THE CONJUGACY PROBLEM - THEORY AND APPLICATIONS

Jens Harlander (BSU), Hannah Lewis (DSC), Jonathan Siegel (UCSC), and Chao Xu (SBU)

The Big Picture

At the center of a crypto system is a mathematical trapdoor, that is, a computational problem that is easy to do in one direction (encryption) but hard to reverse (decryption). A good example is integer multiplication: it is easy to multiply, but hard to factor. The security of classical crypto systems based on number factoring and related problems has various weaknesses. Mathematicians search for trapdoors that involve computations in non-commutative structures that provide more security in crypto systems. One such problem is the conjugacy problem in group theory.

The Conjugacy Problem

Let G be a group. The conjugacy problem in G states: given g_1, $g_2 \in G$, decide if they are conjugate, that is, if there is a group element g so that $g_2 = gg_1g^{-1}$. The conjugacy search problem states: Given two elements g_1 and g_2 that are conjugate in G, find the conjugator g. These problems depend on how the group G is given. Most commonly G is given by a presentation $\langle x_1, ...x_n \mid r_1, ..., r_m\rangle$, where the x_i generate the group and the r_j are relations that hold among the generators. For our project we studied the conjugacy problem in the braid group.

Conjugacy Problem Algorithms in the Braid Groups

In 1969 F. Garside found an algorithm for solving the conjugacy problem in braid groups. His algorithm is inefficient, it does not run in polynomial time. In 1988 W. P. Thurston proved that the braid groups are automatic. This insight provided another approach to solving the conjugacy problem, but his algorithm is equally complex. It is not known to this day if there is an efficient algorithm that does the job. This makes braid groups interesting from a cryptography viewpoint.

The braid group B_3 is special in many ways. Garside's and Thurston's algorithms remain inefficient even in case $n = 3$, but we found algorithms that solve the conjugacy problem in B_3 in linear time.

The Braid Group

The elements of the braid group B_n are n-stranded braids. We multiply braids by concatenation. The braid group has a presentation

$$\langle \sigma_1, ..., \sigma_{n-1} \mid \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1},\ \sigma_i\sigma_j = \sigma_j\sigma_i\rangle,$$

where $i = 1, ..., n-2$ and $j = 1, ..., n-1$, $|i-j| > 1$.

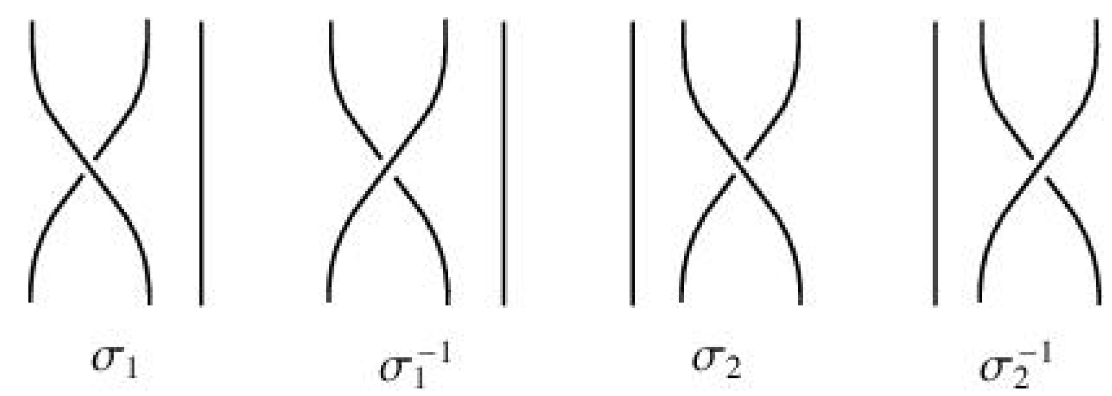

Figure 1: braid generators

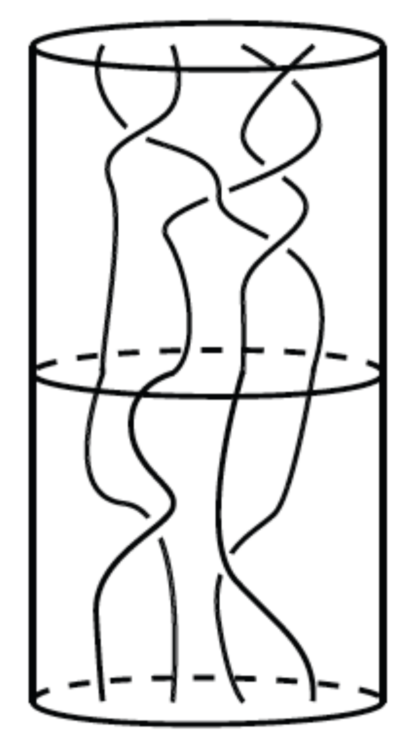

Figure 2: multiplication of two braids

Key Exchange Protocols Based on the Conjugacy Problem

A key exchange protocol is a procedure by which Alice and Bob can establish a common secret encryption key. The protocol works as follows: Let A and B be subgroups of some group G and $\mathscr{A} = \{a_1, ..., a_k\} \subseteq A$ and $\mathscr{B} = \{b_1, ..., b_l\} \subseteq B$ be subsets. This information is public, A and $\mathscr{A}$, B and $\mathscr{B}$ are posted on Alice' and Bob's homepage, respectively. Alice choses a word α in the a_i, $i = 1, ..., k$ and Bob choses a word β in the b_j, $j = 1, ..., l$. The elements $\alpha \in A$ and $\beta \in B$ are Alice' and Bob's private keys. Alice sends Bob the set of conjugates $\{\alpha b_1\alpha^{-1}, ..., \alpha b_l\alpha^{-1}\}$ and Bob send Alice the set of conjugates $\{\beta a_1\beta^{-1}, ..., \beta a_k\beta^{-1}\}$. Alice can then compute $\beta\alpha\beta^{-1}$ by replacing every a_i in the word α by $\beta a_i\beta^{-1}$. Similarly, Bob can compute $\alpha\beta\alpha^{-1}$. Alice now computes the secret encryption key $K = (\beta\alpha\beta^{-1})\alpha^{-1}$ and Bob computes the secret encryption key $K' = ((\alpha\beta\alpha^{-1})\beta^{-1})^{-1}$. Note that $K = K'$. An eavesdropper only sees the a_i and the conjugates $\beta a_i\beta^{-1}$, but not the conjugator β. He only sees the b_j and the conjugates $\alpha b_j\alpha^{-1}$, but not the conjugator α. Thus, the eavesdropper will have to solve the conjugacy search problem in A and B in order to construct the secret encryption key K.

Algorithms and Their Complexity

Algorithm	B_3		B_n	
	Time	Space	Time	Space
Word problem with Artin combing	$O(3^l)$	$O(3^l)$	$O(n3^l)$	$O(n3^l)$
Word problem with unique Graside normal form	$O(l)$	$O(l)$	-	-
Word problem with Thurston left normal form	$O(l^2)$	$O(l)$	$O(l^2n^2)$	$O(ln^2)$
Word problem with $a^2 = b^3$ presentation	$O(l)$	$O(l)$	-	-
Conjugacy problem with $a^2 = b^3$ presentation	$O(l)$	$O(l)$	-	-
Conjugacy problem with automatic groups	$O(2^{12l})$	$O(2^{12l})$	$O((2n-2)^{2ln(n-1)})$	$O((2n-2)^{2ln(n-1)})$
Conjugacy problem with ultra summit sets	$O(kl^2)$	$O(kl^2)$	$O(kl^2n^4)$	$O(kl^2n^4)$

Summary of Results

The standard algorithms by Garside and Thurston do not provide efficient solutions for solving the conjugacy problem in braid groups, not even in the three strand braid group B_3. However, we have found other algorithms that provide linear time solutions for both the word and conjugacy problems in B_3. These algorithms rely on special combinatorial and topological features of B_3.The table below summarizes our findings so far. The table also lists findings concerning the word problem, which is a special case of the conjugacy problem. The word problem asks: given a word w expressed in the generators of a group G, does it presents the trivial element in G?

Questions for Future Work

Do any of the special properties of B_3 carry over to B_n, $n > 3$? Can geometric ideas be applied to study B_n? For which n is B_n non-positively curved? B_4 and B_5 are known to be non-positively curved. Can this be used for the design of efficient algorithms for the conjugacy problem?

References

[1] A. J. Berrick, F. R. Cohen, E. Hanbury, Yan-Loi Wong, Jie Wu (editors), Braids: Introductory Lectures on Braids, Configurations and Their Applications, Lecture Note Series, Institute for Mathematical Sciences, National University of Sinapore, Vol 19, World Scinetific Press, 2009.

[2] J. G. Boiser, *Computational Problems in the Braid Group*, Masters Thesis, San Diego State University, 2009;

[3] F. A. Garside, *The braid group and other groups*, Quart. J. Math. Oxford Ser. (2), 20 1969, 235-254;

[4] C. Kassel, V. Turaev, Braid Groups, Graduate Text in Mathematics, Springer, 2010;

[5] William P. Thurston, *On the geometry and dynamics of diffeomorphisms of surfaces*, Bull. Amer. Math. Soc. (N.S.), 19(2) 1988, 417-431.

Acknowledgements

This work was conducted as part of an REU program on complexity in algebra, geometry and applications at Boise State University in the Summer of 2011. We gratefully acknowledge funding from the National Science Foundation (DMS 1062857) and Boise State University.

NSF

BOISE STATE UNIVERSITY

DSC DIXIE STATE COLLEGE OF UTAH

UNIVERSITY OF CALIFORNIA SANTA CRUZ

STONY BROOK STATE UNIVERSITY OF NEW YORK