem is given by

Proposition 1.1. ([5], see also [11]) *A nonempty reduced braid word cannot be equivalent to the nullstring.*

It follows that any method that would possibly transform a braid word into an equivalent reduced braid word would automatically solve the word problem of braids: assuming that w' is a reduced braid word that is equivalent to w, w is *equivalent* to the nullstring if and only if w' *is* the nullstring. (Of course the general problem of deciding if two braid words w, w' are equivalent reduces to the problem of deciding if one braid word is equivalent to the nullstring ε since $w \equiv w'$ is equivalent to $w^{-1}w' \equiv \varepsilon$.) Now such methods do exist:

Proposition 1.2. ([5]) *Every braid admits a reduced decomposition.*

Unfortunately the original method of [5], though perfectly effective, is intractable in practice: because of a long detour involving self-distributive structures the complexity of the reduced decompositions obtained in this way is in general huge. In particular the method requires considerably increasing the *width* of the braid words, so that it leaves open the natural question of the existence of a width n reduced decomposition for every width n braid. This question was settled positively by R. Laver (unpublished work) using fine results of self-distributive algebra. However his proof is only existential and does not give an effective method. Subsequently, D. Larue described in [12] a reduction method that preserves the width and is effective. His method uses the realization of B_∞ as a group of inner automorphisms of a free group. However the complexity of this method seems to be intrinsically exponential with respect to the length of the words, so that the corresponding solution to the word problem is not efficient in practice.

The aim of this paper is to describe a new reduction method which relies on a very simple geometric idea and happens to give an extremely efficient solution to the word problem. The reason that explains this efficiency is that the method takes advantage of the *order* properties that are implicit in Propositions 1.1 and 1.2. Indeed it is easy to deduce from the latter results

Corollary 1.3. *For any braids β, β', say that $\beta < \beta'$ holds if and only if the braid $\beta^{-1}\beta'$ admits a reduced decomposition of positive type, i.e., a reduced decomposition where the main generator occurs positively. Then the relation $<$ is a linear ordering on B_∞.*

Observe that the defining property of the order $<$ forces every generator σ_j to be preponderant over all $\sigma_k^{\pm 1}$ with $k > j$ in the sense that $\beta'\sigma_j\beta'' > \beta$ holds whenever β, β' and β'' admit decompositions involving only no letter $\sigma_k^{\pm 1}$ with $k \leq j$. (It is shown in [5] that this property essentially characterizes the order $<$. See also [13] and [3] for additional properties of this order.)

So we see that any method that constructs a reduced decomposition for a braid is actually a comparison method that decides if this braid is smaller, equal or larger than the unit braid with respect to the ordering $<$. (Observe that using this additional information can considerably lower the number of comparaisons to be done if one wishes to show that more than two braids are pairwise distinct.)

From now on we consider the problem of transforming an arbitrary braid word into an equivalent reduced word. Our method is a generalization of the free reduction which deletes the pairs xx^{-1} or $x^{-1}x$ in any group presentation. The latter case corresponds formally to the trivial case of width 2 braids, and gives a valuable intuition. However the extension of free reduction we shall introduce heavily depends on the geometry of braids: this explains its efficiency, but also dismisses the hope that the method be possibly generalized to a much larger class of groups.

By definition a braid word is *not* reduced when it contains some alternation of the form $\sigma_i^{\pm 1} \ldots \sigma_i^{\mp 1}$, where σ_i is the main generator. If we consider occurrences of $\sigma_i^{\pm 1}$ that are as close as possible, the intermediate factor will contain only generators $\sigma_k^{\pm 1}$ with $k > i$. Geometrically this corresponds to the fact that the $i+1$-th strand forms a (left) *handle* as in Figure 1.1.

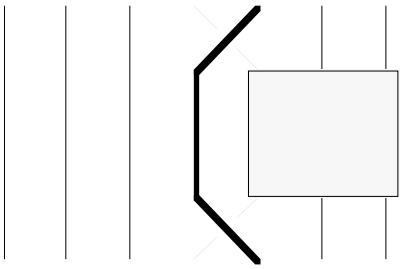


Figure 1.1: A handle

Our method will consist in eliminating the handles as above in order to eventually obtain a reduced word. However it will in general be necessary to consider also similar handles that involve any generator (and not only the main one). So we take

Definition. (Figure 1.2) A σ_j*-handle* is a braid word of the form $\sigma_j^e v \sigma_j^{-e}$, where e is $+1$ or -1 and the word v contains only generators $\sigma_k^{\pm 1}$ with $k < j-1$ or $k > j$. A *main* handle of w is a subword of w that is a σ_i-handle, where σ_i is the main generator of w.

It is easy to imagine geometric transformations that eliminate one handle. For instance Figure 1.3 illustrates the "coarse" method that consists in replacing a handle of the form $\sigma_j^e v \sigma_j^{-e}$ by the equivalent word

$$\sigma_{j+1}^{-e}\sigma_{j+2}^{-e}\ldots\sigma_{n-1}^{-e}\ \theta_j(v)\ \sigma_{n-1}^{e}\ldots\sigma_{j+2}^{e}\sigma_{j+1}^{e},$$

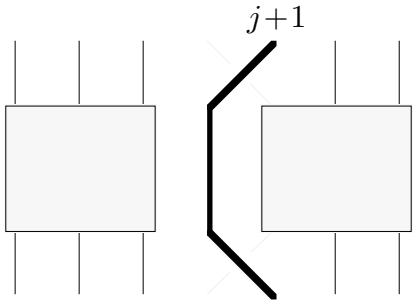


Figure 1.2: A (general) σ_j-handle

where n is the width of the considered braid word, and θ_j denotes the left shift of the generators σ_k with $k > j$, *i.e.*, the partial homomorphism that maps σ_k to σ_{k-1} for $k > j$ and preserves σ_k for $k < j$.

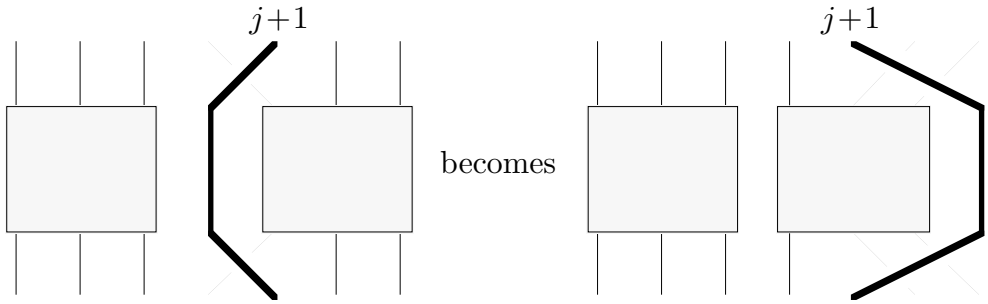


Figure 1.3: Coarse reduction of a handle

In the sequel we shall consider a slightly more careful handle reduction method, namely the "local" reduction where, instead of moving the guilty $j+1$-th strand to the extreme right of the diagram, we let it skirt on the right the "next" crossings, *i.e.*, the ones at position $j+1, j+2$ (so the latter crossings are shifted left in the process). This transformation is illustrated in Figure 1.4 (the orientations of the crossings at position $j+1, j+2$, which can be arbitrary, have not been specified), and it amounts to replacing a handle of generic form

$$\sigma_j^e \; v_0 \; \sigma_{j+1}^{d_1} \; v_1 \ldots \; v_{m-1} \; \sigma_{j+1}^{d_m} \; v_m \; \sigma_j^{-e}, \tag{1.2}$$

where $v_0, \ldots, v_q$ contain no $\sigma_k^{\pm 1}$ with $j-1 \leq k \leq j+1$, with the equivalent word

$$v_0 \; \sigma_{j+1}^{-e} \sigma_j^{d_1} \sigma_{j+1}^{e} \; v_1 \ldots \; v_{m-1} \; \sigma_{j+1}^{-e} \sigma_j^{d_m} \sigma_{j+1}^{e} \; v_m. \tag{1.3}$$

In other words we apply *in the handle* the alphabetical homomorphism $\phi_{j,e}$ defined by

$$\phi_{j,e} \; : \; \begin{cases} \sigma_j^{\pm 1} \mapsto \varepsilon \\ \sigma_{j+1}^{\pm 1} \mapsto \sigma_{j+1}^{-e} \sigma_j^{\pm 1} \sigma_{j+1}^{e} \\ \sigma_k^{\pm 1} \mapsto \sigma_k^{\pm 1} \qquad \text{for } k \neq j, j+1. \end{cases} \tag{1.4}$$

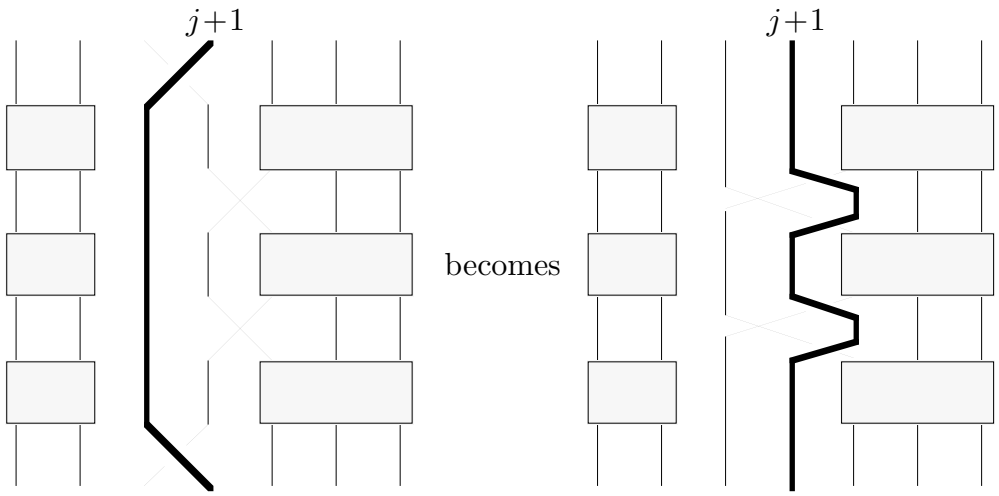


Figure 1.4: Local reduction of a handle

We see at once that applying local reduction (or coarse reduction) will delete one handle, but at the expense of possibly creating (many) new ones. So it is not surprising that the naive approach consisting in simply reducing the main handles until some reduced word is reached does not work. Indeed consider the word $w = \sigma_1\sigma_2\sigma_3\sigma_2^{-1}\sigma_1^{-1}$. There is only one main handle in w, namely w itself, and applying local reduction to w gives the word $\sigma_2^{-1}w\sigma_2$. It follows that repeated reductions will give the words $\sigma_2^{-k}w\sigma_2^k$, none of which is reduced.

The problem in the previous trivial example is that reduction is applied to a handle whose median factor (the one between the first and the last occurrences of the main generator, here $\sigma_2\sigma_3\sigma_2^{-1}$) is not reduced, and, more precisely, that it contains a σ_2-handle. The main result of this paper is that this obstruction is the only possible one, *i.e.*, that handle reduction will *always* lead to reduced words in finitely many steps provided that reducing handles of the above type is avoided.

Definition. i) A σ_j-handle $\sigma_j^e v\sigma_j^{-e}$ is *permitted* if it includes no σ_{j+1}-handle, *i.e.*, if at least one of σ_{j+1}, σ_{j+1}^{-1} does not occur in v.

ii) The word w' is deduced from w *using one step of handle reduction*, or H-reduction, if w' is obtained from w by reducing a permitted σ_j^e-handle of w (using local reduction), *i.e.*, applying in that handle the alphabetical homomorphism $\phi_{j,e}$ of (1.4).

Of course we shall say that w' is deduced from w using N steps of handle reduction from w if there exists a length N sequence of reductions from w to w', *i.e.*, a sequence $w_0 = w, w_1, \ldots, w_N = w'$ such that any term is deduced from the previous using one step of handle reduction. Observe that, with the notations of formula (1.2), the hypothesis that the handle is permitted means that all exponents $d_1, \ldots, d_q$ have a common value.

Example 0.1. Let us come back to the above word $w = \sigma_1\sigma_2\sigma_3\sigma_2^{-1}\sigma_1^{-1}$. The main handle w is not permitted, but the σ_2-handle $\sigma_2\sigma_3\sigma_2^{-1}$ is, and the word $\sigma_1\sigma_3^{-1}\sigma_2\sigma_3\sigma_1^{-1}$ is deduced from w using reduction. The latter word is itself a (main) handle, and it is now a permitted handle since the alternation of σ_2's has been corrected, so reduction is possible, and gives $\sigma_3^{-1}\sigma_2^{-1}\sigma_1\sigma_2\sigma_3$, a reduced word that is terminal for reduction.

A few experiments will show that the situation is in general much less simple. Nevertheless we have

Theorem 1.4. *Handle reduction is noetherian (or well-founded): any sequence of reductions has to be finite. More precisely, if w is a braid word of length ℓ and width n, the length of any sequence of handle reductions from w is bounded above by $2^{n^4\ell}$.*

The words that are terminal with respect to handle reduction are certainly reduced, so the previous result gives a new proof of Proposition 1.2. Actually we obtain a little more.

Definition. The braid word w is *fully reduced* if any two letters σ_j, σ_j^{-1} in w are separated by at least one letter $\sigma_{j-1}^{\pm 1}$.

Lemma 1.5. *For any braid word w the following are equivalent:*
i) w is fully reduced;
ii) w contains no handle;
iii) w is terminal with respect to handle reduction.

Proof. Points (i), (ii) are equivalent by definition of a handle, and they obviously imply (iii). For the converse implication, observe that the leftmost handle of a word (in terms of the position of the last letter of this handle) must always be a permitted one: indeed a σ_j-handle is not permitted just in case there exists a σ_{j+1}-handle that is nested in it, which forbids the previous to be the leftmost handle of w. ■

So from Theorem 1.4 we deduce

Corollary 1.6. *Every braid in B_n admits a fully reduced decomposition of width n.*

2. Termination of handle reduction (I): boundedness

There seems to be no obvious reason that forces any sequence of reductions to eventually reach a fully reduced word, or simply a reduced word. In particular it is clear from the definition of reduction that the length of the words will increase in general. On the other hand it is also not clear why the sequence could not simply enter a loop and therefore continue periodically without ever terminating. Our proof of Theorem 1.4 will consist of two ingredients that more or less solve these two problems, namely a *boundedness* argument, which shows that the words that can be deduced from w using reduction are, in some sense, not more complicated than w itself (even if their length may be larger), and an *acyclicity* argument, which shows that repetitions are forbidden and gives actual termination.

Our starting point for studying reduction will be to connect it with more atomic transformations which will have a simple counterpart in the Cayley graph of B_∞. To this end we appeal to some results of [6] about the transformation of arbitrary braid words into quotients of two positive braid words.

Definition. The word w' is deduced from w *using one step of R-reduction* if w' is obtained from w by replacing some subword of the form $\sigma_i^{-1}\sigma_j$ with the corresponding (equivalent) word $f(\sigma_j, \sigma_i)f(\sigma_j, \sigma_i)^{-1}$, where f is the mapping defined by

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

In this definition the letter R stands for "right": right reduction corresponds to moving *right* the negative generators in order to gather them at the end of the word, while positive generators are gathered at the beginning. Of course, like for handle reduction, we shall iterate R-reduction and consider words that can be deduced from an initial word in any number of R-reduction steps.

Example 0.2. In this section we shall illustrate our constructions using the word $w_0 = \sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_1$. Only one word can be deduced from w_0 using one step of R-reduction, namely $\sigma_1\sigma_2^{-1}\sigma_2\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_3\sigma_1$. Similarly $\sigma_1^2\sigma_2^{-1}\sigma_1^{-1}\sigma_3\sigma_1$ and $\sigma_1\sigma_2^{-1}\sigma_2\sigma_1\sigma_3\sigma_1^{-1}\sigma_1$ can be deduced in two steps, *etc.*We can check that *any* sequence of R-reductions from w_0 ends after five steps in the word $\sigma_1^2\sigma_3\sigma_2\sigma_3^{-1}\sigma_2^{-1}$, which is R-irreducible since no negative letter precedes any positive one. Figure 2.1 illustrates R-reduction in the Cayley graph. (We recall that the Cayley graph of B_∞ is an oriented graph with labelled arrows such that the set of vertices is B_∞, and there is a σ_i-labelled arrow between from the vertex β to the vertex β' if and only if β' is equal to $\beta\sigma_i$.) Observe that, if the arrows of the Cayley graph are always given a left-to-right orientation, R-reduction always transforms a word into another one such that the path associated with the latter lies on the right of the path associated with the initial word.

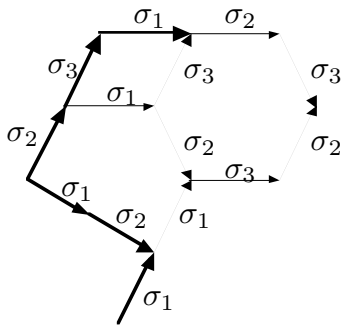


Figure 2.1: R-reduction of $\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_1$

It is clear that, if w' is deduced from w using R-reduction, then w and w' are equivalent, and it is not hard to see that at most one R-irreducible word can be deduced from a given word using R-reduction. It is less obvious that R-reduction has to terminate, *i.e.*, that one reaches an R-irreducible word of the form uv^{-1} with u, v positive. This however is always true, mainly because of the existence of an automatic structure on the group B_n ([9]):

Lemma 2.1. ([6]) *For every braid word w, there exists unique positive braid words $N_R(w)$ and $D_R(w)$ (the right numerator and denominator of w) such that the word $N_R(w)D_R(w)^{-1}$ can be deduced from w using R-reduction. Moreover, if the length of w is ℓ and its width is n, the length of $N_R(w)D_R(w)^{-1}$ is bounded by $\frac{1}{2}n(n-1)\ell$.*

For instance in the above example of w_0, we find that the right numerator and denominator are respectively $\sigma_1^2\sigma_3\sigma_2$ and $\sigma_2\sigma_3$. We refer to [6] for the properties of R-reduction (see also [16] where a similar notion was introduced independently). In particular we can show that two *positive* braid words w, w' are equivalent if and only if the nullstring can be deduced from $w^{-1}w'$ by R-reduction.

Now the braid relations (1.1) are invariant under reversing the order of the letters, so that we can immediately transpose the above notions into their *left* counterparts: we say that w' is deduced from w using one step of *L-reduction* if w can be transformed into w' by replacing some subword of the form $\sigma_i\sigma_j^{-1}$ by the corresponding pattern $g(\sigma_i,\sigma_j)^{-1}g(\sigma_j,\sigma_i)$, where g is the mapping defined by

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

Figure 2.2 below illustrates L-reduction from the word w_0. With the same graphical conventions as above, L-reduction replaces paths in the Cayley graph with new paths on their left. We shall introduce $N_{\scriptscriptstyle L}(w)$, the left numerator of w, and $D_{\scriptscriptstyle L}(w)$, the left denominator of w, which are the unique positive words such that $D_{\scriptscriptstyle L}(w)^{-1}N_{\scriptscriptstyle L}(w)$ can be deduced from w using L-reduction. They exist by the analog of Lemma 2.1. In the case of w_0, we read that the left numerators and denominators are respectively $\sigma_2^2\sigma_3\sigma_1$ and $\sigma_1\sigma_2$.

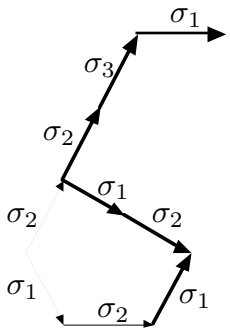


Figure 2.2: L-reduction of $\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_1$

We shall also use a notation for the refinements of braid equivalence that correspond to operating substitutions only on positive, or on negative, subwords.

Definition. The braid word w' is deduced from w using *P-equivalence* (*resp. N-equivalence*) if w can be transformd into w' by replacing finitely many times some *positive* subword by an equivalent positive subword (*resp.* some *negative* subword by an equivalent negative subword).

The interest in introducing these transformations here is that we can express handle reduction in terms of the latter:

Lemma 2.2. *Each step of H-reduction can be decomposed into a finite sequence of steps each of which is either an R- or an L-reduction, or a P- or an N-equivalence.*

Proof. The question is to prove that the transformation of

$$\sigma_j^e\ v_0\ \sigma_{j+1}^d\ v_1 \dots\ v_{m-1}\ \sigma_{j+1}^d\ v_m\ \sigma_j^{-e}, \tag{2.1}$$

(where all letters in v_0, ..., v_m are $\sigma_k^{\pm 1}$ with $|k-j| \geq 2$) into

$$v_0\ \sigma_{j+1}^{-e}\sigma_j^d\sigma_{j+1}^e\ v_1 \dots\ v_{m-1}\ \sigma_{j+1}^{-e}\sigma_j^d\sigma_{j+1}^e\ v_m. \tag{2.2}$$

can be decomposed into a series of "microsteps" of the above types. Assume for instance that e is $+1$ and d is -1. Then reduction can be done by moving right the initial σ_j. First transforming $\sigma_j v_0$ into $v_0\sigma_j$ can be made by a sequence of L-reductions and P-equivalences. Indeed by hypothesis v_0 contains only generators $\sigma_k^{\pm 1}$ with $|k-j| \geq 2$: for such k transforming $\sigma_j\sigma_k$ into $\sigma_k\sigma_j$ is a P-equivalence, and transforming $\sigma_j\sigma_k^{-1}$ into $\sigma_k^{-1}\sigma_j$ is an L-reduction. Then we find the pattern $\sigma_j\sigma_{j+1}^{-1}$, which becomes $\sigma_{j+1}^{-1}\sigma_j^{-1}\sigma_{j+1}\sigma_j$ by an L-reduction. So, at this point, we have transformed the initial word into

$$v_0\ \sigma_{j+1}^{-1}\sigma_j^{-1}\sigma_{j+1}\sigma_j\ v_1\ \sigma_{j+1}^{-1}\ v_2 \dots\ v_{m-1}\ \sigma_{j+1}^{-1}\ v_m\ \sigma_j^{-1}. \tag{2.3}$$

After m such sequences of reductions, and a last L-reduction to delete the final pattern $\sigma_j\sigma_j^{-1}$, we reach the form (2.2), as we wished. The argument is similar in the case $e = -1$, $d = 1$, with N-equivalences instead of P-equivalences. In the case when the exponents e and d have the same sign, we use R-reduction in order to move left the final generator σ_j^{-e} in a symmetric way. ■

It follows from this result that the set of all words that can be deduced by reduction from a given word w is included in the closure of the word w under R- and L-reduction and P- and N-equivalence. We have mentioned that the closure of a word under R-reduction alone, or under L-reduction alone, is certainly finite. Things are not so simple when both operations are allowed simultaneously, as shows the sequence

$$\sigma_1\sigma_2^{-1},\quad \sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_1,\quad \sigma_2^{-1}\sigma_2\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_1,\quad \sigma_2^{-1}\sigma_2\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_1\sigma_1^{-1}\sigma_1, \dots$$

which is obtained by an alternation of L- and R-reductions. However we can see that the above words, yet their lengths are unbounded, remain traced in some *finite* region of the Cayley graph of B_∞.

For any set of braid words S we can consider the subgraph of the Cayley graph of B_∞ (of which a definition was recalled above) made by the paths associated with the elements of S starting at the fixed point 1. Then we have a natural notion of a word *traced* in this subgraph, *i.e.*, a word such that there exists a path in the subgraph whose sucessive labels are the letters of the considered word (when an edge is crossed backwards it contributes the inverse of its label). Formally we take

Definition. (Figure 2.3) Assume that S is a set of positive braid words, and u belongs to S. The braid word w is *traced in S from u* if there exists in the Cayley graph of S a path labelled w that starts from the vertex $\overline{u}$, *i.e.*, if, for every prefix of w of the form

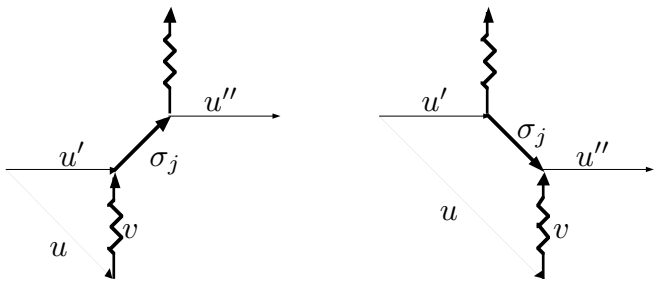


Figure 2.3: Word traced in S

$v\sigma_j$ (*resp.* $v\sigma_j^{-1}$) there exist positive words u', u'' such that $u'\sigma_j u''$ belongs to S and uv is equivalent to u' (*resp.* to $u'\sigma_j$).

Observe that there are always infinitely many words traced in any set S that contains at least one nonempty word: if the word σ_j is traced in S, so are all words $(\sigma_j\sigma_j^{-1})^k$. Nevertheless we shall see in Section 3 how the fact that some words are traced in a finite set can be used as a strong boundedness hypothesis. Presently we observe that, provided that the set S is regular enough, the set of the words traced in S verifies good closure properties.

Lemma 2.3. *Assume that S is the set of all positive decompositions of some positive braid. Then the set of the words traced in S from some given point is closed under R- and L-reductions, and under P- and N-equivalence.*

Proof. Consider first the case of R-reduction (Figure 2.4). Assume that some word $v\sigma_i^{-1}\sigma_j v'$ is traced in S from u. We have to show that the word $vf(\sigma_j,\sigma_i)f(\sigma_j\sigma_i)^{-1}v'$ is also traced in S from u. Now the hypothesis means that there exist positive braid words u', u_1'' and u_2'' such that both $u'\sigma_i u_1''$ and $u'\sigma_j u_2''$ belong to S, and $u'\sigma_i$ is equivalent to uv. By hypothesis the positive words $u'\sigma_i u_1''$ and $u'\sigma_j u_2''$ are equivalent, and so are the words $\sigma_i u_1''$ and $\sigma_j u_2''$. By [10] (or [6]) we know that this implies the existence of a positive word u'' satisfying

$$u_1'' \equiv f(\sigma_j,\sigma_i)u'' \qquad \text{and} \qquad u_2'' \equiv f(\sigma_j,\sigma_i)u''.$$

This shows that the word $f(\sigma_j,\sigma_i)f(\sigma_i,\sigma_j)^{-1}$ is traced in S from uv, which is exactly what we need.

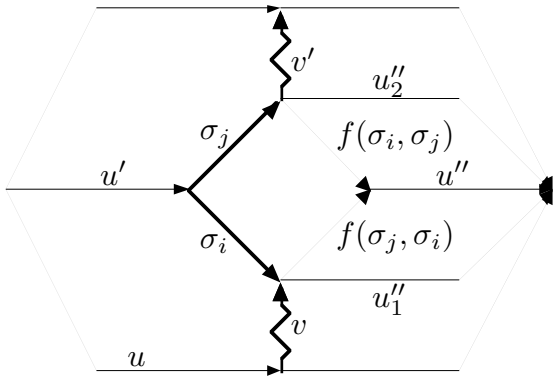


Figure 2.4: Closure of words traced in S under R-reduction

The argument is of course symmetric for L-reduction. Finally the case of P-equivalence is trivial: if, for instance, $v\sigma_i\sigma_{i+1}\sigma_i v'$ is traced in S from u, there exist positive words u' and u'' such that $u'\sigma_i\sigma_{i+1}\sigma_i u$" belongs to S and uv is equivalent to u'. Now $u'\sigma_{i+1}\sigma_i\sigma_{i+1}u$" belongs to S as well, and this shows that $v\sigma_{i+1}\sigma_i\sigma_{i+1}v'$ is still traced in S. The case of N-equivalence is similar and just corresponds to crossing the arrows with reverse orientation. ■

Then we conclude from Lemma 2.1 that, under the hypotheses of Lemma 2.2, the set of all words traced in S is closed under H-reduction. So we are left with the question of finding, for a given initial word w, a positive braid β such that w is traced in the set of the positive decompositions of β. This is easy using the notions of right and left numerators and denominators we have mentioned above.

Definition. Let w be any braid word. Then $S(w)$ is the set of all positive braid words that are equivalent to the word $D_L(w)N_R(w)$.

Example 0.3. Consider again the word w_0. The word $D_L(w_0)N_R(w_0)$ is $\sigma_1\sigma_2\sigma_1^2\sigma_3\sigma_2$, and we find that $S(w_0)$ contains 8 words as shown in Figure 2.5. In this simple example the Cayley graph of $S(w)$ happens to be a planar graph, but this need not be true in general. Observe that the Cayley graph of $S(w_0)$ is a strict extension of the Cayley graph obtained by simply closing w_0 under R- and L-reduction.

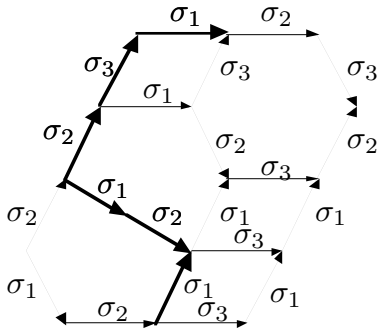


Figure 2.5: The set $S(\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_1)$

It is clear that the set $S(w)$ is finite, and we obtain the following trivial bound on its size.

Lemma 2.4. *Assume that the braid word w has length ℓ and width n. Then the set $S(w)$ is finite with at most $(n-1)^{n^2\ell}$ elements.*

Proof. By Lemma 2.1 the length of the word $D_L(w)N_R(w)$ is at most $n^2\ell$. Now for a positive braid word of length L there are only finitely many positive equivalent braid words, all with the same length L. More precisely, if only $n-1$ different generators may be used, the number of positive words with length L is certainly at most $(n-1)^L$. ■

The last step of our argument is

Lemma 2.5. *Let w be any braid word. Then w is traced in set $S(w)$ from $D_L(w)$.*

Proof. We use induction on the length of w. The result is obvious if w is the nullstring. So assume that the result is proved for w, and consider the case of the words $w\sigma_k^{\pm 1}$. Assume first that w' is $w\sigma_k$. By construction $D_L(w')$ is $D_L(w)$, and $N_R(w')$ is $N_R(w)v$ for some positive word v, namely the right numerator of $D_R(w)^{-1}\sigma_k$ (Figure 2.6). It follows that uv belongs to $S(w')$ for every word u in $S(w)$, and, therefore, every word traced in $S(w)$ from $D_L(w)$ is also traced in $S(w')$ from $D_L(w)$. This applies in particular to w itself. So it only remains to consider the case of the final letter σ_k of w'. Now $D_L(w')w$ is equivalent by construction to $N_L(w)$, and the equivalence

$$N_L(w)\ \sigma_k\ D_R(w)\ \equiv\ D_L(w')\ N_R(w')$$

shows that this final letter σ_k also satisfies the defining condition for a word traced in $S(w')$.

Assume now that w' is $w\sigma_k^{-1}$. There exists a positive word v (namely the left denominator of $N_L(w)\sigma_j^{-1}$) such that $D_L(w')$ is $vD_L(w)$, while $N_R(w')$ is $N_R(w)$. So now the word vu belongs to $S(w')$ for every word u of $S(w)$. It follows that w is traced in $S(w')$ from $vD_L(w)$, which is $D_L(w')$. Again it remains to consider the final letter σ_k^{-1} of w'. Now $D_L(w')w'$ is equivalent to $N_L(w')$, and the equivalence

$$N_L(w')\ \sigma_k\ D_R(w)\ \equiv\ D_L(w')\ N_R(w')$$

gives the conclusion. ■

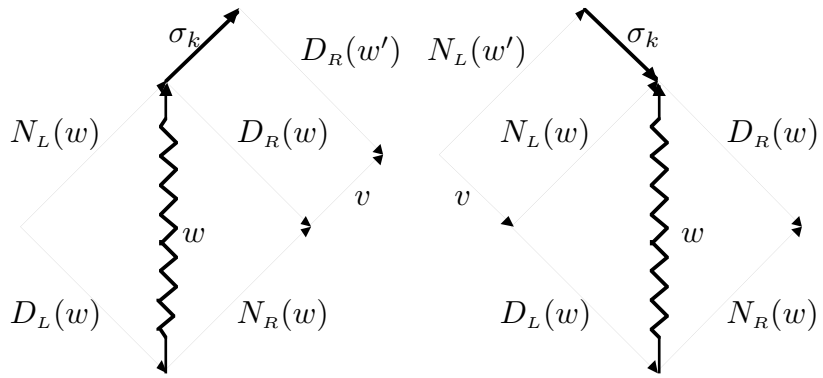


Figure 2.6: The sets $S(w\sigma_k)$ and $S(w\sigma_k^{-1})$

Gathering the previous lemmas gives the main result of this section:

Proposition 2.6. *Let w be any braid word. Then all words w' that can be deduced from w using handle reduction are traced in $S(w)$ (from $D_L(w)$).*

This result gives the boundedness property we were looking for: it means that all words that are deduced from a given word using reduction have to remain traced in some *finite* region of the Cayley graph of B_∞, yet for the moment we have absolutely no bound on their length.

We finish this section with a corollary that will be useful in the sequel:

Lemma 2.7. *Assume that the word w' is obtained from the word w using handle reduction (or, more generally R-, L-reduction and P-, N-equivalence). Then every word traced in $S(w')$ (from $D_L(w')$ is also traced in $\Sigma(w)$ from $D_L(w)$.*

Proof. It suffices to consider the case when w' is obtained using one step of R- or L-reduction, or one P- or N- equivalences. In the three latter cases, the results of [6] show that the words $D_{L}(w')$ and $N_{R}(w')$ are respectively equivalent to $D_{L}(w)$ and $N_{L}(w)$, so the sets $S(w')$ and $S(w)$ merely coincide. Now assume that w' is obtained using one step of R-reduction from w. Again by [6] we know that there exists a positive word u satisfying

$$D_{L}(w) \equiv u\ D_{L}(w) \qquad \text{and} \qquad N_{L}(w) \equiv u\ N_{L}(w').$$

This implies that a positive word v belongs to $S(w')$ if and only if the positive word uv belongs to $S(w)$. Hence any word traced in $S(w')$ from some point v is also traced in $S(w)$ from uv. ■

3. Termination of handle reduction (II): acyclicity

We have so far used the *algebraic* properties of the group B_{∞}, mainly the quadratic isoperimetric inequality that causes R- and L-reductions to terminate in a quadratic number of steps. The above arguments have shown that all words deduced using reduction from a given word remain in some finite region of the Cayley graph. This gives no informtion about the way reduction possibly *progresses* toward the final form. In particular we still have no argument to prove that cycles are impossible in reduction sequences. The latter property will follow from the existence of the *ordered* structure provided by Corollary 1.3.

The main idea is as follows. Let us introduce, for any braid word w of length ℓ, a *characteristic function* $\widehat{w}$ whose domain is the integer interval $(1, \ell)$ and which maps k to the class of the length k prefix of w in B_{∞}. Because B_{∞} equipped with the order $<$ of Corollary 1.3 is a dense linear order without endpoints, hence is isomorphic to the rationals, we can think of $\widehat{w}$ as a numerical function and represent its graph as in Figure 3.1 which corresponds to our former example w_0. (Of course the isomorphism between B_{∞} and the rationals concerns only the order, and *not* the algebraic structure: the "height" of the generators in the diagram must vary.) Now a σ_j-handle beginning with a positive letter (*resp.* a negative one) in the braid word w gives a hill (*resp.* a vail) in the graph of $\widehat{w}$, and we shall see that reduction corresponds to razing, or, at least, smoothing such hills and vails, so that the graphs of the words that appear in a reduction sequence are, in some sense, smoother and smoother. This is essentially why cycles in reduction are impossible. However, at least because the braid order is dense, the acyclicity phenomenon is not sufficient in itself to prove termination, and we shall have to marry it in a convenient way with the results of Section 2 in order to conclude.

Let us now turn to the precise argument. The point is to study how handles are transformed along sequences of reductions: we shall show that, if w' is obtained from w by handle reduction, then (most of) the handles in w' are in some sense the *heirs* of handles that were already present in w, and, moreover, that the heirs of a given handle can be reduced only a finite number of times, bounded by some constant that depends only on the initial word.

In order to describe the heiring phenomenon, we shall consider the *positions* of the letters

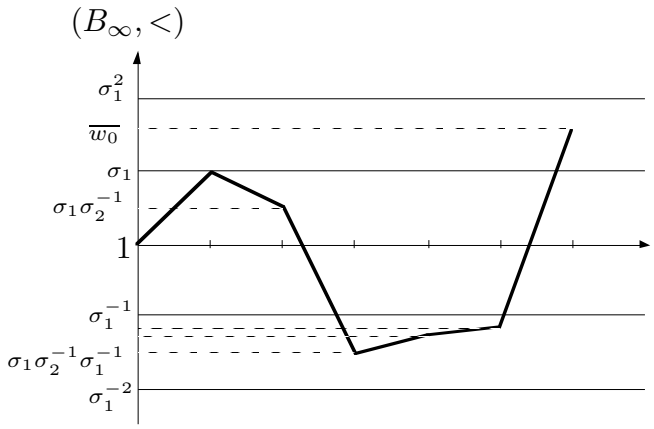


Figure 3.1: The characteristic function of $w_0 = \sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2\sigma_3\sigma_1$

in braid words: we say that x is a position of the letter $\sigma_k^{\pm 1}$ in w just to express that the x-th letter of w (starting from the left) is that letter.

Definition. The integer x is a *critical* position of the letter σ_k^e in w ($e = \pm 1$) if σ_k^e is the x-th letter of w and, in addition, some σ_k-handle of w begins at this position (we do *not* require that this handle be permitted).

Example 0.4. In the word $\sigma_1\sigma_2^2\sigma_3^{-1}\sigma_1^{-1}\sigma_3$ there are two handles, and therefore two critical positions: 1 is a critical position of σ_1, and 4 is a critical position of σ_3^{-1}.

We fix now for a while a pair of words (w, w') such that w' is obtained from w by reducing exactly one σ_j-handle. Our aim is to establish a correspondence between the critical positions in w and the critical positions in w'. We assume that the σ_j-handle that is reduced from w to w' corresponds to the positions p and q in w, and, in this case, we say that p is the *active* critical position in the reduction of w to w'. We write r for the position of $\sigma_j^{\pm 1}$ in w that immediately precedes p (if it exists), and s_1, …, s_m for the positions of $\sigma_{j+1}^{\pm 1}$ between p and q in w (if they exists). So w can be written as

$$\begin{array}{ccccccccccccc} & & r & & p & & s_1 & & & & s_m & & q \\ & & \downarrow & & \downarrow & & \downarrow & & & & \downarrow & & \downarrow \\ w & = & (w_1\ \sigma_j^b)\ u_1 & & \sigma_j^e & & v_1\ \sigma_{j+1}^d & & v_2\ \ldots\ v_{m-1} & & \sigma_{j+1}^d v_m & & \sigma_j^{-e} u_2\ (\sigma_j^c\ w_2) \end{array} \qquad (3.1)$$

where b, c, d, e are ± 1 and there is no $\sigma_j^{\pm 1}$ in u_1 and u_2, and no $\sigma_{j-1}^{\pm 1}$, $\sigma_j^{\pm 1}$, or $\sigma_{j+1}^{\pm 1}$ in v_1, …, v_m. By definition of handle reduction, the word w' is then

$$w' \ = \ (w_1\ \sigma_j^b)\ u_1 \quad v_1\ \sigma_{j+1}^{-e}\sigma_j^d\sigma_{j+1}^e\ v_2\ \ldots\ v_{m-1}\ \ \sigma_{j+1}^{-e}\sigma_j^d\sigma_{j+1}^e\ v_m \quad u_2\ (\sigma_j^c\ w_2) \qquad (3.2)$$

Let ℓ be the length of the word w, and ℓ' be the length of w'. We define a mapping

$$h \ : \ \{1, \ldots, \ell\} \setminus \{p, q\} \ \rightarrow \ \{1, \ldots, \ell'\}$$

as follows:

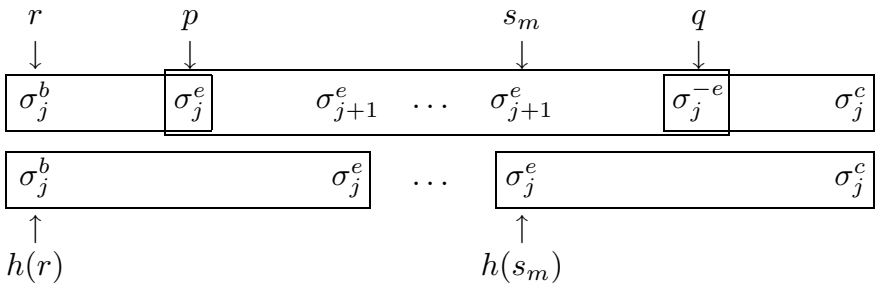
$$h(x) = \begin{cases} x & \text{for } x < p, \\ x + 2(t-1) & \text{for } x = s_t,\ t = 1, \ldots, m, \\ x + 2t - 1 & \text{for } s_{t-1} < x < s_t,\ t = 1, \ldots, m, \text{ with } s_0 = p \text{ and } s_{m+1} = q, \\ x + 2(m-1) & \text{for } x > q. \end{cases}$$

Thus $h(x)$ is the position where the letter at position x in w is copied in w'. More precisely, if x is a position of $\sigma_k^{\pm 1}$ in w, then $h(x)$ is a position of the same letter in w', excepted for the letters σ_{j+1}^d between p and q, which become σ_j^d. The natural idea is to say that $h(x)$ is in w' the heir of x, and to observe that h nearly induces a one-to-one mapping of the critical positions in w onto the critical positions in w'. However this obvious construction has to be improved in the neighbourhood of the active position, and the actual heiring function H we shall use in the sequel will be constructed from h using some additional elements.

Let us first consider the critical positions of $\sigma_k^{\pm 1}$ for $k \neq j, j+1$. Assume that x is such a critical position in w. Then $h(x)$ is a position of the same letter in w'. Moreover the fact that x is critical means that there exists $y > x$ such that y is a position of $\sigma_k^{\mp 1}$ in w and no $\sigma_k^{\pm 1}$ or $\sigma_{k-1}^{\pm 1}$ occurs in w between x and y. It follows that $h(y)$ is a position of $\sigma_k^{\mp 1}$ in w', and that no $\sigma_k^{\pm 1}$ or $\sigma_{k-1}^{\pm 1}$ occurs in w' between $h(x)$ and $h(y)$: this is clear if k is not $j+2$ since, in this case, h inducess an order-preserving bijection between the positions of the letters $\sigma_k^{\pm 1}$ and $\sigma_{k-1}^{\pm 1}$ in w and w'. If k is $j+2$, then $k-1$ is $j+1$, and some letters $\sigma_{j+1}^{\pm 1}$ are modified from w to w'. But observe that the involved modification consists in replacing σ_{j+1}^d with $\sigma_{j+1}^{-e}\sigma_j^d\sigma_{j+1}^e$, and therefore does not change the possible existence of some letter $\sigma_{j+1}^{\pm 1}$ between two letters $\sigma_{j+2}^{\pm 1}$. Hence $h(x)$ is a critical position of $\sigma_k^{\pm 1}$ in w'. Conversely, the same argument shows that, if x' is a critical position of $\sigma_k^{\pm 1}$ in w', then necessarily there exists x such that x' is $h(x)$ and x is a critical position of $\sigma_k^{\pm 1}$ in w. So, if we define, for such positions x in w, $H(x)$ to be $h(x)$, then the critical positions of $\sigma_k^{\pm 1}$ in w' are exactly the heirs (*i.e.*, the images under H) of the similar critical positions in w.

We consider now the critical positions of $\sigma_j^{\pm 1}$. If x is such a position in w, and x is none of p, q, r, then $h(x)$ is again a critical position of the same letter in w', for nothing has been changed between x and the next position of $\sigma_j^{\pm 1}$. For such positions x, we take $H(x) = h(x)$. It remains to look at p, q and r, and at the possible handles associated with these positions. The main point is that the number of σ_j-handles will never increase from w to w', so that we shall be able to complete the definition of the function H in order to make it always surjective (for critical positions of $\sigma_j^{\pm 1}$). It will suffice to consider three cases according to the relative signs of e and d.

Case 1: $m \geq 1$ and $d = e$, *i.e.*, σ_{j+1}^e occurs in the σ_j-handle that is reduced.
The behaviour of the handles can be displayed as follows (the boxed patterns correspond to the possible handles: that they are actual handles depends on the signs of b, c, e, and of the possible letters $\sigma_{j-1}^{\pm 1}$ before p and after q):



At most two handles appear in w', and we see that, if we complete the definition of the mapping H with

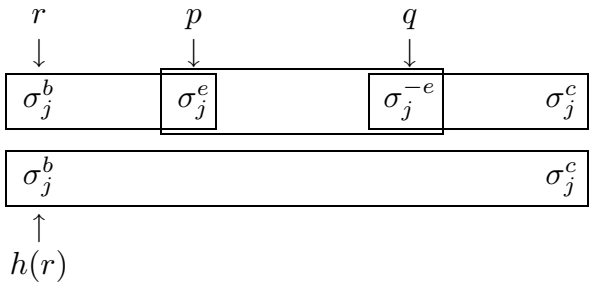
$$H(r) = h(r) = r \qquad \text{if} \qquad r \text{ is critical in } w', \text{ and } b = -e,$$

and

$$H(p) = h(s_m) \qquad \text{if} \qquad h(s_m) \text{ is critical in } w' \text{ and } c = -e,$$

($H(p)$, $H(q)$ and $H(r)$ being undefined in all other cases), then it will be true that any critical position of $\sigma_j^{\pm 1}$ in w' is the heir of some critical position of that letter in w.

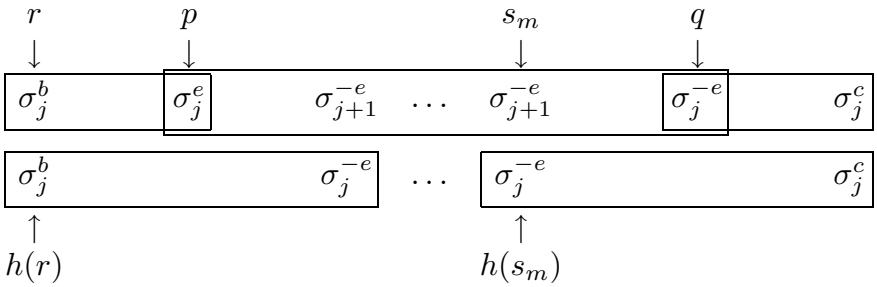
Case 2: $m = 0$, *i.e.*, $\sigma_{j+1}^{\pm 1}$ does not occur in the σ_j-handle that is reduced.
The pattern is now as follows:



We complete the definition of H with

$$H(r) = h(r) = r \qquad \text{if} \qquad r \text{ is critical in } w', \text{ and } b = -e,$$
$$H(p) = h(r) = r \qquad \text{if} \qquad r \text{ is critical in } w',\ b = e, \text{ and } c = -e.$$

Case 3: $m \geq 1$ and $d = -e$, *i.e.*, σ_{j+1}^{-e} occurs in the σ_j-handle that is reduced.
The pattern is here:



We complete now the definition of H with

$$H(p) = h(r) = r \qquad \text{if} \qquad r \text{ is critical in } w', \text{ and } b = e,$$
$$H(q) = h(s_m) \qquad \text{if} \qquad h(s_m) \text{ is critical in } w', \text{ and } c = -e.$$

So we are done with the critical positions of $\sigma_j^{\pm 1}$, and it remains to consider the letters $\sigma_{j+1}^{\pm 1}$. Again there is no problem on the left of p and on the right of q, and for such positions we take $H(x) = h(x)$. Now we see on (3.2) that new σ_{j+1}-handles may appear in w', namely $m-1$ ones that correspond to the factors $\sigma_{j+1}^e v_t \sigma_{j+1}^{-e}$, $t = 1, \dots, m$, and, possibly, two additional ones, one at the right of the factor $u_1 v_1 \sigma_{j+1}^{-e}$ and one at the left of the factor $\sigma_{j+1}^e v_2 u_2$. The example of the pair $(\sigma_1\sigma_2\sigma_1^{-1}\sigma_2^{-1}, \sigma_2^{-1}\sigma_1\sigma_2\sigma_2^{-1})$ shows that there is no hope to obtain a surjective mapping of the σ_{j+1}-handles of w onto the σ_{j+1}-handles of w', so we shall not try to extend further the definition of the heiring function H. However we can summarize the previous easy analysis as

Lemma 3.1. *Assume that w' is obtained from w by reducing one σ_j-handle. Then the critical positions in w' are the heirs of critical positions in w, augmented with at most $m+1$ new critical positions of $\sigma_{j+1}^{\pm 1}$, where m is the number of letters $\sigma_{j+1}^{\pm 1}$ in the σ_j-handle that is reduced from w to w'.*

We arrive to the core of the argument. The fact that a position is active in a sequence of handle reductions does not forbid its heirs to be still active subsequently. But we shall associate with every critical position of the initial word w and its heirs a path traced in the Cayley graph of $S(w)$ in such a way that some characteristic trace is left behind on the path at each time some hair of the considered position is active. The ordering phenomenon of Proposition 1.1 will then imply that such traces can appear only finitely many times – and henceforth force the termination of handle reduction. To this end we shall study the *prefixes* associated with the critical positions and their transformations under heiring.

Definition. For w a braid word and x a position in w, $\pi(w, x)$ is the *prefix* of w that ends at position x, *i.e.*, the word made of the x first letters of w.

Lemma 3.2. *Assume that w' is obtained from w by reducing one handle. Assume that x is a critical position of σ_k^e in w, and that x' is the heir of x in w'. Then there exists a word u such that $\pi(w', x')$ is equivalent to $\pi(w, x)u$, u is traced in $S(w)$ from $D_L(w)\pi(w, x)$, σ_k^e does not occur in u, and σ_k^{-e} occurs (once) in u just in case x is active from w to w'.*

Proof. We keep the previous notations for w, w' and their decompositions, and come back to the study that precedes Lemma 3.1. If x is less than p, then x' is x, and $\pi(w', x')$ is simply $\pi(w, x)$. If x is larger than q, then $\pi(w', x')$ is obtained from $\pi(w, x)$ by reducing the σ_j-handle at position p, q, and $\pi(w', x')$ is equivalent to $\pi(w, x)$. So in both cases we can take for u the nullstring.

Assume now $p \le x \le q$. First, if x is a critical position of $\sigma_k^{\pm 1}$ with $k \neq j, j+1$, then x' is $h(x)$, and the proof of Lemma 2.2 shows that the equivalence

$$\pi(w', x') \equiv \pi(w, x)\ \sigma_j^{-e}$$

holds, and that σ_j^{-e} is traced in $S(w)$ from $D_L(w)\pi(w, x)$: so taking σ_j^{-e} for u is convenient. Next, observe that x cannot be a position of $\sigma_{j+1}^{\pm 1}$ by definition of a permitted

σ_j-handle. It remains to consider the case when x is a critical position of $\sigma_j^{\pm 1}$, *i.e.*, the cases of p and q. Assume $x = p$. By construction the heir of p (when defined) is either r (cases $d = -e$ and $m = 0$), or $h(s_m)$ (case $d = e$). In the first case (Figure 3.2) $\pi(w', x')$ is simply a prefix of $\pi(w, x)$, and from

$$\pi(w', r) \equiv \pi(w, p)\ \sigma_j^{-e} u_1$$

we conclude that taking $\sigma_j^{-e} u_1$ for u is convenient.

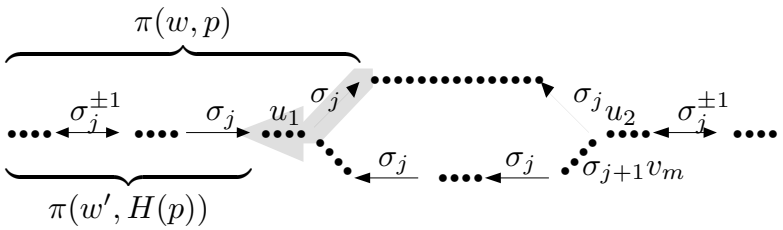


Figure 3.2: Behaviour of prefixes, case $b = d = -e\ (= -1)$

In the second case, the relation is not so simple, but we read on Figure 3.3 the equivalence $\pi(w', h(s_m)) \equiv \pi(w, p)\ u$ with

$$u = v_1 \sigma_{j+1}^e v_2 \ldots v_{m-1} \sigma_{j+1}^e v_m\ \sigma_j^{-e}\ v_m^{-1} \sigma_{j+1}^{-e},$$

and this word u is convenient (σ_j^e does not occur in it, σ_j^{-e} occurs once, and the word is traced in $S(w)$ since the associated path is made of fragments of the paths associated with w and w').

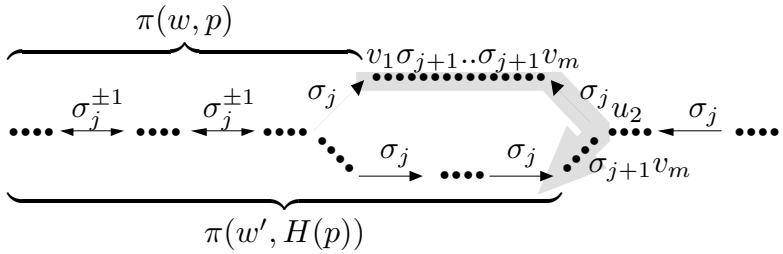


Figure 3.3: Behaviour of prefixes, case $d = e = -c\ (= 1)$

It remains to consider the case $x = q$. By construction the heir of q, when defined, is $h(s_m)$. We read on Figure (3.2) the equivalence

$$\pi(w', h(s_m)) \equiv \pi(w, q)\ v_m^{-1} \sigma_{j+1}^{-1},$$

and see that $u = v_m^{-1} \sigma_{j+1}^{-1}$ is convenient. So all cases have been considered. ■

Remark. The previous result is crucial, and somehow surprising. Indeed we see that the possible heir of the active position is either smaller, or larger than this position according to the sign of the intermediate letters $\sigma_{j+1}^{\pm 1}$, and this corresponds to the fact that the associated prefix becomes either shorter or longer. What is remarkable is the fact that, in both cases, the quotient word u contains a letter σ_j^{-e} (and no letter σ_j^e). This is an *ordering* phenomenon. Indeed let us restrict to the main reductions, *i.e.*, the ones that involve the main generator σ_i of the considered words. Then, for the braid ordering of Section 1, the above result gives the strict inequality

$$\overline{\pi(w', x')} < \overline{\pi(w, x)}$$

where x is the active critical position. The existence of such an inequality is the essential reason why cycles are impossible in handle reduction. Another way to illustrate the phenomenon is to use the graph of the characteristic function introduced above: Figure 3.4 is the exact counterpart of Figures 3.2 and 3.3 in the case of main reductions. In both cases ($d = e$ and $d = -e$) we see that the graph associated to the new word w' is obtained from the graph associated with the old word w by "smoothing" the hill that was associated with the handle that has been reduced (this corresponds to the case $e = 1$, in the case $e = -1$ the transformation would consist in filling a vail instead of eroding a hill). This phenomenon is due to the fact that the main generator is always preponderant over all other generators in the braid ordering.

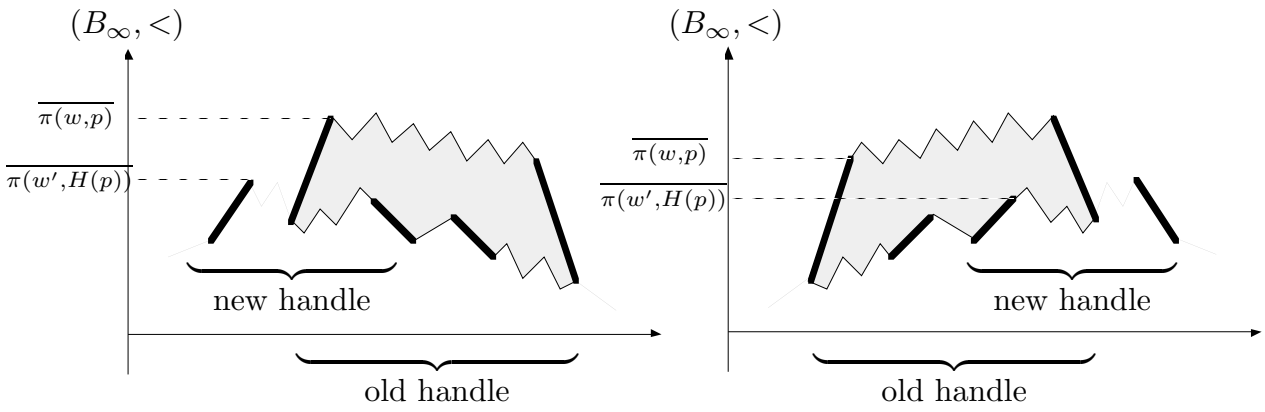


Figure 3.4: Smoothing of characteristic function

Lemma 3.3. *Assume that x is a critical position of σ_k^e in w. If the heirs of x are N times active in some sequence of handle reductions from w, there must exist a word that is traced in $S(w)$ and contains N letters σ_k^{-e} and no letter σ_k^e.*

Proof. Let $w_0 = w$, w_1, ... be the considered sequence of reductions, and let $x_0 = x$, x_1, ... be the iterated heirs of x in these words (as long as they are defined: the sequence of the x_t's can be shorter than the sequence of the w_t's, which we do not assume to be even finite). By Lemma 3.2 we find words u_1, u_2, ... satisfying

$$\pi(w_{t+1}, x_{t+1}) \equiv \pi(w_t, x_t) u_{t+1}$$

(whenever x_{t+1} is defined), and such that u_{t+1} is traced in $S(w_t)$ from $D_L(w_t)\pi(w_t, x_t)$. Because each word w_t is obtained from w by handle reduction, we know by Lemma 2.7

that u_{t+1} is also traced in $S(w)$ from $D_L(w)\pi(w_t, x_t)$, and therefore (any finite prefix of) the word $u = u_1u_2\dots$ is traced in $S(w)$ (from $D_L(w)\pi(w, x)$). By construction there is no σ_k^e in u, and the number of letters σ_k^{-e} is exactly the number of positions x_t that are active in the considered sequence of reductions. ■

If the index k is 1 (or, more generally, corresponds to the main generator of the involved words), a word like u above is a reduced word, and being traced in the finite set $S(w)$ implies a strong limitation on the number of letters σ_1. Actually the general case is similar, thanks to the following extension of Proposition 1.1 (there is no analog strengthening of Proposition 1.2):

Proposition 3.4. ([5], see also [11]) *Say that a braid word is σ_k-reduced if exactly one of the two letters σ_k, σ_k^{-1} occurs (any number of times) in w. Then a braid word that is σ_k-reduced for at least one integer k is not trivial.*

Corollary 3.5. *Assume that the braid word w has length ℓ and width n. Then the number of letters $\sigma_k^{\pm1}$ in any σ_k-reduced word traced in $S(w)$ is bounded above by $(n-1)^{n^2\ell}$.*

Proof. Let u be any σ_k-reduced word traced in $s(w)$. Write u as $u_0\sigma_k^e u_1\sigma_k^e\dots u_{N-1}\sigma_k^e u_N$ ($e = \pm1$), where $\sigma_k^{\pm1}$ does not occur in the words u_t. By Proposition 3.4, a subword of u of the form $u_{t_1}\sigma_k^e\dots\sigma_k^e u_{t_2}\sigma_k^e$ with $t_1 < t_2$ is never trivial. This proves that, in the Cayley graph of $S(w)$, the N arrows that correspond to the N positions of σ_k^e in u must be pairwise distinct. It follows that N is at most equal to the total number number of σ_k-labelled arrows in the graph of $S(w)$. We have seen that the length of $D_L(w)N_R(w)$ is bounded above by $n^2\ell$, and therefore (since $n-1$ letters may appear), $(n-1)^{n^2\ell}$ is certainly an upper bound for N. ■

Definition. Let w be any braid word. The *rank* of w is the maximal number of letters σ_k^e in a σ_k-reduced word traced in $S(w)$ (for all possible k).

So Corollary 3.5 tells us that the rank of any braid word is finite, and is at most equal to $(n-1)^{n^2\ell}$ for a word of length ℓ and width n. Because this bound seems to be rather bad, we shall use the rank in the subsequent evaluations. We deduce from Lemma 3:

Lemma 3.6. *Assume that the rank of the braid word w is r. Then for any critical position x in w, the heirs of x can be active at most r times in any sequence of handle reductions from w.*

Let us say that a handle reduction is a σ_k-reduction if the involved handle is a σ_k-handle. We can state a first finiteness result for reduction:

Proposition 3.7. *Assume that the rank of the braid word w is r and that w contains c σ_k-handles. Then there are at most cr σ_k-reductions in any sequence of handle reductions from w that contains no σ_{k-1}-reduction.*

Proof. As long as no σ_{k-1}-reduction is operated, the only critical positions of $\sigma_k^{\pm 1}$ in the considered words are the heirs of the ones in w. There are c of them, and, by Lemma 3.6, the heirs of each of these initial critical positions are active at most r times. ■

Observe that the previous result applies in particular to the case of main reductions: so we already know that the number of *main* reductions in any sequence of handle reductions from an initial word is finite, and is at most cr, where c is the number of main handles in that word and r is its rank. The sequel is now an easy induction. Precisely we have

Proposition 3.8. *Assume that the braid word w has rank r, contains c handles, and that σ_i is the main generator of w. Then, for every integer n, there are at most $c(2r)^{2n+1}$ σ_j-reductions with $i \leq j < i+n-1$ in any sequence of handle reductions from w.*

Proof. Let N_j denote the maximal (possibly infinite) number of σ_j-reductions in a sequence of handle reductions from w. We claim that the inequality

$$N_{j+1} \leq (c + N_j(r+1))r \tag{3.3}$$

always takes place. Indeed consider an arbitrary sequence of handle reduction from w. We know that the heirs of each initial critical position of $\sigma_{j+1}^{\pm 1}$ are active at most r times in the considered sequence, and that there are at most c of them. Now each σ_j-reduction in the sequence (possibly) creates new $\sigma_{j+1}^{\pm 1}$-positions, that are not the heirs of previous positions, and that will each contribute, together with their heirs, for at most r additional σ_{j+1}-reductions. The number of new critical positions of $\sigma_{j+1}^{\pm 1}$ created by one σ_j-reduction, *i.e.*, the parameter $m+1$ of Lemma 3.1, is variable, but a uniform bound can be given: indeed, by definition of a permitted handle, the intermediate word $v_1\sigma_{j+1}^d v_2 \ldots v_{m-1}\sigma_j^d v_m$ is a σ_j-reduced word, which is traced in $S(w)$. By definition of the rank of w the integer m is at most equal to r. So each σ_j-reduction creates at most $r+1$ new critical positions of $\sigma_{j+1}^{\pm 1}$, and (3.3) follows. Now we have noted that N_i is finite, and is bounded above by cr. So, using (3.3), we inductively obtain

$$N_{i+k} \leq (2^{k+1}-1)cr^{2k+1},$$

whence the announced value follows. ■

Now taking for n the width of the braid word w in the previous result, noting that the number of handles in w is always bounded by the length of w, and using for the rank of w the upper bound of Corollary 3.5 gives at once the bound of Theorem 1.4, which therefore is proved. We can still observe that the arguments of the present section enables us to make Corollary 1.6 slightly more precise:

Corollary 3.9. *Assume that the braid β admits a decomposition of length ℓ and width n. Then β admits a fully reduced decomposition of length at most $(n-1)^{n^2\ell}$ and width n.*

Proof. Assume that w' is fully reduced word and can be deduced from w using handle reduction. Then w' is traced in the set $S(w)$, and, by Proposition 1.1, no nonempty subword of w' may be trivial (since each such subword is reduced): this means that the path associated with w' in the Cayley graph of $S(w)$ cannot visit twice the same vertex, and, therefore, the length of w' is certainly bounded by the number of vertices in this graph. ■

Remark. The argument developed in this section holds for *any* reduction method that consists in replacing a handle with an equivalent word that is reduced. So it works as well for the "coarse reduction" obtained by replacing everywhere the local reduction of Figure 1.4 with the coarse reduction of Figure 1.3. However, because the boundedness result of Section 2 does not hold for coarse reduction (consider for instance the case of the word $w = \sigma_1\sigma_3\sigma_1^{-1}$, which reduces to $\sigma_2^{-1}\sigma_3^{-1}\sigma_2\sigma_3\sigma_2$, a word that is not traced in $S(w) = \{\sigma_1\sigma_3, \sigma_3\sigma_1\}$), we cannot conclude anything about its termination (experiments suggest that the results are quite similar in both cases; however coarse reduction is less efficient in practice). In order to prove the latter termination it would be sufficient to modify the construction for the set $S(w)$ so that the set of the words traced in it becomes closed under coarse reduction. This could be possibly done by transforming the notions of R- and L-reductions so that for instance $\sigma_1^{-1}\sigma_3$ reduces to $\sigma_3\sigma_2\sigma_1\sigma_2^{-1}\sigma_1^{-1}\sigma_2^{-1}$ (instead of $\sigma_3\sigma_1^{-1}$).

4. Algorithmic aspects, and open questions

It is fairly easy to construct practical algorithms using the principle of handle reduction. We shall briefly describe a few of them, and give some hints about their remarkable efficiency (in complete contradistinction with the upper bound of Theorem 1.4), in particular when compared with the previously known algorithms. This discrepancy between the theoretical results and the experimental datas leads to some natural conjectures about handle reduction.

Construction of the algorithms

By definition a braid word that is not reduced must contain main handles which we wish to treat using reduction. But the principle of reduction forces us to reduce only permitted handles, so that we cannot avoid in general reducing first some handles that are not main handles. So the question for building an actual algorithm is to define a strategy for choosing which handles are to be reduced first. We have observed that free reduction, *i.e.*, deletion of factors $\sigma_k\sigma_k^{-1}$ and $\sigma_k^{-1}\sigma_k$, is a special case of handle reduction, and, as a consequence, fully reduced words are always freely reduced words. Practical experiments, as well as some partial results, show that it is more efficient to work with freely reduced braid words. This means that, when choosing the order of the reduction steps to be performed, it is advisable to insert a complete free reduction after each reduction of a handle, rather than waiting that free reductions possibly occur only later when the strategy tells us to consider them as normal reductions. So it is natural to take

Definition. The braid word w' is deduced from w *using one step of HF-reduction* if w' is obtained from w by reducing some permitted handle of w, and then free reducing the resulting word.

The first obvious strategy simply consists in systematically choosing the leftmost handle, *i.e.*, the one that ends at the minimal possible position – whether it is a main one or not. This makes sense, because Lemma 1.5 guarantees that such a leftmost handle must always be a permitted one. So the first reduction algorithm is the following

Algorithm "`FullHRed`":

Start with any braid word w, and iteratively *HF*-reduce the leftmost handle of the current word until a fully reduced word is obtained. (Recall that the final word w' is equivalent to the initial word w, and that w is trivial if and only if w' is empty.)

(See an example below.) As we can expect, fully reduced words are useful if we look for short decompositions. It is clear that, if we are only interested in solving the word problem, *i.e.*, in obtaining simply reduced decompositions, it is not necessary to reduce all handles as in `FullHRed`. So a more "greedy" way to process is to reduce only the main handles and, in a recursive way, the intermediate unavoidable handles that prevent the main handles to be permitted. Such unavoidable handles are easily described:

Definition. Assume that σ_i is the main generator of the braid word w. A σ_j-handle of w lying between positions p and q is *nested* if there exists a sequence of nested intervals

$$(p_j, q_j) = (p, q) \subset (p_{j-1}, q_{j-1}) \subset \ldots \subset (p_i, q_i)$$

such that, for every k, the subword of w lying between p_k and q_k is a σ_{i+k}-handle.

It is clear that any nested σ_j-handle has to be reduced before any σ_{j-1}-handle that includes it. Again we check that the leftmost nested handle of a (non reduced) word must be permitted, so the natural counterpart to `FullHRed` will be the following 'greedy' version

Algorithm "`GreedyHRed`":

Start with any braid word w, and iteratively *HF*-reduce the leftmost *nested* handle of the word until a reduced word is obtained. (Again the final word w' is equivalent to the initial one, and that w is trivial if and only if w' is empty.)

Example 0.5. The action of `FullHRed` and `GreedyHRed` on the word we used in the examples of Section 2 is nearly trivial (it comprises only two steps). We give below a less trivial example where both algorithms differ, namely the braid word $\sigma_1^{-1}\sigma_2^{-1}\sigma_1\sigma_3\sigma_2^{-1}\sigma_3^{-1}\sigma_2^{-1}\sigma_1\sigma_3^{-1}\sigma_2\sigma_1^2$. We shall improve readability by using the convention that `a`, `b`, ... stand for σ_1, σ_2, ..., and that (as in [9] for instance) capitals denote the inverses of the corresponding lower case letters: `A` for σ_1^{-1}, *etc.*So the word above becomes `ABacBCBaCbaa`. The left column below shows the successive words that appear when `FullHRed` is applied, and the right column does the same for `GreedyHRed`. The handles that are reduced at each step are underlined. Since σ_1^{-1} occurs in the final words, but σ_1 does not, the involved braid is below 1 in the braid ordering of Corollary 1.3, and, in particular, it is not trivial. Typical facts appear on this example: for instance, full reduction requires more steps than greedy reduction, but the final word is shorter, and is even shorter than the initial word (but some intermediate words are longer).

<u>ABa</u>cBCBaCbaa	<u>ABa</u>cBCBaCbaa

bABcBCBaCbaa
bbABcbABCbABCbaa
bbAcbCABCbABCbaa
bbAcbCAcBCABCbaa
bbAcbABCABCbaa
bbAcbABCAcBCaa
bbAcbABABCaa
bbAcbAABCa
bbAcbAbABC

bABcBCBaCbaa
bbABcbABCbABCbaa
bbABcbABCbAcBCaa
bbABcbABCbcbABCa
bbABcbABCbcbbABC

Remark. A standard improvement method can be applied to the previous algorithms when they are to be run on long words, namely the "divide-and-conquer" trick: in order to reduce a (long) word w, we divide w into the product of two words w_1, w_2 whose length is approximately the half of the length of w, we apply reduction separately to w_1 and w_2 to obtain reduced words w_1' and w_2', and finally we reduce using `GreedyHRed` (or `FullHRed`) the word $w_1'w_2'$. The potential practical advantage of this variant is clear: in particular the probablilty is 1/2 that the words w_1' and w_2' have the same type (positive or negative), in which case the last step vanishes. However let us observe that this "quick versions" are still reduction strategies.

From Proposition 3.8 we immediately deduce the correctness of the above algorithms (as methods to solve the isotopy problem of braids):

Proposition 4.1. *Assume that the braid word w has length ℓ, width n and rank r. Then the algorithms* `FullHRed`, `GreedyHRed` *(or their "quick" variants) running on w return reduced words (and even a fully reduced word in the first case) in at most $\ell(2r)^{2n+1}$ steps.*

Convex reduction

A cumbersome phenomenon that partially explains the very high bounds obtained in Section 3 for the number of handle reductions is the proliferation of critical positions. With the notations of Lemma 3.1, we have seen that each σ_j-reduction causes $m-1$ intermediate new σ_{j+1}-handles to appear. However it is not difficult to guess what the subsequent reduction of these new handles will be, *i.e.*, what will happen if we choose to reduce them, and, iteratively, the analog σ_{j+k} that will possibly appear, just after they have been created (and not later). Indeed the idea is that the final route of the $j+1$-th strand will be some sort of convex hull that is obtained by skirting on the right all crossings we meet, as illustrated in Figure 4.1 below (practically we can construct for the $j+1$-th strand two half-routes starting from both ends and merging in the middle).

We can then consider 'convex' versions of the previous algorithms

Example 0.6. Using the convex version of `FullHRed`, the complete reduction of the above word considered in the previous example comprises new 6 steps (to be compared with the 9 steps of `FullHRed`), namely

ABacBCBaCbaa

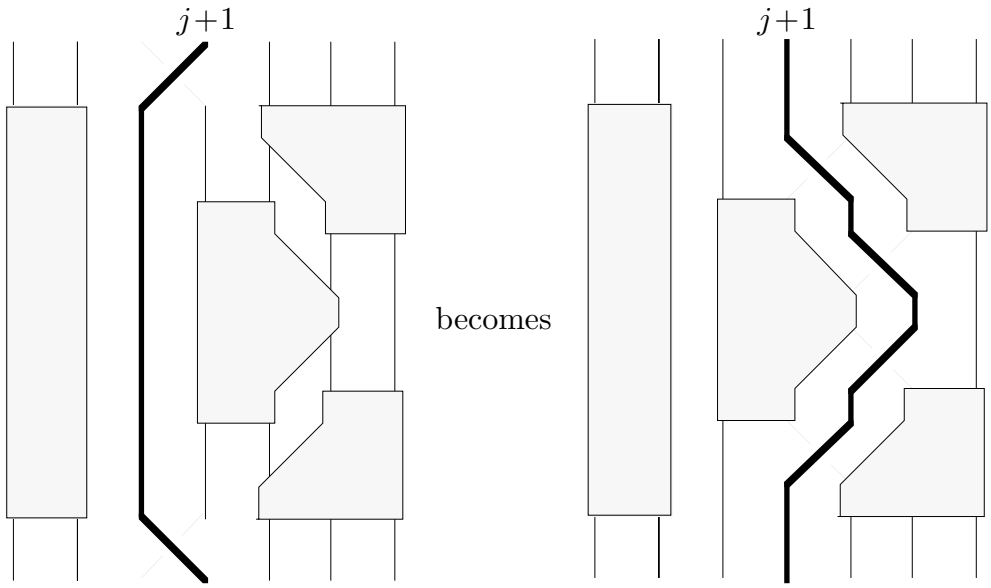


Figure 4.1: Convex reduction of a handle

bABcBCBaCbaa
bbAcbABCABCbaa
bbAcbABCAcBCaa
bbAcbABABCaa
bbAcbAABCa
bbAcbAbABC

The analysis of Section 3 is a little modified when convex reduction replaces local reduction. Indeed reducing a σ_j-handle will still possibly create new critical positions, but only at the ends: the $m-1$ intermediate new critical positions of $\sigma_{j+1}^{\pm 1}$ are avoided, and there remain only the two possible ones associated with the initial pattern $u_1 v_1 \sigma_{j+1}^{-e}$ and with the final pattern $\sigma_{j+1}^{e} v_m u_2$. On the other hand, two new critical positions of all letters $\sigma_{j+k}^{\pm 1}$ with $k \geq 0$ can appear similarly (what did not happen with local reduction). In the evaluation of the number of possible reduction steps, we replace the factor $(r+1)$ in the proof of Proposition 3.8 with a constant factor 2, and, from there, we can lower the exponent $2n+1$ for the rank in Proposition 3.8 down to $n+1$:

Proposition 4.2. *Assume that the braid word w has length ℓ, width n and rank r. Then the convex versions of the algorithms* `FullHRed` *or* `GreedyHRed` *running on w return reduced words (and even a fully reduced word in the first case) in at most $\ell(2r)^{n+1}$ steps.*

This of course does not really change the final bound owing to our present evalution of the rank.

Conjectures

Implementing the preceding methods is very easy, and enables us to gather some statistics. The main point that comes out of these experiments is that handle reduction is *very* efficient, and that the number of reduction steps is, in all (many) cases we have studied, much smaller than the upper bounds that are established above. To give an example, the

average number of reduction steps in the reduction of random braid words of length 1000 with respectively 3, 5, 10 and 50 strands, using the algorithm `GreedyHRed`, are 702, 1420, 1298 and 34. (These numbers correspond to samples of 10,000 words or more, and seem very reliable in particular because the convergence to the limit value is quick and regular.) Such values explain that handle reduction algorithms give extremely quick methods for deciding if a braid word is trivial: using the quick variant of `GreedyHRed`, the average computing time for braid words of length 1000 is never larger than 0.25 sec. for *any* fixed width on a standard microcomputer (Macintosh PPC 601).

Several factors can explain why the bounds of Section 3 seem to be so far from optimum First we observe that the bound of Proposition 3.7 can be nearly reached in some cases (the word $(\sigma_1^r\sigma_1^{-r})^m$ has rank r, contains $2m-1$ handles, and rm reductions are needed to reduce it to the nullstring), but is usually far from optimal when more than one generator is involved: even if long σ_k-positive words are traced in some set $S(w)$, there is no reason why every such word should occur in connection with an actual sequence of reductions. Next the majoration of the rank given by Corollary 3.5 is probably very bad as well. Because every positive braid word is a prefix of some word Δ_n^d, where Δ_n is Garside's 'universal' word on $n-1$ generators (*cf.* [10]), the basic question is to compute the rank of the words Δ_n^d, and, in particular, to decide if this rank is, for fixed n, a polynomial function of the exponent d. We have no answer to this question. However it can be shown that the rank of Δ_n^d is at least d^{n-1}, and, therefore, is not always quadratic with respect to the length of the word. This suggests that using the rank cannot really give the optimal bounds in the evaluation of the number of handle reductions. Indeed all examples we have studied (in particular in terms of growth rates) are compatible with the following conjectures:

Conjecture 4.3. *For any fixed width, the number of HF-reductions from a braid word w is bounded above by a quadratic function of the length of w.*

Conjecture 4.4. *For any fixed width, the lengths of all words deduced from a braid word w using RF- and LF-reductions, and PF- and NF-equivalence (the variants of R-, L-reduction and of P-, N-equivalence obtained by systematically inserting free reduction at each step), and therefore also using HF-reduction, are bounded by a linear function of the length of w.*

Conjecture 4.3 is certainly optimal, as we can easily obtain sequences of words such that the number of possible (main) reductions grows as the square of the length: so are for instance the complete reduction of the width 3 words $(\sigma_2^2\sigma_1^2)^m\sigma_2(\sigma_1^{-2}\sigma_2^{-2})^m$ (using any method) requires $O(m^2)$ main steps. Let us mention two partial results:

Proposition 4.5. *i) Conjecture 4.3 is true for width 3.*

ii) For width 4, there exists a polynomial bound (namely, cubic) for the length of the words that appear in the algorithm similar to `FullHRed` *where the coarse reduction of Figure 1.2 replaces the local reduction of Figure 1.3.*

Proof. i) The case of width 3 is special because the handles that are not main handles are "trivial" handles of the form $\sigma_j\sigma_j^{-1}$ or $\sigma_j^{-1}\sigma_j$, and because the length cannot increase when *HF*-reduction is used (which makes Conjecture 4.3 trivial). It follows that, in the

reduction of a word with exactly one main handle, there can be only *one* back-and-forth move of the main handle, so that a direct argument shows that the number of main reductions is at most quadratic with respect to the length (even if free reduction is not systematically added).

ii) For width 4 and coarse reduction, we can use a specific argument involving the length with respect to a extended family of generators that comprises σ_1, σ_2, σ_3, $\sigma_1\sigma_2$, $\sigma_2\sigma_3$ and $\sigma_1\sigma_2\sigma_3$ and their mirror images. This argument does not seem to extend to the general case. ■

Conjecture 4.4 remains open, although weaker forms are well known: if we consider only *RF*-reduction, or only *LF*-reduction, then Lemma 2.1 applies. We can still prove some facts when *RF*-reduction and *PF*- and *NF*-equivalences are considered simultaneously, but adding *LF*-reduction seems to require new tools beyond the classic results about the divisors of Δ_n (*i.e.*, the automatic structure of B_n). In any case the fact that free reduction is needed here (otherwise the trivial counterexample of Section 2 refutes the conjecture) shows that the involved statement involves the Cayley graph of B_∞ rather than the braid words themselves.

Remark. The above conjectures deal with the worst cases in handle reduction. Presently we have no precise conjecture for the average values of the considered parameters. The experiments suggest that there is a rather large gap between the average case and the worst case. Actually the words that are "bad" for handle reduction seem to be very special (in particular the Cayley graph of the associated set $S(w)$ seems to be planar), which could explain the above gap. Such a situation would be reminiscent of a scheme that is rather common in the neighbouring case of statistics for the symmetric group.

Comparison with Thurston's normal form

We finish this paper with a brief comparison of handle reduction with Thurston's method as described in [17] or [9]. To make the comparison easier we consider the variant of Thurston's algorithm that is described in [6] and appeals to *R*- and *L*-reduction in order to construct, for any braid word w, an equivalent braid word $u^{-1}v$ where u and v are positive and are the shortest possible words with that property (so that in particular w is trivial if and only if the words u and v are both empty). Thurston's algorithm constructs such positive words using a notion of normal form for positive words (the 'greedy normal form') whose construction follows from the automatic structures of the groups B_n, while [6] directly obtains such words by means of a double reduction (and without any normal form): start from w, reduce it on the right to $u'v'^{-1}$, and then reduce the latter word on the left to $u^{-1}v$. We shall call this method the *RL*-reduction of the word w. If u, v and u_1, v_1 are the positive words produced from w respectively using *RL*-reduction and using Thurston's method, then u and u_1 are equivalent (but not equal in general), and so are v and v_1. So we can consider both methods as essentially equivalent, and in particular the length of the final words obtained are the same.

Observe that the counterparts of Conjectures 4.3 and 4.4 are true for *LR*-reduction (as well as for Thurston's method): for a fixed width n, the number of elementary steps in the *RL*-reduction of a word w of length ℓ is bounded above by a quadratic function of ℓ (we can take $(1/32)n^4\ell^2$)), and the length of the final word is bounded above by a linear function of ℓ (we can take $(1/2)n^2\ell$).

The experiments show that handle reduction is practically much more efficient than the previous methods, specially when the width of the braids increases (as mentioned in [9], it is not easy to implement Thurston's method when the width n goes beyond say 8, because it appeals to a precomputed table of $n! \times n!$ complements, while LR-reduction avoids such a feature, and therefore can be easily implemented for any width). To obtain a valuable comparison we have considered the number of cells in the Cayley graph that are visited during the reduction process (counted with their multiplicity): in other words we count how many calls to the basic braid relations (1.1) are made (simply comparing the number of reduction steps would not be correct since one step of handle reduction actually decomposes into possibly many R- or L-reductions). Also we have considered the version of LR-reduction where free reductions are inserted at each step (which diminishes the number of steps).

The comparison for random braid words shows that the number of visited cells is always much smaller in handle reduction: for instance, for random braid words with 100 crossings, the values are 41 *vs.* 292 in the case of 3 strands, 115 *vs.* 2,360 for 5 strands, 94 *vs.* 11,810 for 10 strands, and 18 *vs.* 7,815 for 50 strands. A clear heuristic reason explains the difference: both reductions happen in the Cayley graph of the set $S(w)$, but RL-reduction "blindly" crosses twice this graph, a first time to the right, a second time to the left, while handle reduction is "piloted" by the braid order and keeps the same orientation toward the final word during the whole process. The same phenomenon occurs in the graphs of the characteristic functions (Figure 4.2): in one case we smooth the initial graph at each step, while in the other case we successively go up to a unique peak, and then down to a unique valley. (In some sense we could see R- and L-reduction as piloted by the *partial* ordering of braids considered in [8].)

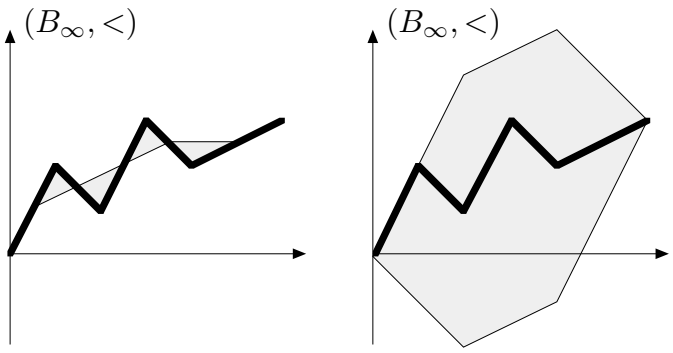


Figure 4.2: reduction *vs.* RL-reduction

Our final remark will concern the length of the final word obtained using handle reduction when compared with the length of the words obtained using RL-reduction (which, we remind, is equal to the length of Thurston's normal form). The same difference can be found again: always for random words of initail length 100 and respectively 3, 5, 10, and 50 strands, we find (using `FullHRed`) for the final length compared values of $32vs.40$, $62vs.105$, $67vs.211$ and $72vs.146$. The phenomenon illustrated in Figure 4.2 explains the difference again. In a more general way, the above observations suggest that (full) handle reduction is an efficient tool for producing short decompositions of braids. It

is not always true that the final word obtained in this way is shorter than the initial word, but it could happen that this word is, in some sense, a good approximation for the minimal decomposition (the one with minimal length) of the considered braid. If this is true, and a polynomial bound can be proved for the number of steps in handle reduction, this possible property should be compared with the NP-completeness result of [14]. Let us mention that the above results about short decompositions can still be enhanced by mixing the full reduction process and the operation of iteratively minimizing the number of generators with low index using the transformation $\sigma_j\sigma_{j+1}\sigma_j \mapsto \sigma_{j+1}\sigma_j\sigma_{j+1}$.

References

[1] E. Artin, *Theory of Braids*, Ann. of Math. **48** (1947) 101–126.

[2] J. Birman, *Braids, links, and mapping class groups*, Annals of Math. Studies **82** Princeton Univ. Press (1975).

[3] S. Burckel, *The wellordering of positive braids*, J.P. Appl. Algebra, to appear.

[4] P. Dehornoy, *Deux propriétés des groupes de tresses*, C. R. Acad. Sci. Paris **315** (1992) 633–638.

[5] —, *Braid Groups and Left Distributive Operations*, Trans. Amer. Math. Soc **345-1** (1994) 115–151.

[6] —, *Groups with a Complemented Presentation*, preprint (1993).

[7] —, *From Large Cardinals to Braids via Distributive Algebra*, J. Knot Theory & Ramifications **4-1** (1995) 33–79.

[8] E. A. Elrifai & H. R. Morton, *Algorithms for positive braids*, Quart. J. Math. Oxford **45-2** (1994) 479–497.

[9] D. Epstein & *al.*, *Word Processing in Groups*, Jones & Barlett Publ. (1992).

[10] F. A. Garside, *The Braid Group and other Groups*, Quart. J. Math. Oxford **20** No.78 (1969) 235–254.

[11] D. Larue, *On Braid Words and Irreflexivity*, Algebra Univ. **31** (1994) 104–112.

[12] —, *Left-Distributive and Left-Distributive Idempotent Algebras*, Ph D Thesis, University of Colorado, Boulder (1994).

[13] R. Laver, *Braid group actions on left distributive structures and well-orderings in the braid group*, preprint (1993).

[14] M.S. Paterson & A. A. Razborov, *The Set of Minimal Braids Is Co-NP-complete*, J. of Algorithms **12** (1991) 393–408.

[15] J. Pedersen & M. Yoder, *Term Rewriting for the Conjugacy Problem and the Braid Groups*, preprint (1993), to appear in J. Symb. Comp.

[16] K. Tatsuoka, *An Isometric Imequality for Artin Groups of Finite Type*, Trans. Amer. Math. Soc. **339–2** (1993) 537–551.

[17] W. Thurston, *Finite state algorithms for the braid group*, Circulated notes (1988).

Mathématiques, Université, 14 032 Caen, France
dehornoy@geocub.greco-prog.fr