Available online at www.sciencedirect.com
SCIENCE DIRECT®

Journal of Algebra 292 (2005) 282–302

JOURNAL OF Algebra

www.elsevier.com/locate/jalgebra

A new approach to the conjugacy problem in Garside groups

Volker Gebhardt

School of Quantitative Methods and Mathematical Sciences, University of Western Sydney, Australia

Received 10 January 2005

Available online 17 March 2005

Communicated by Patrick Dehornoy

Abstract

The cycling operation endows the super summit set S_x of any element x of a Garside group G with the structure of a directed graph Γ_x. We establish that the subset U_x of S_x consisting of the circuits of Γ_x can be used instead of S_x for deciding conjugacy to x in G, yielding a faster and more practical solution to the conjugacy problem for Garside groups. Moreover, we present a probabilistic approach to the conjugacy search problem in Garside groups. The results have implications for the security of recently proposed cryptosystems based on the hardness of problems related to the conjugacy (search) problem in braid groups.

Keywords: Garside groups; Braid groups; Conjugacy problem; Conjugacy search problem; Super summit set; Ultra summit set; Braid group cryptography

1. Introduction

Given a group G, the *conjugacy problem* in G is to decide for given elements $a, b \in G$, whether a and b are conjugate in G, that is, whether there exists an element $c \in G$ such that $a^c = b$. The *conjugacy search problem* in G, on the other hand, is to find for given elements a and b which are known to be conjugate in G, an element $c \in G$ such that $a^c = b$.

E-mail address: v.gebhardt@uws.edu.au.

0021-8693/$ – see front matter
doi:10.1016/j.jalgebra.2005.02.002

Both problems are known to be solvable in Garside groups, that is, in particular in braid groups [3,8,10,11,15]. However, all known algorithms involve computing a particular invariant of the conjugacy class, the so-called *super summit set*, for either a or b and both the memory and the time complexity of these algorithms are proportional to the cardinality of this set. In the case of the braid group B_n, the best proven bound for this cardinality is exponential in both the braid index n and the element length r and, while the existence of polynomial bounds is conjectured, computations in practice are hard or infeasible even for moderate values of n and r.

Recently, braid groups came under interest as possible sources for public key cryptosystems and the security of most of the proposed cryptosystems depends on the hardness of variations of the conjugacy (search) problem [1,13]. Hence an improved understanding of the conjugacy problems is highly desirable.

The crucial point in computing the super summit set S_x of an element x is the following "convexity" property. For any pair of elements $u, v \in S_x$ there are elements $u_0, \ldots, u_k$ with $u_0 = u$ and $u_k = v$, such that for $i = 1, \ldots, k$, u_i is obtained from u_{i-1} by conjugation with a suitable element from a finite set D. This allows us to compute S_x, starting with a single representative, as the closure with respect to conjugation by elements of D.

In this paper we establish that a subset of the super summit set, which in general is much smaller, can be used for deciding conjugacy in Garside groups. The set S_x can be endowed with the structure of a directed graph and we will show that the union of the circuits of this graph has the same "convexity" property as described above, that is, can be computed in a similar way. The graph structure used for proving this result also yields a fast probabilistic algorithm for solving the conjugacy search problem.

1.1. Garside groups and monoids

We start with a brief review of some basic terminology and facts about Garside groups. The results can be found, for example, in [3,6–9,11,15]. Throughout this section, let M be a (left and right) cancellative monoid.

Definition 1.1. We define partial orderings $\preccurlyeq$ and $\succcurlyeq$ on the elements of M as follows. For $a, b \in M$ we say $a \preccurlyeq b$ if there exists an element $c \in M$ such that $ac = b$ and we say $a \succcurlyeq b$ if there exists an element $c \in M$ such that $a = cb$.

We call m a (*left*) *lcm* of a and b if $a \preccurlyeq m$, $b \preccurlyeq m$ and if for any $x \in M$, $a \preccurlyeq x$ and $b \preccurlyeq x$ implies $m \preccurlyeq x$. Similarly, we call d a (*left*) *gcd* of a and b if $d \preccurlyeq a$, $d \preccurlyeq b$ and if for any $x \in M$, $x \preccurlyeq a$ and $x \preccurlyeq b$ implies $x \preccurlyeq d$.

Definition 1.2. $x \in M$ is called an *atom* if $x \neq 1$ and if $x = ab$ for $a, b \in M$ implies $a = 1$ or $b = 1$. M is called *atomic* if M is generated by its atoms and if for every $a \in M$ there exists a bound N_a such that a cannot be written as product of more than N_a atoms.

Definition 1.3. For $\delta \in M$ we define the sets $D^l_\delta = \{x \in M\colon\ x \preccurlyeq \delta\}$ and $D^r_\delta = \{x \in M\colon$ $\delta \succcurlyeq x\}$. The element δ is called a *Garside element* of M if $D^l_\delta = D^r_\delta$ and if D^l_δ is finite and generates M.

The monoid M is called a *Garside monoid* if it is atomic, has a Garside element δ and if for all $a, b \in M$ a gcd and a lcm of a and b exist. In this case, the lcm and gcd of a and b are unique; we denote them by $a \vee b$ and $a \wedge b$. We call the elements of D_δ^l the *simple elements* of M.

Theorem 1.4. *Let M be a Garside monoid with Garside element δ and group of fractions G.*

(a) *M embeds into G.*
(b) *If a is an atom of M then $a \preccurlyeq \delta$.*
(c) *M is invariant under conjugation by δ.*

Definition 1.5. Let M be a Garside monoid with Garside element δ. Its group of fractions G is called a *Garside group*. We identify the elements of M with their images in G and call them the *positive elements* of G. Let $\tau : x \mapsto x^\delta = \delta^{-1} x \delta$ be the automorphism of G induced by conjugation with δ.

The partial orderings $\preccurlyeq$ and $\succcurlyeq$, and thus the notions of left gcd and left lcm, can be extended to G as follows. For $a, b \in G$, we say $a \preccurlyeq b$ if there exists an element $c \in M$ such that $ac = b$ and we say $a \succcurlyeq b$ if there exists an element $c \in M$ such that $a = cb$. Clearly $\preccurlyeq$ and $\succcurlyeq$ are invariant under τ.

Theorem 1.6. *Let M be a Garside monoid with Garside element δ and with group of fractions G.*

(a) *For every $x \in G$ there are integers r and s such that $\delta^r \preccurlyeq x \preccurlyeq \delta^s$.*
(b) *There is an integer k such that δ^k is central in G.*

Example 1.7. Consider the monoid B_n^+ defined by the presentation

$$\begin{aligned} \bigl\langle \sigma_1, \ldots, \sigma_{n-1} \mid \sigma_i \sigma_j = \sigma_j \sigma_i \ (1 \leqslant i < j+1 \leqslant n), \\ \sigma_i \sigma_{i+1} \sigma_i = \sigma_{i+1} \sigma_i \sigma_{i+1} \ (1 \leqslant i \leqslant n-2) \bigr\rangle. \end{aligned} \tag{1}$$

Its quotient group is the braid group B_n on n strings [2]. B_n^+ is a Garside monoid with Garside element $(\sigma_1 \cdots \sigma_{n-1})(\sigma_1 \cdots \sigma_{n-2}) \cdots (\sigma_1 \sigma_2)\sigma_1$. The positive elements of B_n are simply the words in $\sigma_1, \ldots, \sigma_{n-1}$ not involving inverses of generators. There are $n!$ simple elements, corresponding to those braids in which any two strings cross at most once. A simple element is described uniquely by the permutation it induces on the strings and every permutation of the n strings corresponds to a simple element.

Example 1.8. The monoid BKL_n^+ generated by $\{a_{t,s}\colon n \geqslant t > s \geqslant 1\}$ subject to the relations

$$\begin{aligned} a_{t,s} a_{r,q} = a_{r,q} a_{t,s} \quad &\text{if } (t-r)(t-q)(s-r)(s-q) > 0, \\ a_{t,s} a_{s,r} = a_{t,r} a_{t,s} = a_{s,r} a_{t,r} \quad &\text{if } t > s > r \end{aligned} \tag{2}$$

also has the braid group B_n as its quotient group [3]. In terms of presentation (1), $a_{t,s} = (\sigma_{t-1}\cdots\sigma_{s+1})\sigma_s(\sigma_{s+1}^{-1}\cdots\sigma_{t-1}^{-1})$ is a possible choice for the generators $a_{t,s}$.

BKL_n^+ is a Garside monoid with Garside element $a_{n,n-1}a_{n-1,n-2}\cdots a_{2,1}$. The number of simple elements of BKL_n^+ is $(2n)!/(n!(n+1)!)$. Again, a simple element is described uniquely by the permutation it induces on the strings, but not every permutation of the n strings corresponds to a simple element.

Notation 1.9. From now on let M be a Garside monoid with Garside group G, Garside element δ and set of simple elements D.

1.2. Normal forms

Definition 1.10. By Theorem 1.6 for every $x \in G$ there exist integers $r \geqslant 0$ and k such that $\delta^k \preccurlyeq x \preccurlyeq \delta^{k+r}$. Choose k maximal and r minimal subject to this condition. We call k the *infimum*, denoted by $\inf(x)$, r the *canonical length*, denoted by $\operatorname{len}(x)$, and $k+r$ the *supremum*, denoted by $\sup(x)$, of x.

There are uniquely defined elements $A_1,\ldots,A_r \in D$ such that $x = \delta^k A_1\cdots A_r$ and $A_i^{-1}\delta \wedge A_{i+1} = 1$ for $i = 1,\ldots,r-1$. We call this representation of x the *normal form* of x. It is easy to see that $A_1,\ldots,A_r$ can be expressed recursively as $A_i = \delta \wedge (\delta^k A_1\cdots A_{i-1})^{-1}x$ for $i = 1,\ldots,r$. Note that, as $A_i^{-1}\delta \preccurlyeq \delta$, we have $A_{i+1}\cdots A_r \wedge A_i^{-1}\delta = A_{i+1}\cdots A_r \wedge \delta \wedge A_i^{-1}\delta = A_{i+1} \wedge A_i^{-1}\delta = 1$.

1.3. Super summit sets

The notion of super summit sets was developed in [8,11] in the context of braid groups and extended to Garside groups in [15]. It is crucial for testing conjugacy in Garside groups. More details and proofs of the results quoted in this section can be found in the references above.

Definition 1.11. Let $x \in G$ and denote by x^G the set of conjugates of x. Let $\inf_s(x) = \max\{\inf(y)\colon\ y \in x^G\}$ and $\sup_s(x) = \min\{\sup(y)\colon\ y \in x^G\}$.

The set $S_x = \{y \in x^G\colon\ \inf(y) = \inf_s(x),\ \sup(y) = \sup_s(x)\}$ is called the *super summit set* of x. We define $\operatorname{len}_s(x) = \sup_s(x) - \inf_s(x)$.

Definition 1.12. Let $\delta^k A_1\cdots A_r \in G$ be the normal form of $x \in G$. If $r = 0$, let $\mathbf{c}(x) = \mathbf{d}(x) = x$, otherwise let $\mathbf{c}(x) = x^{\tau^{-k}(A_1)}$ and $\mathbf{d}(x) = x^{A_r^{-1}}$. We call $\mathbf{c}(x)$ and $\mathbf{d}(x)$ the *cycling* of x and the *decycling* of x, respectively.

Theorem 1.13 ([4,8,15]). *Let $x \in G$.*

(a) *S_x is finite and not empty.*
(b) *A representative of S_x can be obtained effectively by applying a finite sequence of cycling and decycling operations to x.*
(c) *If $y \in S_x$ then $\mathbf{c}(y) \in S_x$ and $\mathbf{d}(y) \in S_x$.*
(d) *For all $y \in G$, $\tau(\mathbf{c}(y)) = \mathbf{c}(\tau(y))$ and $\tau(\mathbf{d}(y)) = \mathbf{d}(\tau(y))$.*

The following result is crucial for computing super summit sets.

Theorem 1.14 (El-Rifai, Morton [8], Picantin [15]). *Let $x \in G$.*

(a) *For any $y, z \in S_x$ there exists $u \in M$ such that $y^u = z$.*
(b) *If $y \in S_x$ and $u \in M$ such that $y^u \in S_x$ then $y^{\delta \wedge u} \in S_x$.*
(c) *For any $y, z \in S_x$ there exist elements $y_0, \dots, y_t \in S_x$ and elements $c_1, \dots, c_t \in D$ such that $y_0 = y$, $y_t = z$ and $y_{i-1}^{c_i} = y_i$ for $i = 1, \dots, t$.*

Hence S_x can be computed as follows. First obtain $\tilde{x} \in S_x$ according to Theorem 1.13(b) and set $S = \{\tilde{x}\}$. Now keep conjugating elements of S by simple elements and add those conjugates with infimum $\inf_s(x)$ and supremum $\sup_s(x)$ to S. When no new elements of S_x can be found using this method, that is, $S = \{y^c\colon\ y \in S,\ c \in D,\ y^c \in S_x\}$, then $S = S_x$.

Franco and González-Meneses improved this algorithm as follows.

Theorem 1.15 (Franco, González-Meneses [10]). *Let $x \in G$, $y \in S_x$ and $u, v \in D$. If $y^u \in S_x$ and $y^v \in S_x$ then $y^{u \wedge v} \in S_x$.*

Hence, for an element $y \in S$ in the algorithm outlined above, only the conjugates by those elements which are minimal with respect to $\preccurlyeq$ in the set $\{c \in D\colon\ c \neq 1,\ y^c \in S_x\}$ have to be considered. Franco and González-Meneses remark in [10] that the number of such $\preccurlyeq$-minimal elements is bounded by the number of atoms in M and give an algorithm for computing them.

1.4. Testing conjugacy of elements

Since S_x by definition only depends on the conjugacy class of x, conjugacy of elements x and y of G can be tested as follows [8,10,15].

Compute representatives $\tilde{x}$ of S_x and $\tilde{y}$ of S_y according to Theorem 1.13(b). If $\inf(\tilde{x}) \neq \inf(\tilde{y})$ or $\sup(\tilde{x}) \neq \sup(\tilde{y})$ then x and y are not conjugate. Otherwise, start computing S_x as described in Section 1.3. The elements x and y are conjugate if and only if $\tilde{y} \in S_x$. Note that if x and y are conjugate, an element conjugating x to y can be found by keeping track of the conjugations during the computations of $\tilde{x}$, $\tilde{y}$ and S_x.

Remark 1.16. It is obvious that in the worst case, both the space and the time requirements of the algorithm outlined above are proportional to the cardinality of S_x.

In the cases of the monoids B_n^+ and BKL_n^+, the only known upper bounds for the size of S_x are exponential in n and $\operatorname{len}(x)$. It is conjectured however, that for fixed n, at least for B_n^+ a polynomial bound in $\operatorname{len}(x)$ exists [9].

Nevertheless, the rapidly growing super summit sets make computations in general infeasible for values larger than $n \approx 10$ due to lack of memory.

Note also that distributing the computation of S_x is not practical, as the set S defined in Section 1.3 is constantly accessed and modified by all nodes.

1.5. Ultra summit sets

Definition 1.17. By Theorem 1.13, the super summit set S_x of $x \in G$ can be made into a finite directed graph Γ_x with set of vertices S_x and set of edges $\{(y, \mathbf{c}(y))\colon\ y \in S_x\}$. Obviously, τ induces an automorphism of Γ_x.

Let U_x, the *ultra summit set* of x, be the subset of vertices which are contained in a circuit of Γ_x, that is, $U_x = \{y \in S_x\colon\ \mathbf{c}^k(y) = y \text{ for some } k > 0\}$.

For $y \in S_x$, define the *trajectory* $T_y = \{\mathbf{c}^k(y)\colon\ k \geqslant 0\}$. A representative of U_x can be obtained by computing T_y for an arbitrary $y \in S_x$. For any $z \in T_y$, computing $s_z \in M$ satisfying $y^{s_z} = z$ is straightforward.

The following main result of this paper will be proved in Section 2. It tells us that a "convexity" property analogous to the one established in Theorem 1.14 for super summit sets holds for ultra summit sets, whence the ultra summit set U_x of an element x can be computed as the closure of any non-empty subset U of U_x under conjugation by (minimal) simple elements as outlined in Section 1.3.

Theorem 1.18. *Let $x \in G$, $y \in U_x$ and let $u, v \in M$ such that $y^u \in U_x$ and $y^v \in U_x$. Then $y^{u \wedge v} \in U_x$.*

Corollary 1.19. *Let $x \in G$ and $y, z \in U_x$. There exist elements $y_0, \ldots, y_t \in U_x$ and elements $c_1, \ldots, c_t \in D$ such that $y_0 = y$, $y_t = z$ and $y_{i-1}^{c_i} = y_i$ for $i = 1, \ldots, t$.*

Proof. We may assume $y \neq z$. First note that $y \in U_x$ implies $y^\delta = \tau(y) \in U_x$ as τ is an automorphism of Γ_x. By Theorem 1.14(a), there exists $u \in M$ with $y^u = z$. Let $s = \sup(u)$. Choose $c_1 = \delta \wedge u \in D$ and let $y_1 = y^{c_1}$ and $\tilde{u} = c_1^{-1} u \in M$. By Theorem 1.18 $y_1 \in U_x$. Moreover, $y_1^{\tilde{u}} = z$ and $\sup(\tilde{u}) < s$. Iteration yields $y_1, \ldots, y_t \in U_x$ and $c_1, \ldots, c_t \in D$ as desired. □

Definition 1.20. Let $x \in G$ and $y \in U_x$.

(a) For any $s \in D$, Theorem 1.18 implies the existence of a unique $\preccurlyeq$-minimal element $c_s = c_s(y)$ satisfying $s \preccurlyeq c_s \preccurlyeq \delta$ and $y^{c_s} \in U_x$.
(b) Define $D_y = \{u \in D \setminus \{1\}\colon\ y^u \in U_x\}$ and let C_y be the set of elements of D_y which are $\preccurlyeq$-minimal in D_y. Clearly $C_y \subseteq \{c_a(y)\colon\ a \in A\}$, where A is the set of atoms of M. In particular, $|C_y| \leqslant |A|$.

Corollary 1.21. *Let $x \in G$ and $\emptyset \neq U \subseteq U_x$. If $\{y^c\colon\ y \in U,\ c \in C_y\} \subseteq U$ then $U = U_x$.*

Proof. This follows directly from Corollary 1.19. □

The following result will also be proved in Section 2. It tells us that it is sufficient to test the conjugates of representatives of trajectories when computing the ultra summit set U_x of an element x as the closure of a non-empty subset U of U_x under conjugation by minimal simple elements.

Theorem 1.22. *Let $x \in G$, $y \in U_x$ and $z \in T_y$. For any $s \in C_z$ there exists $t \in C_y$ such that $z^s \in T_{y^t}$.*

Corollary 1.23. *Let $x \in G$, $\emptyset \neq I \subseteq U_x$ and $U = \bigcup_{y \in I} T_y \subseteq U_x$. If $\{y^c \colon c \in C_y\} \subseteq U$ for all $y \in I$ then $U = U_x$.*

Proof. This follows directly from Corollary 1.21 and Theorem 1.22. □

Algorithm 1.24. Given an element x of a Garside group, the following algorithm computes the ultra summit set U_x of x.

Compute $\tilde{x} \in U_x$, set $U = T_{\tilde{x}}$ and $U_0 = \emptyset$.
if $\tilde{x} = \delta^k$ for some k **then**
  return $\{\delta^k\}$
end if
while $U \neq U_0$ **do**
  Let $y_1, \ldots, y_m \in U$ such that $U = U_0 \cup T_{y_1} \cup \cdots \cup T_{y_m}$. Set $U_0 = U$.
  for $y \in \{y_1, \ldots, y_m\}$ **do**
    Compute C_y and set $U = U \cup \bigcup_{c \in C_y} T_{y^c}$. [∗]
  end for
end while
return U

The computation of the set C_y in step [∗] will be discussed in Section 4.

Two elements x and y of G are conjugate in G if and only if $U_x = U_y$, or indeed, if and only if $U_x \cap U_y \neq \emptyset$. Hence, conjugacy of elements x and y of G can be tested, and a conjugating element can be computed, as outlined in Section 1.4, using ultra summit sets instead of super summit sets.

2. Proof of Theorems 1.18 and 1.22

Throughout this section let $x \in G$ be an element of its super summit set with non-zero canonical length, that is, let $\delta^k A_1 \cdots A_r$ be the normal form of x, with $r > 0$, $k = \inf(x) = \inf_s(x)$ and $r + k = \sup(x) = \sup_s(x)$.

We need to understand how the normal forms of conjugates of x are related to the normal form of x.

Proposition 2.1. *Let x be as above and let $u \in M$ such that $x^u \in S_x$. There are elements $u_0, \ldots, u_r$ in M such that $u_0 = \tau^k(u)$, $u_r = u$ and the normal form of x^u is $\delta^k(u_0^{-1} A_1 u_1) \cdots (u_{r-1}^{-1} A_r u_r)$. Here, the factors in brackets are understood to be the simple elements occurring in the normal form of x^u. Explicitly, $u_i = A_{i+1} \cdots A_r u \wedge \delta\tau(A_i^{-1} u_{i-1})$.*

Proof. Let $u_0 = \tau^k(u)$ and $u_r = u$. Define $w_1 = \delta^{-k}x^u = u_0^{-1}A_1\cdots A_r u_r$ and $w_{i+1} = (w_i \wedge \delta)^{-1}w_i$ for $i = 1, \dots, r-1$. By the observation in Definition 1.10, w_i has infimum 0 and canonical length $r+1-i$ and the normal form of x^u is $\delta^k(w_1 \wedge \delta)\cdots(w_r \wedge \delta)$. Assume $u_{i-1} \in M$ has been found such that $w_i = u_{i-1}^{-1}A_i \cdots A_r u_r$. Then, $A_i \preccurlyeq \delta \preccurlyeq u_{i-1}\delta$ implies $u_{i-1}^{-1}A_i \preccurlyeq w_i \wedge \delta$, that is, there is an element $u_i \in M$ such that $w_i \wedge \delta = u_{i-1}^{-1}A_i u_i$. Now $w_{i+1} = (w_i \wedge \delta)^{-1}w_i = u_i^{-1}A_{i+1}\cdots A_r u_r$ and $u_i = A_i^{-1}u_{i-1}(w_i \wedge \delta) = A_{i+1}\cdots A_r u \wedge \delta\tau(A_i^{-1}u_{i-1})$ as claimed. □

Corollary 2.2. *Let x be as above and let $u, v \in M$ such that $x^u \in S_x$ and $x^v \in S_x$. Let $u_0, \dots, u_r$ and $v_0, \dots, v_r$ be the positive elements obtained by applying Proposition* 2.1 *to (x, u) and (x, v), respectively.*

(a) *If $u = \delta$ then $u_i = \delta$ for $i = 0, \dots, r$.*
(b) *If $u \preccurlyeq v$ then $u_i \preccurlyeq v_i$ for $i = 0, \dots, r$. More specifically, if $v = uw$ with $w \in M$ and $w_0, \dots, w_r$ are the positive elements obtained by applying Proposition* 2.1 *to (x^u, w) then $v_i = u_i w_i$ for $i = 0, \dots, r$.*
(c) *If $\sup(u) = b$ then $\sup(u_i) \leqslant b$ for $i = 0, \dots, r$. In particular, if u is simple then u_i is simple for $i = 0, \dots, r$.*
(d) *If $u \wedge v = 1$ then $u_i \wedge v_i = 1$ for $i = 0, \dots, r$.*
(e) *Let $t = u \wedge v$ and let $t_0, \dots, t_r$ be the positive elements obtained by applying Proposition* 2.1 *to (x, t). Then $t_i = u_i \wedge v_i$ for $i = 0, \dots, r$.*

Proof. (a) By Proposition 2.1, we have $u_i = \delta\tau(A_{i+1}\cdots A_r \wedge A_i^{-1}u_{i-1})$. As $u_0 = \delta$ and $A_{i+1}\cdots A_r \wedge A_i^{-1}\delta = 1$ by Definition 1.10, $u_i = \delta$ follows by induction.

(b) $v_0 = u_0 w_0$ is obvious. Assume $v_{i-1} = u_{i-1}w_{i-1}$. By Proposition 2.1,

$$w_i = \left(u_i^{-1}A_{i+1}u_{i+1}\right)\cdots\left(u_{r-1}^{-1}A_r u_r\right)w \wedge \delta\tau\left(\left(u_{i-1}^{-1}A_i u_i\right)^{-1}w_{i-1}\right),$$

whence

$$u_i w_i = A_{i+1}\cdots A_r v \wedge \delta\tau\left(A_i^{-1}v_{i-1}\right) = v_i,$$

again using Proposition 2.1. Hence the claim follows by induction.

(c) Follows from parts (a) and (b), as $\sup(u) \leqslant b$ if and only if $u \preccurlyeq \delta^b$.

(d) $u_0 \wedge v_0 = 1$ is obvious. Assume $u_{i-1} \wedge v_{i-1} = 1$. By Proposition 2.1, $u_i \wedge v_i = A_{i+1}\cdots A_r(u \wedge v) \wedge A_i^{-1}\delta\tau(u_{i-1} \wedge v_{i-1}) = A_{i+1}\cdots A_r \wedge A_i^{-1}\delta = 1$, where in the last step Definition 1.10 was used. Hence the claim follows by induction.

(e) Note that $x^t \in S_x$ by Theorems 1.14(b) and 1.15, that is, Proposition 2.1 can be applied to (x, t). The claim then follows from parts (b) and (d), writing $u = t\bar{u}$ and $v = t\bar{v}$ with $\bar{u} \wedge \bar{v} = 1$. □

Lemma 2.3. *Let x be as above, $u \in M$ such that $x^u \in S_x$. Let $u_0, \dots, u_r$ be the positive elements obtained by applying Proposition* 2.1 *to (x, u). Let $\varphi_x(u) = \tau^{-k}(u_1)$.*

(a) $\varphi_x(u) \in M$ *satisfies* $\mathbf{c}(x^u) = \mathbf{c}(x)^{\varphi_x(u)}$.
(b) $\sup(\varphi_x(u)) \leqslant \sup(u)$. *In particular, if* u *is simple then* $\varphi_x(u)$ *is simple.*
(c) *The conjugating element along any path in the diagram*

$$\begin{tikzcd} x^u \arrow[r, Rightarrow, "\tau^{-k}(u_0^{-1}A_1u_1)"] & \mathbf{c}(x^u) \\ x \arrow[u, "u"] \arrow[r, Rightarrow, "\tau^{-k}(A_1)"] & \mathbf{c}(x) \arrow[u, "\varphi_x(u)"'] \end{tikzcd}$$

only depends on the starting point and the end point of the path. (*Double arrows indicate cycling.*)

Proof. Part (a) follows from $\mathbf{c}(x)^{\tau^{-k}(u_1)} = x^{\tau^{-k}(A_1u_1)} = (x^u)^{\tau^{-k}(u_0^{-1}A_1u_1)} = \mathbf{c}(x^u)$. The conjugating element along the circuit $x \to x^u \to \mathbf{c}(x^u) \to \mathbf{c}(x) \to x$ is $u \cdot \tau^{-k}(u_0^{-1}A_1u_1) \cdot \varphi_x(u)^{-1} \cdot \tau^{-k}(A_1)^{-1} = 1$, proving (c). Part (b) follows from Corollary 2.2(c). □

Definition 2.4. In the situation of Lemma 2.3, we call $\varphi_x(u)$ the *transport* of u along $x \to \mathbf{c}(x)$. If x is obvious from the context, we define $u^{(0)} = u$ and $u^{(i+1)} = \varphi_{\mathbf{c}^i(x)}(u^{(i)})$ for $i \geqslant 0$.

Lemma 2.5. *Let* x *be as above and let* $u, v \in M$ *such that* $x^u = x^v \in S_x$. *If* $\varphi_x(u) = \varphi_x(v)$ *then* $u = v$.

Proof. Let $u_0, \ldots, u_r$ and $v_0, \ldots, v_r$ be the positive elements obtained by applying Proposition 2.1 to (x, u) and (x, v), respectively. As $x^u = x^v$, we have $(u_0^{-1}A_1u_1) = \delta \wedge \delta^{-k}x^u = \delta \wedge \delta^{-k}x^v = (v_0^{-1}A_1v_1)$. The claim then follows from Lemma 2.3(c). □

Lemma 2.6. *Let* x *be as above, let* $u \in M$ *such that* $x^u \in S_x$ *and let* $\mathbf{c}^N(x) = x$ *and* $\mathbf{c}^N(x^u) = x^u$ *for some integer* $N > 0$. *There is an integer* $m > 0$ *such that* $u^{(mN)} = u$, *where we use the notation from Definition* 2.4.

Proof. By Lemma 2.3(b), $u^{(iN)} \in M$ and $\sup(u^{(iN)}) \leqslant \sup(u)$ for every integer $i \geqslant 0$. Since the number of such elements is at most $|D|^{\sup(u)}$, in particular finite, there must exist integers $i_2 > i_1 \geqslant 0$ such that $u^{(i_1N)} = u^{(i_2N)}$; let i_2 be minimal subject to this condition. Assume $i_1 > 0$. Then we can for $l = 1, \ldots, N$ conclude $u^{(i_1N-l)} = u^{(i_2N-l)}$ from

$$\begin{aligned} \varphi_{\mathbf{c}^{(N-l)}(x)}\big(u^{(i_1N-l)}\big) &= \varphi_{\mathbf{c}^{(i_1N-l)}(x)}\big(u^{(i_1N-l)}\big) = u^{(i_1N-(l-1))} = u^{(i_2N-(l-1))} \\ &= \varphi_{\mathbf{c}^{(i_2N-l)}(x)}\big(u^{(i_2N-l)}\big) = \varphi_{\mathbf{c}^{(N-l)}(x)}\big(u^{(i_2N-l)}\big), \end{aligned}$$

using Lemma 2.5. In particular, $u^{((i_1-1)N)} = u^{((i_2-1)N)}$, contradicting the minimality of i_2. Hence, $i_1 = 0$ and $u^{(i_2N)} = u^{(0)} = u$ as claimed. □

Corollary 2.7. *Let x be as above and let $y \in U_x$. Recall the sets D_y and C_y introduced in Definition* 1.20*.*

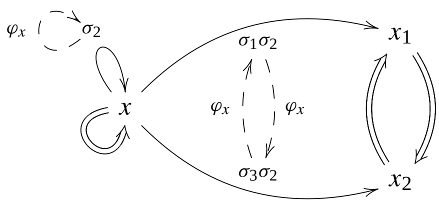
(a) *The restriction $\varphi_y|_{D_y \cup \{1\}} : D_y \cup \{1\} \to D_{\mathbf{c}(y)} \cup \{1\}$ is a bijection.*
(b) *The restriction $\varphi_y|_{C_y} : C_y \to C_{\mathbf{c}(y)}$ is a bijection.*

Proof. Claim (a) follows from Lemma 2.6. By Corollary 2.2(b), $\varphi(u) \preccurlyeq \varphi(v)$ if and only if $u \preccurlyeq v$ holds for all $u, v \in D_y \cup \{1\}$, yielding claim (b). $\square$

Example 2.8. It is worth pointing out that two trajectories in U_x do not necessarily have the same length and that the integer m from Lemma 2.6 can be greater than 1. We illustrate this with a very simple example.

Consider $x = \delta \cdot \sigma_3\sigma_2\sigma_1\sigma_2\sigma_3 \in B_4^+$. As $\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3$ is simple, x is in normal form as written; in particular, $\mathrm{len}_s(x) = 1$. Hence $\mathbf{c}(y) = y^\delta$ and $\mathbf{c}^2(y) = y$ for all $y \in S_x$, that is, $U_x = S_x$. Using the results cited in Section 1.3 and taking advantage of the equality $U_x = S_x$ it is easy to compute the sets U_x and C_x. We obtain $U_x = \{x, x_1, x_2\}$, where $x_1 = x^{\sigma_1\sigma_2} = \delta \cdot \sigma_2\sigma_1\sigma_3\sigma_2\sigma_3$ and $x_2 = x^{\sigma_3\sigma_2} = \delta \cdot \sigma_3\sigma_2\sigma_1\sigma_3\sigma_2 = x_1^\delta$, and $C_x = \{\sigma_2, \sigma_1\sigma_2, \sigma_3\sigma_2\}$. In particular, $\mathbf{c}(x) = x$, $\mathbf{c}(x_1) = x_2$ and $\mathbf{c}(x_2) = x_1$, that is, U_x consists of two trajectories under cycling which have different sizes.

The structure of U_x and the conjugations of x by elements of C_x are given in the following diagram; double arrows indicate cycling.



The transport map φ_x induces a bijection on the set $C_x = C_{\mathbf{c}(x)}$. Note that $\varphi_x(\sigma_1\sigma_2) = \sigma_3\sigma_2$, that is, transporting $s = \sigma_1\sigma_2$ once along the trajectory of x does not fix s. However, $s^{(2)} = s$, that is, $m = 2$ in Lemma 2.6.

We remark that examples with more than two trajectory lengths and values of $m > 2$ exist.

Theorem 2.9. *Let x be as above, $u, v \in M$ such that $u \wedge v = 1$. If $x^u \in U_x$ and $x^v \in U_x$ then $x \in U_x$.*

Proof. First note that we may assume that $\mathbf{c}(x) \in U_x$, since if x is a counterexample with $\mathbf{c}(x) \notin U_x$, consider $\bar{x} = \mathbf{c}(x) \in S_x$, $\bar{u} = \varphi_x(u)$ and $\bar{v} = \varphi_x(v)$. Clearly, $\bar{x}^{\bar{u}} = \mathbf{c}(x^u) \in U_x$ and $\bar{x}^{\bar{v}} = \mathbf{c}(x^v) \in U_x$. Moreover, $\bar{u} \wedge \bar{v} = 1$ by Corollary 2.2(d). Repeating this process finitely many times, we arrive at a counterexample x with $\mathbf{c}(x) \in U_x$.

Choose $N > 0$ such that $\mathbf{c}^N(x^u) = x^u$, $\mathbf{c}^N(x^v) = x^v$, and $\mathbf{c}^{N+1}(x) = \mathbf{c}(x)$. We use the notation from Definition 2.4. According to Lemma 2.6, we can further assume that

$u^{(N+1)} = u^{(1)}$ and $v^{(N+1)} = v^{(1)}$, replacing N by a suitable multiple if necessary. Now consider the conjugations by the conjugating elements indicated in the following diagram where double arrows indicate cycling.

$$\begin{tikzcd}
x^u \arrow[r, Rightarrow, "\alpha_u"] & \mathbf{c}(x^u) \arrow[r, Rightarrow] & \cdots \arrow[r, Rightarrow] & \mathbf{c}^N(x^u) = x^u \arrow[r, Rightarrow, "\beta_u"] & \mathbf{c}(x^u) \\
x \arrow[u, "u"] \arrow[d, "v"'] \arrow[r, Rightarrow, "\alpha"] & \mathbf{c}(x) \arrow[u, "u^{(1)}"'] \arrow[d, "v^{(1)}"] \arrow[r, Rightarrow] & \cdots \arrow[r, Rightarrow] & \mathbf{c}^N(x) \arrow[u, "u^{(N)}"'] \arrow[d, "v^{(N)}"] \arrow[r, Rightarrow, "\beta"] & \mathbf{c}^{N+1}(x) = \mathbf{c}(x) \arrow[u, "u^{(N+1)}"'] \arrow[d, "v^{(N+1)}"] \\
x^v \arrow[r, Rightarrow, "\alpha_v"] & \mathbf{c}(x^v) \arrow[r, Rightarrow] & \cdots \arrow[r, Rightarrow] & \mathbf{c}^N(x^v) = x^v \arrow[r, Rightarrow, "\beta_v"] & \mathbf{c}(x^v)
\end{tikzcd}$$

Obviously, $\alpha_u = \tau^{-k}(\delta \wedge \delta^{-k} x^u) = \beta_u$ and $\alpha_v = \tau^{-k}(\delta \wedge \delta^{-k} x^v) = \beta_v$ and by Corollary 2.2(d), we have $u^{(i)} \wedge v^{(i)}$ for $i = 1, \ldots, N$. Hence,

$$\begin{aligned}
\alpha^{-1} &= \alpha^{-1}(u \wedge v) = \alpha^{-1}u \wedge \alpha^{-1}v = u^{(1)}\alpha_u^{-1} \wedge v^{(1)}\alpha_v^{-1} \\
&= u^{(N+1)}\beta_u^{-1} \wedge v^{(N+1)}\beta_v^{-1} = \beta^{-1}u^{(N)} \wedge \beta^{-1}v^{(N)} = \beta^{-1}
\end{aligned}$$

where we used Lemma 2.3(c) four times. We conclude $x \in U_x$ from

$$x = \mathbf{c}(x)^{\alpha^{-1}} = \mathbf{c}(x)^{\beta^{-1}} = \left(\mathbf{c}^{N+1}(x)\right)^{\beta^{-1}} = \mathbf{c}^N(x). \qquad \square$$

Theorems 1.18 and 1.22 now follow easily.

Theorem 1.18. *Let $x \in G$, $y \in U_x$ and let $u, v \in M$ such that $y^u \in U_x$ and $y^v \in U_x$. Then $y^{u \wedge v} \in U_x$.*

Proof. If $\inf_s(x) = \sup_s(x) = k$ then $U_x = S_x = \{\delta^k\}$ and the claim follows from Theorems 1.14(b) and 1.15. Hence assume $\sup_s(x) > \inf_s(x)$.

Let $t = u \wedge v$. Then $u = t\bar{u}$, $v = t\bar{v}$ with $\bar{u} \wedge \bar{v} = 1$. By Theorems 1.14(b) and 1.15, $y^t \in S_x$. As $(y^t)^{\bar{u}} = y^u \in U_x$ and $(y^t)^{\bar{v}} = y^v \in U_x$, Theorem 2.9 implies $y^t \in U_x$. $\square$

Theorem 1.22. *Let $x \in G$, $y \in U_x$ and $z \in T_y$. For any $s \in C_z$ there exists $t \in C_y$ such that $z^s \in T_{y^t}$.*

Proof. By Corollary 2.7(b), $C_{\mathbf{c}(y)} = \{\varphi_y(u)\colon u \in C_y\}$. The claim follows by induction. $\square$

3. A probabilistic approach to the conjugacy search problem

Given elements $x, y \in G$ which are conjugate in G, we can use the structure of the graph Γ_x for computing an element $s \in G$ satisfying $x^s = y$ without having to compute the entire ultra summit set U_x.

Applying cycling and decycling operations to x and y, respectively, we can obtain $\tilde{x}, \tilde{y} \in U_x = U_y$ as well as $s_x, s_y \in G$ satisfying $x^{s_x} = \tilde{x}$ and $y^{s_y} = \tilde{y}$. For $z \in T_{\tilde{x}}$, that is, $z = \mathbf{c}^k(\tilde{x})$ for some k, let $s(z)$ satisfy $\tilde{x}^{s(z)} = z$.

Algorithm 3.1. Given a Garside group G and elements $x, y \in G$ which are conjugate in G, the following Las Vegas algorithm computes an element $s \in G$ such that $x^s = y$.

Compute $\tilde{x}$, s_x, $T_{\tilde{x}}$ and $\{s(z)\colon z \in T_{\tilde{x}}\}$ as above.
Compute $\tilde{y}$ and s_y as above. Set $z = \tilde{y}$ and $s = s_y$.
loop
  if $z \in T_{\tilde{x}}$ **then**
    return $s_x \cdot s(z) \cdot s^{-1}$
  end if
  Choose a random atom a of M. Compute $c_a = c_a(z)$. [∗]
  Set $z = z^{c_a}$, $s = s \cdot c_a$.
end loop

The computation of c_a in step [∗] (recall Definition 1.20) will be discussed in Section 4.

Remark 3.2. The expected number of iterations of the loop in Algorithm 3.1 is the number of circuits of the graph Γ_x. This loop can easily be parallelised, since no communication between nodes is necessary.

4. Computing minimal elements

Throughout this section let $x \in G$ be an element of its ultra summit set with normal form $\delta^k A_1 \cdots A_r$, where $r > 0$, and let N be the minimal positive integer satisfying $\mathbf{c}^N(x) = x$.

In this section we show how the elements $c_s = c_s(x)$ $(s \in D)$ and the set C_x introduced in Definition 1.20 can be computed efficiently.

For any $s \in D$, Theorem 1.15 implies the existence of a unique $\preccurlyeq$-minimal element $\rho_s = \rho_s(x)$ satisfying $s \preccurlyeq \rho_s \preccurlyeq \delta$ and $x^{\rho_s} \in S_x$. An algorithm for computing ρ_s is given in [10]. Obviously, if $s = 1$ then $c_s = \rho_s = 1$.

Note that $\rho_s \preccurlyeq c_s$ since $U_x \subseteq S_x$. We know from Lemma 2.6 that c_s is in a period under transport. We will show that c_s can be computed by applying iterated transport to a suitable element derived from ρ_s until this period is reached.

Definition 4.1. Let $u \in D$ such that $x^u \in S_x$. Using the notation from Definition 2.4, we consider the elements $u^{(iN)}$ $(i \geqslant 0)$. By Lemma 2.3(b) and since D is finite, there are integers $i_2 > i_1 \geqslant 0$ such that $u^{(i_1 N)} = u^{(i_2 N)}$. Let i_1 and i_2 be minimal subject to this condition and define $l_x(u) = i_2 - i_1$ and $F_x(u) = \{u^{(iN)}\colon i_1 \leqslant i < i_2\}$.

Note that $1 \in F_x(u)$ if and only if $F_x(u) = \{1\}$. Moreover, if $x^u \in U_x$ then $i_1 = 0$ by Lemma 2.6, that is, $u \in F_x(u)$.

Lemma 4.2. *Let $u \in D$ such that $x^u \in S_x$, let $v \in F_x(u)$ and let $l = l_x(u)$. Then, $v^{(ilN)} = v$ for all integers $i > 0$. Moreover, $x^v \in U_x$.*

Proof. As $v^{(lN)} = v$, the first claim follows by induction. For the second claim note that $\mathbf{c}^{lN}(x^v) = x^{(v^{(lN)})} = x^v$, whence $x^v \in U_x$. □

Lemma 4.3. *Let $s \in D$. If $c_s \preccurlyeq c_s^{(iN)}$ for some $i > 0$ then $c_s^{(iN)} = c_s$.*

Proof. Let $c_s^{(iN)} = c_s\gamma$ with $\gamma \in M$. By induction, $c_s\gamma \preccurlyeq c_s^{(\beta iN)}$ for all $\beta \geqslant 1$ from Corollary 2.2(b). Using Lemma 4.2, this in particular implies $c_s \preccurlyeq c_s\gamma \preccurlyeq c_s^{(l_x(c_s)iN)} = c_s$, that is, $\gamma = 1$. □

Lemma 4.4. *Let $p, s \in D$ satisfy $p \preccurlyeq c_s$ and $x^p \in S_x$. Let $F = F_x(p)$.*

(a) *If there exists $v \in F$ such that $s \preccurlyeq v$ then $c_s = v$.*
(b) *If $F \neq \{1\}$ and $s \not\preccurlyeq v$ for all $v \in F$ then c_s is not $\preccurlyeq$-minimal in D_x.*

Proof. First note that by Corollary 2.2(b), $p^{(i)} \preccurlyeq c_s^{(i)}$ for all $i > 0$.

(a) As $s \preccurlyeq v$ and $x^v \in U_x$ by Lemma 4.2, minimality of c_s implies $c_s \preccurlyeq v$. Now $v = p^{(iN)}$ for some i, whence $c_s \preccurlyeq v = p^{(iN)} \preccurlyeq c_s^{(iN)}$. Lemma 4.3 yields $v = c_s$.

(b) Let i be a multiple of $l_x(c_s)$ sufficiently large so that $v = p^{(iN)} \in F$. Since $1 \notin F$, we have $v \in D_x$ by Lemma 4.2 and Corollary 2.2(c). Moreover, again using Lemma 4.2, $v = p^{(iN)} \preccurlyeq c_s^{(iN)} = c_s$ and $v \neq c_s$, since $s \not\preccurlyeq v$. □

Example 4.5. Consider $x = \sigma_3\sigma_2\sigma_1\sigma_2\sigma_3 \cdot \sigma_3 \in B_4^+$, in normal form as written, and $s = \sigma_1$. It is easy to check that $\mathbf{c}^3(x) = \mathbf{d}^3(x) = x$, that is, $x \in U_x$. Since $x^s = \sigma_3\sigma_2\sigma_1\sigma_3 \cdot \sigma_1\sigma_3$ is in normal form as written, $x^s \in S_x$, that is, $\rho_s = s$.

However, from $\mathbf{c}(x) = \mathbf{c}(x^s) = \sigma_1\sigma_3\sigma_2\sigma_3 \cdot \sigma_2\sigma_1$, we obtain $s^{(1)} = \varphi_x(s) = 1$ and hence $F_x(s) = \{1\}$, that is, the requirements of Lemma 4.4 are not satisfied.

Example 4.5 shows that there are situations in which it is impossible to compute c_s or to prove that c_s is not $\preccurlyeq$-minimal in D_x by iterated transport of ρ_s. The solution is to apply iterated transport not to ρ_s itself, but to a related element p for which the existence of $v \in F_x(p)$ with $s \preccurlyeq v$ is guaranteed.

Definition 4.6. Let $s \in D$ and let $y \in U_x$. By Theorems 1.14(b) and 1.15 and Corollary 2.2, (a) and (e), there exists a unique $\preccurlyeq$-minimal element $\pi_y(s) \in D$ satisfying $y^{\pi_y(s)} \in S_x$ and $s \preccurlyeq \varphi_y(\pi_y(s))$. We call $\pi_y(s)$ the *pullback* of s along $y \to \mathbf{c}(y)$. If y is obvious from the context, we define $s_{(0)} = s$ and $s_{(i+1)} = \pi_{\mathbf{c}^\alpha(y)}(s_{(i)})$ for $i \geqslant 0$, where $0 \leqslant \alpha \equiv -i \pmod N$.

Proposition 4.7. *Let $s \in D$ and let $\delta^k B_1 \cdots B_r$ be the normal form of $y \in U_x$. Define*

$$b = \left(1 \vee \tau^{-k}(B_1)s\delta^{-1}\right) \vee \left(1 \vee B_r^{-1} \cdots B_2^{-1}\tau^k(s)\right).$$

Then $b \in D$ and $\rho_b = \pi_y(s)$.

Proof. We show that $s \preccurlyeq \varphi_y(t)$ is equivalent to $b \preccurlyeq t$ for any $t \in M$ satisfying $y^t \in S_x$. Then $\rho_b = \pi_y(s)$ follows directly from the definitions of ρ_b and $\pi_y(s)$. Moreover, $1 \preccurlyeq b \preccurlyeq \rho_b = \pi_y(s) \in D$, that is, $b \in D$.

By Proposition 2.1, $\tau^k(\varphi_y(t)) = (B_2 \cdots B_r t) \wedge (B_1^{-1}\tau^k(t)\delta)$ for any $t \in M$ satisfying $y^t \in S_x$. Hence $s \preccurlyeq \varphi_y(t)$ if and only if $\tau^k(s) \preccurlyeq B_2 \cdots B_r t$ and $\tau^k(s) \preccurlyeq B_1^{-1}\tau^k(t)\delta$, which, in turn, is equivalent to $B_r^{-1} \cdots B_2^{-1}\tau^k(s) \preccurlyeq t$ and $\tau^{-k}(B_1)s\delta^{-1} \preccurlyeq t$. As $t \in M$, the latter is equivalent to $b \preccurlyeq t$. $\square$

Remark 4.8. We can easily compute b as in Proposition 4.7 as $b = b_0 \vee b_r$, where

$$b_0 = 1 \vee \tau^{-k}(B_1)s\delta^{-1} = \tau^{-1}\bigl(\tau^{-k}\bigl(B_1^{-1}\delta\bigr)^{-1} \cdot \bigl(\tau^{-k}\bigl(B_1^{-1}\delta\bigr) \vee s\bigr)\bigr), \qquad b_1 = \tau^k(s) \quad \text{and}$$

$$b_i = 1 \vee B_i^{-1}b_{i-1} = B_i^{-1} \cdot (B_i \vee b_{i-1}) \quad \text{for } i = 2, \ldots, r.$$

In particular, all computations can be performed in the set D of simple elements.

Proposition 4.9. *Let $s \in D$ and consider for $i \geqslant 0$ the elements $s_{(iN)}$ obtained by applying Definition 4.6 for $y = \mathbf{c}^{N-1}(x)$. As D is finite, there are integers $i_2 > i_1 \geqslant 0$ such that $s_{(i_1N)} = s_{(i_2N)}$. Choose minimal values for i_1 and i_2, let $l = i_2 - i_1$ and choose an integer j such that $jl \geqslant i_1$. Finally, let $p = p_x(s) = s_{(jlN)}$.*

Then, $p \preccurlyeq c_s$ and there exists $v \in F_x(p)$ with $s \preccurlyeq v$. In particular, $v = c_s$.

Proof. Let $\beta \geqslant j$ be a multiple of $l_x(c_s)$ large enough such that $p^{(\beta lN)} \in F_x(p)$. Let $v = p^{(\beta lN)}$. By Definition 4.6 and Corollary 2.2(b), $p = s_{(jlN)} = s_{(\beta lN)}$ is the unique $\preccurlyeq$-minimal element satisfying $x^p \in S_x$ and $s \preccurlyeq p^{(\beta lN)}$. Since $x^{c_s} \in U_x \subseteq S_x$ and $s \preccurlyeq c_s = c_s^{(\beta lN)}$, we have $p \preccurlyeq c_s$. By Lemma 4.4, $v = c_s$. $\square$

Example 4.10. Consider the situation from Example 4.5. The trajectory of x under cycling has length 3; $\mathbf{c}^3(x) = x = \sigma_3\sigma_2\sigma_1\sigma_2\sigma_3 \cdot \sigma_3$, $\mathbf{c}(x) = \sigma_1\sigma_3\sigma_2\sigma_3 \cdot \sigma_2\sigma_1$ and $\mathbf{c}^2(x) = \sigma_2\sigma_1\sigma_3 \cdot \sigma_1\sigma_2\sigma_3$ in normal form.

We compute iterated pullbacks of $s = s_{(0)} = \sigma_1$ and obtain $s_{(1)} = \sigma_2\sigma_1$, $s_{(2)} = \sigma_3$, $s_{(3)} = \sigma_1\sigma_2$, $s_{(4)} = \sigma_2\sigma_1$, $s_{(5)} = \sigma_3$ and $s_{(6)} = \sigma_1\sigma_2$. Hence, using the notation from Proposition 4.9, $p = p_x(s) = s_{(3)} = \sigma_1\sigma_2$.

Next we compute iterated transports of $p = p^{(0)} = \sigma_1\sigma_2$. We obtain $p^{(1)} = \sigma_3$, $p^{(2)} = \sigma_2\sigma_1$, $p^{(3)} = \sigma_1\sigma_2\sigma_3$, $p^{(4)} = \sigma_3$, $p^{(5)} = \sigma_2\sigma_1$ and $p^{(6)} = \sigma_1\sigma_2\sigma_3$. Hence, $F_x(p) = \{p^{(3)}\}$ and as $s \preccurlyeq p^{(3)}$, we obtain $c_s = p^{(3)} = \sigma_1\sigma_2\sigma_3$.

Note that $p \notin F_x(p)$, that is, computing iterated transports is necessary even after reaching a stable loop under iterated pullback. We further note that σ_1 is the only atom $a \in B_4^+$ satisfying $a \preccurlyeq c_s$ and that c_s is $\preccurlyeq$-minimal in D_x.

The following result gives another sufficient condition for identifying c_s as not $\preccurlyeq$-minimal in D_x which can be used to speed up the computation of C_x; see Algorithm 4.12.

Lemma 4.11. *Let $p, s \in D \setminus \{1\}$ such that $x^p \in S_x$. If there exists an integer $i > 0$ such that $p^{(i)} = 1$ then $p \wedge \tau^{-k}(A_1) \neq 1$.*

If moreover $p \preccurlyeq c_s$ and $c_s \not\preccurlyeq \tau^{-k}(A_1)$ then c_s is not $\preccurlyeq$-minimal in D_x.

Proof. If $p^{(1)} = 1$ then Proposition 2.1 implies $\tau^k(p) \preccurlyeq A_1$. Thus we assume $p^{(1)} \neq 1$ and $i > 1$. Let $\delta^k B_1 \cdots B_r$ be the normal form of $\mathbf{c}(x) = x^{\tau^{-k}(A_1)}$. According to Proposition 2.1, $(\tau^{-k}(A_1))^{(1)} = \varphi_x(\tau^{-k}(A_1)) = \tau^{-k}(B_1)$. By induction $(p^{(1)})^{(i-1)} = p^{(i)} = 1$ yields

$$\big(p \wedge \tau^{-k}(A_1)\big)^{(1)} = p^{(1)} \wedge \big(\tau^{-k}(A_1)\big)^{(1)} = p^{(1)} \wedge \tau^{-k}(B_1) \neq 1$$

using Corollary 2.2(e). This completes the proof of the first claim.

Let $c = c_s \wedge \tau^{-k}(A_1) \preccurlyeq c_s$. If $c_s \not\preccurlyeq \tau^{-k}(A_1)$ then $c \neq c_s$. Now $p \preccurlyeq c_s$ implies $c \neq 1$ and $c \in D_x$ by Theorem 1.18, since $\mathbf{c}(x) = x^{\tau^{-k}(A_1)} \in U_x$. □

Algorithm 4.12. Given $s \in D$ and a boolean value `f` indicating whether elements which are known not to be $\preccurlyeq$-minimal in D_x should be discarded, the following algorithm returns c_s or identifies it as not $\preccurlyeq$-minimal in D_x.

Compute ρ_s as described in [10] and compute $F_x(\rho_s)$.
if $\exists\, v \in F_x(\rho_s)$ such that $s \preccurlyeq v$ **then**
  return v
end if
if `f` **and** $F_x(\rho_s) \neq \{1\}$ **then**
  return `not minimal`
end if
Compute $p_x(s)$ and $F_x(p_x(s))$. [∗]
Choose $v \in F_x(p_x(s))$ such that $s \preccurlyeq v$.
return v

In the case that `f` is `true`, the algorithm can be aborted returning `not minimal` in step [∗] if c_s is at any point found to be not $\preccurlyeq$-minimal in D_x by Lemma 4.11.

Remark 4.13. A superset of C_x whose cardinality is bounded by the number of atoms of M can be computed using Algorithm 4.12 with `f = true`, by letting s range over all atoms of M. Obvious short-cuts, similar to the ones described in [10], can be used to increase the efficiency of this process.

Remark 4.14. By Proposition 4.9, we could skip both `if` statements in Algorithm 4.12 and start with step [∗]. The reason for not doing this in practice is that computing pullbacks is relatively expensive and frequently not necessary.

5. Practical comparisons

In this section, we present empirical results for Artin braid groups B_n given by the presentation (1) from Section 1.1.

For several values of n and r, we consider a set of elements $x \in B_n$ with $\mathrm{len}_s(x) = r$, chosen at random, and compute for each such x its super summit set S_x and its ultra summit set U_x. Let t_S and t_U be the times spent on computing S_x and U_x, respectively, and let n_U be the number of trajectories under cycling of which U_x consists. We compare the average and maximal values of $|S_x|$, $|U_x|$, t_S, t_U and n_U. (See Tables 1 and 2.)

Random elements for these tests were obtained as follows. We choose independent random simple elements $A_1, A_2, \ldots$ until $\mathrm{len}(A_1 \cdots A_m) = r$, choose a random integer

Table 1
Average/maximal values for $|U_x|$, $|S_x|$, the time t_U for computing U_x, the time t_S for computing S_x and the number n_U of cycling orbits of U_x for various values of braid index n and canonical length r

n	3					
r	2	5	10	20	100	1000
$\|U_x\|$	3.1/4	9.8/10	20/20	40/40	200/200	2000/2000
$\|S_x\|$	3.1/4	9.8/10	20/20	40/40	200/200	2000/2000
t_U	0.1/10	0.2/10	0.4/11	1.1/11	22/31	4.1 s/5.4 s
t_S	0.1/9	0.3/10	1.0/11	3.4/11	79/90	15 s/19 s
n_U	1.2/2	1.5/2	1.5/2	1.5/2	1.4/2	1.6/2
n	4					
r	2	5	10	20	100	1000
$\|U_x\|$	5.6/10	12/50	20/40	40/40	200/200	2000/2000
$\|S_x\|$	11/24	47/128	100/464	190/660	920/1704	9000/1.0e4
t_U	0.2/11	0.5/11	0.7/11	1.8/11	45/81	7.8 s/13.5 s
t_S	0.4/11	2.6/11	9.2/51	29/121	650/1250	210 s/272 s
n_U	1.6/3	1.7/10	1.5/8	1.5/2	1.5/2	1.6/2
n	6					
r	2	5	10	20	100	1000
$\|U_x\|$	15/72	17/1440	21/60	40/40	200/200	2000/2000
$\|S_x\|$	270/1004	3800/8.3e4	1.1e4/2.9e5	–	–	–
t_U	1.3/11	1.9/151	1.6/30	3.1/20	53/90	5.2 s/12 s
t_S	18/71	600/15 s	24 s/672 s	–	–	–
n_U	3.1/18	2.6/262	1.5/4	1.5/2	1.4/2	1.6/2
n	8					
r	2	5	10	20	100	1000
$\|U_x\|$	43/448	14/188	21/56	40/40	200/200	2000/2000
$\|S_x\|$	1.3e4/7.3e4	–	–	–	–	–
t_U	4.9/59	2.5/80	1.9/40	4.7/11	67/150	7.7 s/17 s
t_S	27 s/165 s	–	–	–	–	–
n_U	6.9/64	2.7/94	1.5/2	1.5/2	1.5/2	1.4/2

Times are given in ms, unless stated otherwise. Where no values of $|S_x|$ and t_S are given, computing super summit sets exceeded the available memory of 512 MB. The size of the samples was 1000 for $r \leqslant 100$ and 100 for $r = 1000$.

$k \in \{0, 1\}$ and compute $x = \delta^k \cdot A_1 \cdots A_m$. If $\mathrm{len_s}(x) = r$, we use the element x, otherwise we discard x and try again. (See Remark 5.1.) Note that δ^2 is central in B_n, whence there is a natural isomorphism of the graphs Γ_x and $\Gamma_{\delta^{2m}x}$ for arbitrary m. Our choice of k thus is no restriction.

In a second series of tests we consider for several values of n and r a set of elements $x = \delta^k \cdot A_1 \cdots A_r \in B_n$ obtained by choosing a random integer $k \in \{0, 1\}$ and independent random simple elements $A_1, \ldots, A_r$. We compare the average values of $\mathrm{len}(x)$ and $\mathrm{len_s}(x)$, as well as the percentages ϵ_S and ϵ_U of elements x satisfying $x \in S_x$ and $x \in U_x$, respectively. (See Table 3.)

All computations were performed on a Linux PC with a 2.4 GHz Pentium 4 CPU, 533 MHz system bus and 512 MB of RAM using the author's implementation in C, which is part of the computational algebra system MAGMA [5].

5.1. Results

The main results of the tests can be summarised as follows.

(a) The average size of S_x grows very fast with increasing values of n. S_x is in general not computable on typical current computers for $n \geqslant 10$ or $n > 5$, $r > 15$, due to extreme memory requirements.

Table 2
Average/maximal values for $|U_x|$, the time t_U for computing U_x and the number n_U of cycling orbits of U_x for various values of braid index n and canonical length r

n	10					
r	2	5	10	20	100	1000
$\|U_x\|$	63/1408	15/54	21/40	40/78	200/200	2000/2000
t_U	12/290	3.3/21	4.2/40	6.3/90	100/190	16 s/32 s
n_U	11/104	2.0/8	1.5/4	1.6/2	1.5/2	1.5/2
n	20					
r	2	5	10	20	100	1000
$\|U_x\|$	30/280	12/20	20/40	40/40	200/200	2000/2000
t_U	10/151	3.4/11	4.7/11	9.7/21	100/221	19 s/46 s
n_U	7.7/70	1.9/4	1.5/4	1.5/2	1.6/2	1.5/2
n	50					
r	2	5	10	20	100	1000
$\|U_x\|$	7.0/64	10/20	20/20	40/40	200/200	2000/2000
t_U	7.8/50	8.4/21	12/21	18/30	130/241	21 s/48 s
n_U	2.3/16	1.6/4	1.5/2	1.5/2	1.5/2	1.6/2
n	100					
r	2	5	10	20	100	1000
$\|U_x\|$	5.2/32	10/10	20/20	40/40	200/200	2000/2000
t_U	20/101	27/50	36/61	49/69	210/370	23 s/32 s
n_U	1.7/8	1.4/2	1.5/2	1.6/2	1.5/2	1.5/2

Times are given in ms, unless stated otherwise. For all parameter values in this table computing super summit sets exceeded the available memory of 512 MB. The size of the samples was 1000 for $r \leqslant 100$ and 100 for $r = 1000$.

Table 3
Average values of $\mathrm{len}(x)$ and $\mathrm{len}_s(x)$ and percentages ϵ_S and ϵ_U of pseudo-random elements x satisfying $x \in S_x$ and $x \in U_x$, respectively, for various values of braid index n and number of simple factors r

n	3						4					
r	2	5	10	20	100	1000	2	5	10	20	100	1000
$\mathrm{len}(x)$	1.0	1.8	2.7	4.7	19	170	1.4	2.7	4.5	7.8	34	330
$\mathrm{len}_s(x)$	0.8	1.4	2.1	3.7	17	170	1.2	2.1	3.6	6.6	33	330
ϵ_S	89	72	64	56	52	51	77	53	41	36	32	32
ϵ_U	89	72	64	56	52	51	72	40	22	11	8.7	8.0
n	6						10					
r	2	5	10	20	100	1000	2	5	10	20	100	1000
$\mathrm{len}(x)$	1.9	3.8	6.7	12	58	570	2.0	4.8	9.0	17	85	840
$\mathrm{len}_s(x)$	1.6	3.1	5.6	11	57	570	2.0	4.3	8.4	17	84	840
ϵ_S	77	42	33	32	32	31	96	63	55	51	54	53
ϵ_U	30	4.0	1.1	0.9	1.0	1.4	1.4	0.3	0.0	0.0	0.1	0.0
n	15						30, 50, 75, 100					
r	2	5	10	20	100	1000	2	5	10	20	100	1000
$\mathrm{len}(x)$	2.0	5.0	9.9	20	98	980	2.0	5.0	10	20	100	1000
$\mathrm{len}_s(x)$	2.0	4.9	9.8	20	98	980	2.0	5.0	10	20	100	1000
ϵ_S	100	94	89	87	88	87	100	100	100	100	100	100
ϵ_U	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

The size of each sample was 1000.

(b) With the exception of very small values of r ($r = 2, 5$), the average size of U_x is of the order of $2r$, in particular almost independent of n, for the case of presentation (1) from Section 1.1. Similar tests for presentation (2) yield an average size of the order of nr for not too small values of r.

There are, however, elements whose ultra summit sets are much larger than the average values. With growing values of n and r, these exceptions seem to get rarer, so in some sense the situation then becomes easier.

In the tests, U_x remained sufficiently small to be computed easily over the entire parameter range.

(c) The average number of connected components (trajectories) of U_x is approximately 1.5 for larger values of r. Note that this implies that computing conjugating elements by Algorithm 3.1 is very efficient.

Another consequence of this is that even in the case $n = 3$ where $U_x = S_x$, computing U_x is much faster than computing S_x for large values of r, since the decomposition of U_x into trajectories is used efficiently (Theorem 1.22).

(d) A random element of the form $\delta^k \cdot A_1 \cdots A_r$ with independent random simple elements $A_1, \ldots, A_r$ is surprisingly likely to be a super summit element, that is, satisfy $x \in S_x$. In the tests for $n > 20$, the probability for this is indistinguishable from 1 and the elements moreover satisfy $\mathrm{len}_s(x) = r$.

Random elements as above which are ultra summit elements, on the other hand, are very rare for $n > 5$ and were not encountered at all in the tests for $n > 20$.

This suggests that, with the exception of braid groups on very few strings, the ultra summit set of an element, in general, is a very small subset of the super summit set.

Remark 5.1. Other methods of constructing pseudo-random elements may produce different distributions on the set of all elements $x \in B_n$ satisfying $\mathrm{len}_\mathrm{s}(x) = r$ and $x \in S_x$. However, at least for larger values of the braid index n, according to our results a product x of a random power of δ and r independently chosen random simple elements is extremely likely to satisfy both $x \in S_x$ and $\mathrm{len}(x) = \mathrm{len}_\mathrm{s}(x) = r$. In this sense, the distribution of random super summit elements with given canonical length produced by the method used in our tests is very natural.

According to tests with other methods of generating random elements, the main results as formulated in Section 5.1 do not seem to depend crucially on the details of random element generation. Consider, for example, creating pseudo-random elements by choosing random sequences of Artin generators and their inverses. In this case, the number of elements with larger than average ultra summit sets increases compared to the results from Section 5.1 for small values of the canonical length r ($r \approx 10$). The asymptotic behaviour, however, remains unchanged: the size of the ultra summit set is almost always $2r$ for large values of r.

Remark 5.2. The structure of ultra summit sets in general is not well understood. One exception is the Artin braid group B_3 on three strings, for which ultra summit sets can be completely described. If $x \in B_3$ then $\mathbf{c}^{K \cdot \mathrm{len}_\mathrm{s}(x)}(y) = y$ for all $y \in S_x$, where $K = 1$ if $\inf_\mathrm{s}(x)$ is even and $K = 2$ if $\inf_\mathrm{s}(x)$ is odd. In particular, $U_x = S_x$. Moreover, U_x consists either of a single orbit under cycling or of a pair of orbits conjugate by δ. Hence $|S_x| = |U_x| \leqslant \max\{1, 2 \cdot \mathrm{len}_\mathrm{s}(x)\}$. It is also possible to derive regular expressions classifying the sequences of simple elements in the normal forms of ultra (or super) summit elements with even and odd infimum.

Little is known for other groups; even for the special case of Artin braid groups the understanding is limited. The behaviour seen in computational results as in Section 5.1 has been linked to the Nielsen–Thurston classification of braids viewed as isotopy classes of homeomorphisms of a disk with n punctures (where n is the braid index); see [16] for details. It seems likely that pseudo-Anosov braids[1] have small ultra summit sets whereas the ultra summit sets of periodic[2] and reducible[3] braids may be much larger. As a sufficiently long product of random simple elements is with high probability pseudo-Anosov [14,16], this would explain the results from Section 5.1. However, a complete understanding of the size and structure of ultra summit sets has not been achieved yet.

Following a similar approach for understanding ultra summit sets in the situation of general Garside groups would require replacing the geometric concepts provided by the

[1] A braid is pseudo-Anosov if it is represented by a homeomorphism which preserves two transverse measured foliations, while scaling their measures by factors λ and $1/\lambda$, respectively.

[2] A braid x is periodic if there are integers u and v such that $x^u = \delta^v$.

[3] A braid is reducible if there is an essential closed one-dimensional sub-manifold which it leaves invariant. A braid which is not reducible is either periodic or pseudo-Anosov.

Nielsen–Thurston classification and dependent results by algebraic alternatives. Whether this is possible is unclear at present.

6. Conclusions

We define in this paper a new invariant of conjugacy classes in Garside groups, the ultra summit set, using the digraph structure of the well-known super summit set induced by the cycling operation and establish that it satisfies "convexity" properties analogous to the ones holding for super summit sets. Ultra summit sets seem to be rather natural objects and promise to be useful for further theoretical analysis of Garside groups.

Apart from their theoretical significance, our results allow efficient computation of ultra summit sets, providing a practical solution to the conjugacy decision and search problems in Garside groups.

Our tests for Artin's presentation of B_n show that, in particular for larger braid index n, super summit elements are extremely common and super summit sets hence are much too large to be of computational use. Ultra summit elements, on the other hand, seem to be extremely rare and ultra summit sets can be computed easily even for large values of braid index and canonical length. We demonstrate that, using ultra summit sets, random instances of the conjugacy decision and search problems can be solved in very little time on current computers for elements of canonical length 1000 in B_{100}. This has, among others, implications for the security of certain braid-based cryptographic protocols. An attack on these protocols which employs conjugacy search using ultra summit sets is presented in [12]. It is shown there that the considered protocols are insecure for almost all random choices of keys.

Hence from both a theoretical and a computational point of view, the notion of ultra summit sets appears to be a significant advance in the study of the conjugacy problems in Garside groups.

Acknowledgments

I am grateful to Professor John Cannon and the University of Sydney for their support for this research project. I thank Professor Patrick Dehornoy and Professor Hugh Morton for valuable comments on an earlier version of this paper.

References

[1] I. Anshel, M. Anshel, D. Goldfeld, An algebraic method for public-key cryptography, Math. Res. Lett. 6 (3–4) (1999) 287–291.
[2] E. Artin, Theory of braids, Ann. of Math. (2) 48 (1947) 101–126.
[3] J. Birman, K.H. Ko, S.J. Lee, A new approach to the word and conjugacy problems in the braid groups, Adv. Math. 139 (2) (1998) 322–353.
[4] J.S. Birman, K.H. Ko, S.J. Lee, The infimum, supremum, and geodesic length of a braid conjugacy class, Adv. Math. 164 (1) (2001) 41–56.

[5] W. Bosma, J. Cannon, C. Playoust, The MAGMA algebra system I: The user language, J. Symbolic Comput. 24 (1997) 235–265, http://magma.maths.usyd.edu.au/magma/.
[6] P. Dehornoy, Groupes de Garside, Ann. Sci. École Norm. Sup. (4) 35 (2) (2002) 267–306.
[7] P. Dehornoy, L. Paris, Gaussian groups and Garside groups, two generalisations of Artin groups, Proc. London Math. Soc. (3) 79 (3) (1999) 569–604.
[8] E.A. El-Rifai, H.R. Morton, Algorithms for positive braids, Quart. J. Math. Oxford Ser. (2) 45 (1994) 479–497.
[9] D.B.A. Epstein, J.W. Cannon, D.F. Holt, S.V.F. Levy, M.S. Paterson, W.P. Thurston, Word Processing in Groups, Jones & Bartlett, Boston, MA, 1992, Chapter 9.
[10] N. Franco, J. González-Meneses, Conjugacy problem for braid groups and Garside groups, J. Algebra 266 (1) (2003) 112–132.
[11] F.A. Garside, The braid group and other groups, Quart. J. Math. Oxford Ser. (2) 20 (1969) 235–254.
[12] V. Gebhardt, Conjugacy search in braid groups, Appl. Algebra Engrg. Comm. Comput. (2005), in press.
[13] K.H. Ko, S.J. Lee, J.H. Cheon, J.W. Han, J.-S. Kang, C. Park, New public-key cryptosystem using braid groups, in: Advances in Cryptology, CRYPTO 2000, Santa Barbara, CA, in: Lecture Notes in Comput. Sci., vol. 1880, Springer-Verlag, Berlin, 2000, pp. 166–183.
[14] E. Lee, S.J. Lee, S.G. Hahn, Pseudorandomness from braid groups, in: Advances in Cryptology, CRYPTO 2001, Santa Barbara, CA, in: Lecture Notes in Comput. Sci., vol. 2139, Springer-Verlag, Berlin, 2001, pp. 486–502.
[15] M. Picantin, The conjugacy problem in small Gaussian groups, Comm. Algebra 29 (3) (2001) 1021–1039.
[16] W.P. Thurston, On the geometry and dynamics of diffeomorphisms of surfaces, Bull. Amer. Math. Soc. (N.S.) 19 (2) (1988) 417–431.