J. Cryptol. (2015) 28: 601–622
DOI: 10.1007/s00145-013-9170-9

Journal of CRYPTOLOGY

Polynomial-Time Solutions of Computational Problems in Noncommutative-Algebraic Cryptography

Boaz Tsaban

Department of Mathematics, Bar-Ilan University, Ramat Gan 52900, Israel
tsaban@math.biu.ac.il
url: http://www.cs.biu.ac.il/~tsaban

Communicated by Nigel Smart.

Received 28 January 2013
Online publication 15 November 2013

Abstract. We introduce the *linear centralizer method*, and use it to devise a provable polynomial-time solution of the Commutator Key Exchange Problem, the computational problem on which, in the passive adversary model, the security of the Anshel–Anshel–Goldfeld (Anshel et al., Math. Res. Lett. 6:287–291, 1999) *Commutator* key exchange protocol is based. We also apply this method to solve, in polynomial time, the computational problem underlying the *Centralizer* key exchange protocol, introduced by Shpilrain and Ushakov in (Contemp. Math. 418:161–167, 2006).

This is the first provable polynomial-time cryptanalysis of the Commutator key exchange protocol, hitherto the most important key exchange protocol in the realm of noncommutative algebraic cryptography, and the first cryptanalysis (of any kind) of the Centralizer key exchange protocol. Unlike earlier cryptanalyses of the Commutator key exchange protocol, our cryptanalyses cannot be foiled by changing the distributions used in the protocol.

Key words. Noncommutative-algebraic cryptography, Group theory-based cryptography, Braid-based cryptography, Commutator key exchange, Centralizer key exchange, Braid Diffie–Hellman key exchange, Linear cryptanalysis, Invertibility lemma, Schwartz–Zippel lemma, Linear centralizer method, Braid infinimum reduction, Algebraic cryptanalysis.

1. Introduction

Since Diffie and Hellman's 1976 key exchange protocol, few alternative proposals for key exchange protocols (KEPs) resisted cryptanalysis. This, together with the (presently, theoretical) issue that the Diffie–Hellman and other classic KEPs can be broken in polynomial time by quantum computers, is a strong motivation for searching for substantially different KEPs. Lattice-based KEPs [31] seem to be a viable potential alternative. All classic KEPs as well as the Lattice-based ones are based on commutative algebraic structures.

In 1999, Anshel, Anshel, and Goldfeld [2] (cf. [3]) introduced the *Commutator KEP*, a general method for constructing KEPs based on *noncommutative* algebraic structures. Around the same time, Ko, Lee, Cheon, Han, Kang, and Park [21] introduced the *Braid Diffie–Hellman KEP*, another general method achieving the same goal. The security of both KEPs is based on variations of the *Conjugacy Search Problem*: Given conjugate elements g, h in a noncommutative group, find x in that group such that $x^{-1}gx = h$. Both papers [2] and [21] proposed to use the *braid group* $\mathbf{B}_N$, a finitely presented, infinite noncommutative group parameterized by a natural number N, as the platform group.

The introduction of the Commutator KEP and the Braid Diffie–Hellman KEP was followed by a stream of heuristic attacks (e.g., [11–13,15–18,24–27,29,30]),[1] demonstrating that these protocols, *when using the two most simple distributions* on the braid group $\mathbf{B}_N$, are insecure. Consequently, a program was set forth, by several independent research groups, to find efficiently samplable distributions on the braid group that, when used with the above-mentioned protocols, foil all heuristic attacks (e.g., [1,14,22,25]). The abstract of [14] concludes: "Proper choice ... produces a key exchange scheme which is resistant to all known attacks". Moreover, a very practical distribution is announced in [37], which foils the strongest known methods for solving the Conjugacy Search Problem in $\mathbf{B}_N$.

Most of the mentioned heuristic attacks address the Commutator KEP, and not the Braid Diffie–Hellman KEP. The reason is that in 2003, Cheon and Jun published an expected polynomial-time cryptanalysis of the Braid Diffie–Hellman KEP, using a novel representation theoretic method [8]. In their paper, Cheon and Jun stress that their cryptanalysis *does not apply to the Commutator KEP* and that an extra ingredient is needed. Thus far, no expected polynomial-time attack was found on the Commutator KEP, whose success does not depend on the distributions used in the protocols.

The main result of the present paper is a Las Vegas, provable expected polynomial-time solution of the Commutator Key Exchange Problem (also referred to as the *Anshel–Anshel–Goldfeld Problem* [28, §15.1.2]), the computational problem underlying the Commutator KEP. This forms a cryptanalysis of the Commutator KEP [2], in the passive adversary model, which succeeds regardless of the distributions used to generate the keys.

The *linear centralizer* method, developed for our solution of the Commutator Key Exchange Problem, is applicable to additional computational problems and KEPs in the context of group theory-based cryptography. We present an application of these methods to the *Centralizer KEP*, introduced by Shpilrain and Ushakov in 2006 [35], to obtain an expected polynomial-time attack. This is the first cryptanalysis, of any kind, of the Centralizer KEP.

We stress that the cryptanalyses presented here, like the Cheon–Jun cryptanalysis, while of expected polynomial time, are impractical for standard values of N (e.g., $N = 100$). These results are of theoretic nature. Ignoring logarithmic factors, the complexity of our cryptanalyses is about N^{17}, times a cubic polynomial in the other relevant parameters. Incidentally, though, these cryptanalyses establish the first provable practical attacks in the case where the index N of the braid group $\mathbf{B}_N$ is small, e.g., when $N = 8$.

[1] Surveys of some of the heuristic attacks are provided in Dehornoy [9] and Garber [10].

The paper is organized as follows. Section 2 introduces the Commutator KEP and the braid group. In Sect. 3, we eliminate a technical complexity theoretic obstacle. Section 4 applies a method of Cheon and Jun to reduce our problem to matrix groups over finite fields. Section 5 is the main ingredient of our cryptanalysis, presenting the new method and cryptanalyzing the Commutator KEP in matrix groups. This section is independent of the other sections and readers without prior knowledge of the braid group may wish to read it first. Section 6 fills a gap in our proof, by applying the Schwartz–Zippel Lemma to obtain a lower bound on the probability that certain random matrices are invertible. Section 7 is a cryptanalysis of the Centralizer KEP, using the methods introduced in the earlier sections. The Braid Diffie–Hellman KEP is introduced in Sect. 8, where we survey the Cheon–Jun cryptanalysis and explain why it does not apply to the Commutator KEP or to the Centralizer KEP. We also describe applications of the new methods to a generalized version of the Braid Diffie–Hellman KEP and to Stickel's KEP. Some additional discussion is provided in Sect. 9.

2. The Commutator KEP and the Braid Group $\mathbf{B}_N$

We will use, throughout, the following basic notation.

Notation 1. For a noncommutative group G and group elements $g, x \in G$, $g^x = x^{-1}gx$, the conjugate of g by x.

Useful identities involving this notation that are easy to verify, include $g^{xy} = (g^x)^y$, and $g^c = g$ for every *central* element $c \in G$, that is, such that $ch = hc$ for all $h \in G$.

The *Commutator KEP* [2] is described succinctly in Fig. 1.[2] In some detail:

1. A noncommutative group G and elements $a_1, \ldots, a_k, b_1, \ldots, b_k \in G$ are publicly given.[3]
2. Alice and Bob choose free group words in the variables $x_1, \ldots, x_k$, $v(x_1, \ldots, x_k)$ and $w(x_1, \ldots, x_k)$, respectively.[4]
3. Alice substitutes $a_1, \ldots, a_k$ for $x_1, \ldots, x_k$, to obtain a secret element $a = v(a_1, \ldots, a_k) \in G$. Similarly, Bob computes $b = w(b_1, \ldots, b_k) \in G$.
4. Alice sends the conjugated elements ${b_1}^a, \ldots, {b_k}^a$ to Bob, and Bob sends ${a_1}^b, \ldots, {a_k}^b$ to Alice.
5. The shared key is the *commutator* $a^{-1}b^{-1}ab$.

As conjugation is a group isomorphism, we have

$$v\left({a_1}^b, \ldots, {a_k}^b\right) = v(a_1, \ldots, a_k)^b = a^b = b^{-1}ab.$$

Thus, Alice can compute the shared key $a^{-1}b^{-1}ab$ as $a^{-1}v({a_1}^b, \ldots, {a_k}^b)$, using her secret a, $v(x_1, \ldots, x_k)$ and the public elements ${a_1}^b, \ldots, {a_k}^b$. Similarly, Bob computes $a^{-1}b^{-1}ab$ as $w({b_1}^a, \ldots, {b_k}^a)^{-1}b$.

[2] In our diagrams, green letters indicate publicly known elements, and red ones indicate secret elements, known only to the secret holders. Results of computations involving elements of both colors may be either publicly known, or secret, depending on the context. The colors are not necessary to follow the diagrams.

[3] By adding elements, if needed, we assume that the number of a_i's is equal to the number of b_i's.

[4] A free group word in the variables $x_1, \ldots, x_k$ is a product of the form $x_{i_1}^{\epsilon_1} x_{i_2}^{\epsilon_2} \cdots x_{i_m}^{\epsilon_m}$, with $i_1, \ldots, i_m \in \{1, \ldots, k\}$ and $\epsilon_1, \ldots, \epsilon_m \in \{1, -1\}$, and with no subproduct of the form $x_i x_i^{-1}$ or $x_i^{-1} x_i$.

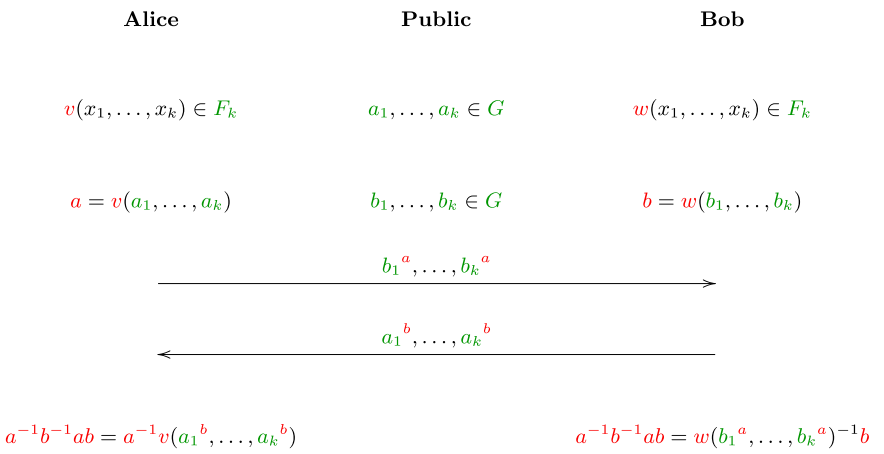

Fig. 1. The commutator KEP.

For the platform group G, it is proposed in [2] to use the *braid group* $\mathbf{B}_N$, a group parameterized by a natural number N. Elements of $\mathbf{B}_N$ are called *braids*, for they may be identified with braids on N strands. Braid group multiplication is motivated geometrically, but the details will play no role in the present paper. The interested reader will find detailed information on $\mathbf{B}_N$ in almost each of the papers in the bibliography and, in particular, in the survey [6], but prior knowledge is not necessary: we quote here the information needed for the present paper.

Let S_N be the symmetric group of permutations on N symbols. For our purposes, the braid group $\mathbf{B}_N$ is a group of elements of the form

$$(i,\mathbf{p}),$$

where i is an integer, and $\mathbf{p}$ is a finite (possibly, empty) sequence of elements of S_N, that is, $\mathbf{p}=(p_1,\dots,p_\ell)$ for some $\ell\geq 0$ and $p_1,\dots,p_\ell\in S_N$. The sequence $\mathbf{p}=(p_1,\dots,p_\ell)$ is requested to be *left weighted* (a property whose definition will not be used here), and p_1 must not be the involution $p(k)=N-k+1$.[5]

For "generic" braids $(i,(p_1,\dots,p_\ell))\in\mathbf{B}_N$, i is negative and $|i|$ is $O(\ell)$, but this is not always the case. Note that the bit-length of an element $(i,(p_1,\dots,p_\ell))\in\mathbf{B}_N$ is $O(\log|i|+\ell N\log N)$.

Multiplication is defined on $\mathbf{B}_N$ by an algorithm of complexity $O(\ell^2 N\log N+\log|i|)$. Inversion is of linear complexity. Explicit implementations are provided, for example, in [7].

For a passive adversary to extract the shared key of the Commutator KEP out of the public information, it suffices to solve the following problem, also referred to as the *Anshel–Anshel–Goldfeld Problem* [28, §15.1.2].

[5] For readers familiar with the braid group, we point out that the sequence $(i,(p_1,\dots,p_\ell))$ encodes the left normal form $\Delta^i p_1\cdots p_\ell$ of the braid, in Artin's presentation, with Δ being the fundamental, half twist braid on N strands.

Problem 2 (Commutator KEP Problem). *Let* $a_1, \dots, a_k, b_1, \dots, b_k \in \mathbf{B}_N$, *each of the form* $(i, \mathbf{p})$ *with* $\mathbf{p}$ *of length* $\le \ell$. *Let* a *be a product of at most* m *elements of* $\{a_1, \dots, a_k\}^{\pm 1}$, *and let* b *be a product of at most* m *elements of* $\{b_1, \dots, b_k\}^{\pm 1}$.

Given $a_1, \dots, a_k, b_1, \dots, b_k, a_1^b, \dots, a_k^b, b_1^a, \dots, b_k^a$, *compute* $a^{-1}b^{-1}ab$.

Our solution of Problem 2 consists of several ingredients.

3. Reducing the Infimum

The *infimum* of a braid $b = (i, \mathbf{p})$ is the integer $\inf(b) := i$. As the bit-length of b is $O(\log|i| + \ell N \log N)$, an algorithm polynomial in $|i|$ would be at least *exponential* in the bit-length. We first remove this obstacle.

In cases where $\mathbf{p}$ is the empty sequence, we write (i) instead of $(i, \mathbf{p})$. The properties of $\mathbf{B}_N$ include, among others, the following ones.

(a) $(i) \cdot (j, \mathbf{p}) = (i + j, \mathbf{p})$ for all integers i and all $(j, \mathbf{p}) \in \mathbf{B}_N$.
In particular, $(i) = (1)^i$ for all i.
(b) $(2) \cdot (i, \mathbf{p}) = (i, \mathbf{p}) \cdot (2)$ for all for all $(i, \mathbf{p}) \in \mathbf{B}_N$.

Thus, $(2j)$ is a central element of $\mathbf{B}_N$ for each integer j. If follows that, for each $(i, \mathbf{p}) \in \mathbf{B}_N$,

$$(i, \mathbf{p}) = \big(i - (i \bmod 2)\big) \cdot (i \bmod 2, \mathbf{p}).$$

This way, every braid $b \in \mathbf{B}_N$ decomposes to a unique product $c\tilde{b}$, where c is of the form $(2j)$ (and thus *central*), and $\inf(\tilde{b}) \in \{0, 1\}$.

Consider the public information in Fig. 1. For each $j = 1, \dots, k$, decompose as above

$$a_j = c_j \tilde{a}_j,$$
$$b_j = d_j \tilde{b}_j,$$

with c_j, d_j central and $\inf(\tilde{a}_j), \inf(\tilde{b}_j) \in \{0, 1\}$ for all $j = 1, \dots, k$. Let

$$\tilde{a} = v(\tilde{a}_1, \dots, \tilde{a}_k);$$
$$\tilde{b} = w(\tilde{b}_1, \dots, \tilde{b}_k);$$
$$c = v(c_1, \dots, c_k);$$
$$d = w(d_1, \dots, \tilde{d}_k).$$

As the elements c_j, d_j are central, we have

$$\tilde{a} = v\big(c_1^{-1}a_1, \dots, c_k^{-1}a_k\big) = v\big(c_1^{-1}, \dots, c_k^{-1}\big) \cdot v(a_1, \dots, a_k) = c^{-1}a.$$

Similarly, $\tilde{b} = d^{-1}b$. As c and d are central,

$$a_j^b = (c_j\tilde{a}_j)^b = c_j\tilde{a}_j^b = c_j\tilde{a}_j^{d\tilde{b}} = c_j\tilde{a}_j^{\tilde{b}}$$

for all $j = 1, \dots, k$. Thus, $\tilde{a}_j^{\tilde{b}}$ can be computed for all j. Similarly, $\tilde{b}_j^{\tilde{a}}$ can be computed. Now,

$$a^{-1}b^{-1}ab = (c\tilde{a})^{-1}(d\tilde{b})^{-1}(c\tilde{a})(d\tilde{b}) = \tilde{a}^{-1}c^{-1}\tilde{b}^{-1}d^{-1}c\tilde{a}d\tilde{b} = \tilde{a}^{-1}\tilde{b}^{-1}\tilde{a}\tilde{b}.$$

This shows that the Commutator KEP Problem is reducible, in linear time, to the same problem using $\tilde{a}_1, \ldots, \tilde{a}_k, \tilde{b}_1, \ldots, \tilde{b}_k$ instead of $a_1, \ldots, a_k, b_1, \ldots, b_k$. Thus, we may assume that

$$\inf(a_1), \ldots, \inf(a_k), \inf(b_1), \ldots, \inf(b_k) \in \{0, 1\}$$

to start with. Assume that henceforth.

For a braid $x = (i, \mathbf{p})$, let $\ell(\mathbf{p})$ be the number of permutations in the sequence $\mathbf{p}$. For integers i, s, let

$$[i, s] = \big\{x \in \mathbf{B}_N : i \leq \inf(x) \leq \inf(x) + \ell(x) \leq s\big\}.$$

We use the following basic facts about $\mathbf{B}_N$:

1. If $x_1 \in [i_1, s_1]$ and $x_2 \in [i_2, s_2]$, then $x_1 x_2 \in [i_1 + i_2, s_1 + s_2]$.
2. If $x \in [i, s]$, then $x^{-1} \in [-s, -i]$.

Thus, for each $x \in \{a_1, \ldots, a_k, b_1, \ldots, b_k\}^{\pm 1}$, $x^{\pm 1} \in [-\ell - 1, \ell + 1]$, and therefore, in the notation of our problem, $a, b \in [-m(\ell + 1), m(\ell + 1)]$. Thus,

$$a^{-1}b^{-1}ab \in \big[-4m(\ell + 1), 4m(\ell + 1)\big].$$

Corollary 3. *In the Commutator KEP Problem,* $a^{-1}b^{-1}ab \in [-4m(\ell+1), 4m(\ell+1)]$.

4. Reducing to a Matrix Group over a Finite Field

In this section, we apply methods of Cheon and Jun [8] in our setting.

Let n be a natural number. As usual, we denote the algebra of all $n \times n$ matrices over a field $\mathbb{F}$ by $\mathrm{M}_n(\mathbb{F})$, and the group of invertible elements of this algebra by $\mathrm{GL}_n(\mathbb{F})$. A *matrix group* is a subgroup of $\mathrm{GL}_n(\mathbb{F})$. A *faithful representation* of a group G in $\mathrm{GL}_n(\mathbb{F})$ is a group isomorphism from G onto a matrix group $H \leq \mathrm{GL}_n(\mathbb{F})$. A group is *linear* if it has a faithful representation.

Bigelow and, independently, Krammer, established in their breakthrough papers [5,23] that the braid group $\mathbf{B}_N$ is linear, by proving that the so-called *Lawrence–Krammer representation*

$$\mathrm{LK}\colon \mathbf{B}_N \longrightarrow \mathrm{GL}_{\binom{N}{2}}\left(\mathbb{Z}\left[t^{\pm 1}, \frac{1}{2}\right]\right),$$

whose dimension is

$$n := \binom{N}{2},$$

is injective.[6] The Lawrence–Krammer representation of a braid can be computed in polynomial time.[7] It is proved implicitly in [23], and explicitly in [8], that this representation is also invertible in (similar) polynomial time. The following result follows from Corollary 1 of [8].

[6] Bigelow proved this theorem for the coefficient ring $\mathbb{Z}[t^{\pm 1}, q^{\pm 1}]$ with two variables. Krammer proved, in addition, that one may replace q by any real number from the interval $(0, 1)$.

[7] When the infimum i is polynomial in the other parameters, which we proved in Sect. 3 that we may assume. Alternatively, by computing the representation of (i) separately, using properties of the Lawrence–Krammer representation.

Theorem 4 (Cheon–Jun [8]). *Let $x \in [i, s]$ in $\mathbf{B}_N$. Let $M \geq \max(|i|, |s|)$. Then*:

1. *The degrees of t in $\mathrm{LK}(x) \in \mathrm{GL}_n(\mathbb{Z}[t^{\pm 1}, \frac{1}{2}])$ are in $\{-M, -M+1, \ldots, M\}$.*
2. *The rational coefficients $\frac{c}{2^d}$ in $\mathrm{LK}(x)$ (c integer, d nonnegative integer) satisfy $|c| \leq 2^{N^2 M}$, $|d| \leq 2NM$.*

In the notation of Theorem 4, Theorem 2 in Cheon–Jun [8] implies that inversion of $\mathrm{LK}(x)$ is of order $N^6 \log M$ multiplications of entries. Ignoring logarithmic factors and thus assuming that each entry multiplication costs $NM \cdot N^2 M = N^3 M^2$, this accumulates to $N^8 M^2$. We will invert the function LK as part of our cryptanalysis below. However, the complexity of the other steps in our cryptanalysis (in particular, the linear centralizer step—Sect. 5) dominate the complexity of inverting LK.

Let us return to the Commutator KEP Problem 2. By Corollary 3,

$$K := a^{-1} b^{-1} a b \in \big[-4m(\ell+1), 4m(\ell+1)\big].$$

Let $M = 4m(\ell+1)$. By Theorem 4, we have

$$\big(2^{2NM} t^M\big) \cdot \mathrm{LK}(K) \in \mathrm{GL}_n\big(\mathbb{Z}[t]\big),$$

the absolute values of the coefficients in this matrix are bounded by $2^{N^2(M+1)}$, and the maximal degree of t in this matrix is bounded by $2M$.

Let p be a prime slightly greater than $2^{N^2 M}$, and $f(t)$ be an irreducible polynomial over $\mathbb{Z}_p$, of degree d slightly larger than $2M$. Then

$$\big(2^{2NM} t^M\big) \cdot \mathrm{LK}(K) = \big(2^{2NM} t^M\big) \cdot \mathrm{LK}(K) \bmod \big(p, f(t)\big) \in \mathrm{GL}_n\big(\mathbb{Z}[t]/\langle p, f(t)\rangle\big),$$

under the natural identification of $\{-(p-1)/2, \ldots, (p-1)/2\}$ with $\{0, \ldots, p-1\}$.

Let $\mathbb{F} = \mathbb{Z}[t]/\langle p, f(t)\rangle = \mathbb{Z}[t^{\pm 1}, \frac{1}{2}]/\langle p, f(t)\rangle$. $\mathbb{F}$ is a finite field of cardinality p^d, where d is the degree of $f(t)$. It follows that the complexity of field operations in $\mathbb{F}$ is, up to logarithmic factors, of order

$$d^2 \log p = O\big(M^3 N^2\big) = O\big(m^3 \ell^3 N^2\big).$$

Thus, the key K can be recovered as follows:

1. Apply the composed function $\mathrm{LK}(x) \bmod (p, f(t))$ to the input of the Commutator KEP Problem, to obtain a version of this problem in $\mathrm{GL}_n(\mathbb{F})$.
2. Solve the problem there, to obtain $\mathrm{LK}(K) \bmod (p, f(t))$.
3. Compute $(2^{2NM} t^M) \cdot \mathrm{LK}(K) \bmod (p, f(t)) = (2^{2NM} t^M) \cdot \mathrm{LK}(K)$.[8]
4. Divide by $(2^{2NM} t^M)$ to obtain $\mathrm{LK}(K)$.
5. Compute K using the Cheon–Jun inversion algorithm.

It remains to devise a polynomial-time solution of the Commutator KEP Problem in arbitrary groups of matrices.

[8] The equality here is over the integers.

5. Linear Centralizers

In this section, we solve the Commutator KEP Problem in matrix groups. We first state the problem in a general form. As usual, for a group G and elements $g_1, \ldots, g_k \in G$, $\langle g_1, \ldots, g_k \rangle$ denotes the subgroup of G generated by $g_1, \ldots, g_k$. Throughout, we assume that the given groups are represented in an efficient way.

Problem 5 (Commutator KEP Problem). *Let G be a group. Let $a_1, \ldots, a_k, b_1, \ldots, b_k \in G$. Let $a \in \langle a_1, \ldots, a_k \rangle$, $b \in \langle b_1, \ldots, b_k \rangle$.*
Given $a_1, \ldots, a_k, b_1, \ldots, b_k, a_1^b, \ldots, a_k^b, b_1^a, \ldots, b_k^a$, compute $a^{-1}b^{-1}ab$.

We recall a classic definition.

Definition 6. Let $S \subseteq \mathrm{M}_n(\mathbb{F})$ be a set. The *centralizer* of S (in $\mathrm{M}_n(\mathbb{F})$) is the set

$$C(S) = \{c \in \mathrm{M}_n(\mathbb{F}) : cs = sc \text{ for all } s \in S\}.$$

For $a_1, \ldots, a_k \in \mathrm{M}_n(\mathbb{F})$, $C(\{a_1, \ldots, a_k\})$ is also denoted as $C(a_1, \ldots, a_k)$.

Basic properties of $C(S)$ that are easy to verify, include:

1. $C(S)$ is a vector subspace (indeed, a matrix subalgebra) of $\mathrm{M}_n(\mathbb{F})$.
2. $C(C(S)) \supseteq S$.
3. $C(S) = C(\operatorname{span} S)$.
4. If $S \subseteq \mathrm{GL}_n(\mathbb{F})$, then $C(S) = C(\langle S \rangle)$, where $\langle S \rangle$ is the subgroup of $\mathrm{GL}_n(\mathbb{F})$ generated by S.

A key observation is the following one: Let V be a vector subspace of $\mathrm{M}_n(\mathbb{F})$, and $G \leq \mathrm{GL}_n(\mathbb{F})$ be a matrix group such that $V \cap G$ is nonempty. It may be computationally infeasible to find an element in $V \cap G$. However, it is easy to compute a basis for $V \cap U$ for any vector subspace U of $\mathrm{M}_n(\mathbb{F})$. In particular, this is true for $U = C(C(G))$, which contains G. In certain cases, as the ones below, a "random" element in $V \cap C(C(G))$ is as good as one in $V \cap G$.

Algorithm 7 below addresses the Commutator KEP Problem in a matrix group $G \leq \mathrm{GL}_n(\mathbb{F})$. The analysis of this algorithm is based on the forthcoming Lemma 9, which shows that one can efficiently find an invertible matrix in a vector space of matrices containing at least one invertible matrix. To this end, we assume that $|\mathbb{F}|/n \geq c > 1$ for some constant c. In the above section, $|\mathbb{F}|/n$ is at least exponential. Fix a finite set $S \subseteq \mathbb{F}$ of cardinality greater than cn (the larger the better) that can be sampled efficiently. In the most important case, where $\mathbb{F}$ is a finite field, take $S = \mathbb{F}$. By *random element* of a vector subspace V of $\mathrm{M}_n(\mathbb{F})$, with a prescribed basis $\{v_1, \ldots, v_d\}$, we mean a linear combination

$$\alpha_1 v_1 + \cdots + \alpha_k v_k$$

with $\alpha_1, \ldots, \alpha_k \in S$ uniform, independently distributed.

It is natural to split the Commutator KEP Problem and the algorithm for solving it into an offline (preprocessing) phase and an online phase.

Algorithm 7.
Offline phase:

1. *Input:* $b_1, \ldots, b_k \in G$.
2. *Execution:*
   (a) Compute a basis $S = \{s_1, \ldots, s_d\}$ for $C(b_1, \ldots, b_k)$, by solving the following homogeneous system of linear equations in the n^2 entries of the unknown matrix x:
   $$\begin{aligned} b_1 \cdot x &= x \cdot b_1, \\ &\vdots \\ b_k \cdot x &= x \cdot b_k. \end{aligned}$$
   (b) Compute a basis for $C(S) = C(C(b_1, \ldots, b_k))$, by solving the following homogeneous system of linear equations in the n^2 entries of the unknown matrix x:
   $$\begin{aligned} s_1 \cdot x &= x \cdot s_1, \\ &\vdots \\ s_d \cdot x &= x \cdot s_d. \end{aligned}$$
3. *Output:* A basis for $C(C(b_1, \ldots, b_k))$.

Online phase:

1. *Input:* $a_1, \ldots, a_k, b_1, \ldots, b_k, a_1^b, \ldots, a_k^b, b_1^a, \ldots, b_k^a \in G$, where $a \in \langle a_1, \ldots, a_k \rangle$, $b \in \langle b_1, \ldots, b_k \rangle$ are unknown.
2. *Execution:*
   (a) Solve the following homogeneous system of linear equations in the n^2 entries of the unknown matrix x:
   $$\begin{aligned} b_1 \cdot x &= x \cdot {b_1}^a, \\ &\vdots \\ b_k \cdot x &= x \cdot {b_k}^a. \end{aligned}$$
   (b) Fix a basis for the solution space, and pick random solutions x until x is invertible.
   (c) Solve the following homogeneous system of linear equations in the n^2 entries of the unknown matrix y:
   $$\begin{aligned} a_1 \cdot y &= y \cdot {a_1}^b, \\ &\vdots \\ a_k \cdot y &= y \cdot {a_k}^b, \end{aligned}$$
   subject to the *linear constraint* that $y \in C(C(b_1, \ldots, b_k))$.
   (d) Fix a basis for the solution space, and pick random solutions y until y is invertible.
   (e) *Output:* $x^{-1} y^{-1} x y$.

Let ω be the matrix multiplication constant, that is, the minimal such that matrix multiplication is $O(n^{\omega+o(1)})$. For our applications, one may take $\omega = \log_2 7 \approx 2.81$. As

usual, *Las Vegas algorithm* means an algorithm that always outputs the correct answer in finite time. For the proof of the following theorem, note that if $g^x = g^y$, then $g^{xy^{-1}} = g$, or in other words, $xy^{-1} \in C(g)$. Finally, note that it does not make much sense to consider the case where $k > n^2$, in which the matrices become linearly dependent and thus redundant.

Theorem 8. *Assume that* $|\mathbb{F}|/n \geq c > 1$ *for some constant* c, *and* $k \leq n^2$. *Algorithm* 7 *is a Las Vegas algorithm for the Commutator KEP Problem*, *with running time*, *in units of field operations*:

1. Offline phase: $O(n^{2\omega+2})$.
2. Online phase: $O(kn^{2\omega})$.

Proof. We use the notation of Algorithm 7. First, assume that the algorithm terminates. We prove that its output is $a^{-1}b^{-1}ab$.

$$x^{-1}y^{-1}xy = x^{-1}y^{-1}\big(xa^{-1}\big)ay.$$

The equations 2(a) in the online phase of Algorithm 7 assert that ${b_i}^x = {b_i}^a$ for all $i = 1, \ldots, k$. Thus, $xa^{-1} \in C(b_1, \ldots, b_k)$. As $y \in C(C(b_1, \ldots, b_k))$, y commutes with xa^{-1}, and therefore so does y^{-1}. Thus,

$$x^{-1}y^{-1}\big(xa^{-1}\big)ay = x^{-1}\big(xa^{-1}\big)y^{-1}ay = a^{-1}y^{-1}ay = a^{-1}a^y.$$

By the equations 2(c) in the online phase of Algorithm 7, ${a_i}^y = {a_i}^b$ for all $i = 1, \ldots, k$. As $a \in \langle a_1, \ldots, a_k \rangle$, we have $a^y = a^b$. Indeed, let $a = a_{i_1}^{\epsilon_1} \cdots a_{i_m}^{\epsilon_m}$. As conjugation is an isomorphism,

$$\begin{aligned} a^y &= \big(a_{i_1}^{\epsilon_1}\big)^y \cdots \big(a_{i_m}^{\epsilon_m}\big)^y = \big(a_{i_1}^y\big)^{\epsilon_1} \cdots \big(a_{i_m}^y\big)^{\epsilon_m} = \big(a_{i_1}^b\big)^{\epsilon_1} \cdots \big(a_{i_m}^b\big)^{\epsilon_m} \\ &= \big(a_{i_1}^{\epsilon_1}\big)^b \cdots \big(a_{i_m}^{\epsilon_m}\big)^b = a^b. \end{aligned}$$

Thus,

$$a^{-1}a^y = a^{-1}a^b = a^{-1}b^{-1}ab.$$

Thus, the algorithm returns the correct answer when it terminates. It remains to analyze the running time of the algorithm, which we do step-by-step.

Offline phase, Step 2(a): These are kn^2 equations in n^2 variables, and thus the running time is $O(k(n^2)^\omega) = O(kn^{2\omega})$.

Offline phase, Step 2(b): As $C(b_1, \ldots, b_k)$ is a vector subspace of $\mathrm{M}_n(\mathbb{F})$, its dimension d is at most n^2. Thus, the running time of this step is $O(n^2 \cdot n^{2\omega}) = O(n^{2\omega+2})$.

Online phase, Step 2(a): The running time is $O(kn^{2\omega})$, as in Step 2(a) of the offline phase.

Online phase, Step 2(b): There is an invertible solution to the equations 2(a), namely: a. Thus, by the Invertibility Lemma (Lemma 9 below), the probability that a random solution is *not* invertible may be assumed arbitrarily close to $n/|\mathbb{F}| \leq 1/c < 1$. Thus, the expected number of random elements picked until an invertible one is found is constant. To generate one random element, one takes a linear combination of a basis of the solution space. If m is the dimension, then $m \leq n^2$ and the linear combination

takes $mn^2 \leq n^4$ operations. Checking invertibility is faster. The total expected running time of this step is, therefore, $O(n^4)$, and $n^4 \leq n^{2\omega}$.

Online phase, Step 2(c): Let $\{s_1, \ldots, s_m\}$ be the basis computed in the offline phase. Then $m \leq n^2$. In the present step, one sets $y = t_1 s_1 + \cdots + t_m s_m$, with $t_1, \ldots, t_m$ variables, and obtains kn^2 equations in the $m \leq n^2$ variables $t_1, \ldots, t_m$. The complexity is $O(\frac{kn^2}{m} m^\omega)$, and $\frac{kn^2}{m} m^\omega = kn^2 \cdot m^{\omega-1} \leq kn^{2\omega}$.

Online phase, Step 2(d): Using the same arguments as in Step 2(b), the running time of this step is $O(n^{2\omega})$. □

6. Finding an Invertible Solution when There Is One

The results in the previous section assume that we are able to find, efficiently, an invertible matrix in any subspace of $\mathrm{M}_n(\mathbb{F})$ containing an invertible element. This is taken care of by the following Lemma.

Lemma 9 (Invertibility Lemma). *Let $a_1, \ldots, a_m \in \mathrm{M}_n(\mathbb{F})$ be such that*

$$\mathrm{span}\{a_1, \ldots, a_m\} \cap \mathrm{GL}_n(\mathbb{F}) \neq \emptyset.$$

Let S be a finite subset of $\mathbb{F}$. If $\alpha_1, \ldots, \alpha_m$ are chosen uniformly and independently from S, then the probability that $\alpha_1 a_1 + \cdots + \alpha_m a_m$ is invertible is at least $1 - \frac{n}{|S|}$.

Proof. Let

$$f(t_1, \ldots, t_m) = \det(t_1 a_1 + \cdots + t_m a_m) \in \mathbb{F}[t_1, \ldots, t_m],$$

where $t_1, \ldots, t_m$ are scalar variables. This is a determinant of a matrix whose coefficients are linear in the variables. By the definition of determinant as a sum of products of n elements, f is a polynomial of degree n. As $\mathrm{span}\{a_1, \ldots, a_m\} \cap \mathrm{GL}_n(\mathbb{F}) \neq \emptyset$, f is nonzero.

The proof is completed by applying the Schwartz–Zippel Lemma (Lemma below). □

For the reader's convenience, we include a proof for the following classic lemma.

Lemma 10 (Schwartz–Zippel). *Let $f(t_1, \ldots, t_m) \in \mathbb{F}[t_1, \ldots, t_m]$ be a nonzero multivariate polynomial of degree n. Let S be a finite subset of $\mathbb{F}$. If $\alpha_1, \ldots, \alpha_m$ are chosen uniformly and independently from S, then the probability that $f(\alpha_1, \ldots, \alpha_m) \neq 0$ is at least $1 - \frac{n}{|S|}$.*

Proof. We prove the lemma by induction on m.

If $m = 1$, then f is a univariate polynomial of degree n, and thus has at most n roots.

For the inductive step, assume that $m > 1$ and write

$$f(t_1, \ldots, t_m) = f_0(t_2, \ldots, t_m) + f_1(t_2, \ldots, t_m) t_1 + f_2(t_2, \ldots, t_m) t_1^2 + \cdots + f_k(t_2, \ldots, t_m) t_1^k,$$

with $k \leq n$ maximal such that $f_k(t_2, \ldots, t_m)$ is nonzero. The degree of $f_k(t_2, \ldots, t_k)$ is at most $m - k$. For each choice of $\alpha_2, \ldots, \alpha_m \in \mathbb{F}$ with $f_k(\alpha_2, \ldots, \alpha_m) \neq 0$,

$f(t_1, \alpha_2, \ldots, \alpha_m)$ is a univariate polynomial of degree k in the variable t_1. By the induction hypothesis (for $m = 1$), for random $\alpha_1 \in S$, $f(\alpha_1, \alpha_2, \ldots, \alpha_m)$ is nonzero with probability at least $1 - k/|S|$. By the induction hypothesis,

$$\begin{aligned}
&\Pr\big[f(\alpha_1, \ldots, \alpha_m) \neq 0\big] \\
&\quad \geq \Pr\big[f_k(\alpha_2, \ldots, \alpha_m) \neq 0\big] \cdot \Pr\big[f(\alpha_1, \ldots, \alpha_m) \neq 0 \mid f_k(\alpha_2, \ldots, \alpha_m) \neq 0\big] \\
&\quad \geq \left(1 - \frac{n-k}{|S|}\right)\left(1 - \frac{k}{|S|}\right) \geq 1 - \frac{n}{|S|}.
\end{aligned}$$

□

7. Application to the Centralizer KEP

Definition 11. For a group G and an element $g \in G$, the *centralizer of g in G* is the set

$$C_G(g) := \{h \in G : gh = hg\}.$$

The *Centralizer KEP*, introduced by Shpilrain and Ushakov in 2006 [35], is described in Fig. 2. In this protocol, a_1 commutes with b_1 and a_2 commutes with b_2. Consequently, the keys computed by Alice and Bob are identical, and equal to $a_1 b_1 g a_2 b_2$.

As in the Commutator KEP, it is proposed in [35] to use the braid group $\mathbf{B}_N$ as the platform group G. The group elements are chosen in a special way, so as to foil attacks attempted at earlier braid group-based KEPs. We apply the methods developed in the previous sections to obtain an expected polynomial-time cryptanalysis of this KEP. We omit some details, which are similar to those in the earlier sections.

Problem 12 (Centralizer KEP Problem). *Assume that $g, a_1, b_2 \in \mathbf{B}_N$, $g_1, \ldots, g_k \in C_{\mathbf{B}_N}(a_1)$, $h_1, \ldots, h_k \in C_{\mathbf{B}_N}(b_2)$, each of the form $(i, \mathbf{p})$ with $\mathbf{p}$ of length $\leq \ell$. Let a_2 be a product of at most m elements of $\{h_1, \ldots, h_k\}^{\pm 1}$, and let b_1 be a product of at most m elements of $\{g_1, \ldots, g_k\}^{\pm 1}$.*

Given $g, g_1, \ldots, g_k, h_1, \ldots, h_k, a_1 g a_2, b_1 g b_2$, compute $a_1 b_1 g a_2 b_2$.

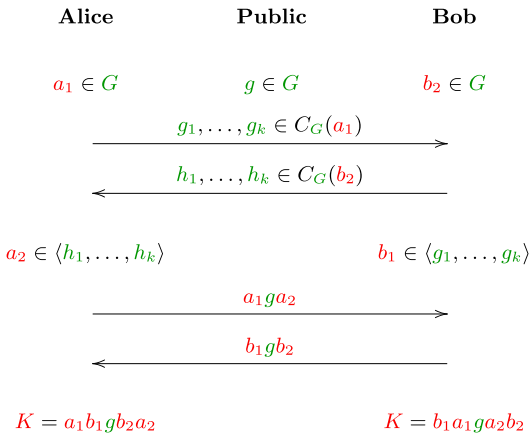

Fig. 2. The centralizer KEP.

7.1. *Solving the Centralizer KEP Problem in Matrix Groups*

For a group G, $Z(G) = C_G(G)$ is the set of all central elements of G. Consider the variation of the Centralizer KEP Problem 12, where the group is $G \leq \mathrm{GL}_n(\mathbb{F})$ instead of $\mathbf{B}_N$. The following variation of this problem is formally harder than this variation.[9]

Problem 13. *Let* $G \leq \mathrm{GL}_n(\mathbb{F})$. *Assume that* $g, a_1, b_2 \in G$, $g_1, \ldots, g_k \in C_G(a_1)$, $h_1, \ldots, h_k \in C_G(b_2)$, $a_2 \in \langle\{h_1, \ldots, h_k\} \cup Z(G)\rangle$, *and* $b_1 \in \langle\{g_1, \ldots, g_k\} \cup Z(G)\rangle$.

Given $g, g_1, \ldots, g_k, h_1, \ldots, h_k, a_1 g a_2, b_1 g b_2$, *compute* $a_1 b_1 g a_2 b_2$.

Following is an algorithm for solving Problem 13. As before, for $S \subseteq \mathrm{M}_n(\mathbb{F})$, $C(S)$ (without subscript) is the centralizer of S *in the matrix algebra* $\mathrm{M}_n(\mathbb{F})$.

Algorithm 14.

1. *Input:* $g, g_1, \ldots, g_k, h_1, \ldots, h_k, a_1 g a_2, b_1 g b_2 \in G$.
2. *Execution:*
   (a) Compute bases for the subspaces $C(g_1, \ldots, g_k)$, $C(C(h_1, \ldots, h_k))$ of $\mathrm{M}_n(\mathbb{F})$.
   (b) Solve
   $$x \cdot g = a_1 g a_2 \cdot y$$
   subject to the linear constraints $x \in C(g_1, \ldots, g_k)$, $y \in C(C(h_1, \ldots, h_k))$.
   (c) Take random linear combinations of the basis of the solution space to obtain solutions (x, y), until y is invertible.
3. *Output:* $x \cdot b_1 g b_2 \cdot y^{-1}$.

Theorem 15. *Let* $G \leq \mathrm{GL}_n(\mathbb{F})$. *Assume that* $|\mathbb{F}|/n \geq c > 1$ *for some constant* c, *and* $k \leq n^2$. *Algorithm 14 is a Las Vegas algorithm for Problem 13, with running time, in units of field operations,* $O(n^{2\omega+2})$.

Proof. The proof is similar to that of Theorem 8.

First, assume that the algorithm terminates. We prove that its output is $a_1 b_1 g a_2 b_2$. As $x \in C(g_1, \ldots, g_k)$ and $b_1 \in \langle g_1, \ldots, g_k\rangle$, x commutes with b_1. As b_2 commutes with $h_1, \ldots, h_k$, $b_2 \in C(h_1, \ldots, h_k)$. As $y \in C(C(h_1, \ldots, h_k))$, y commutes with b_2, and therefore so does y^{-1}. Thus,

$$x b_1 g b_2 y^{-1} = b_1 x g y^{-1} b_2.$$

As $xg = a_1 g a_2 y$, $xgy^{-1} = a_1 g a_2$. Thus,

$$b_1 x g y^{-1} b_2 = b_1 a_1 g a_2 b_2 = a_1 b_1 g a_2 b_2.$$

The analysis of the running time of the algorithm is essentially identical to the analysis in Theorem 8. In Step 2(c), let

$$H = \left\{(x, y) \in C(g_1, \ldots, g_k) \times C\big(C(h_1, \ldots, h_k)\big) : x \cdot g = a_1 g a_2 \cdot y\right\}$$

[9] Since $Z(G) \subseteq C_G(g)$ for every $g \in G$, the mentioned problems are, under mild technical hypotheses perhaps, equivalent. We will not use this feature.

be the solution space, and let $(x_1, y_1), \ldots, (x_d, y_d)$ be a basis for H. As H is a subspace of $\mathrm{M}_n(\mathbb{F}) \times \mathrm{M}_n(\mathbb{F})$, $d \leq 2n^2$. Let $H_2 = \{y : (x, y) \in H\}$, the projection of H on the second coordinate. Then

$$H_2 = \mathrm{span}\{y_1, \ldots, y_d\}.$$

$(a_1, a_2^{-1}) \in H$, and thus $a_2^{-1} \in H_2$. In particular, there is an invertible element in H_2. By the Invertibility Lemma (Lemma 9), a random linear combination of $y_1, \ldots, y_d$ is invertible with probability at least $1/c$. The total expected running time of this step is, therefore, $O(n^4)$, and $n^4 \leq n^{2\omega}$. □

7.2. *Solving the Centralizer KEP Problem in the Braid Group*

We now address Problem 12. We begin by reducing to the case where our braids have a restricted form.

In Sect. 3, we explained how each $x \in \mathbf{B}_N$ can be decomposed (in linear time) as $x = c\tilde{x}$ with c central and $\inf(x) \in \{0, 1\}$.

We may assume that

$$\inf(g) \in \{0, 1\}.$$

Indeed, assume that we have an algorithm solving the problem when $\inf(g) \in \{0, 1\}$. Write $g = c\tilde{g}$ with c central and $\inf(g) \in \{0, 1\}$. Compute

$$c^{-1}a_1 g a_2 = a_1 c^{-1} g a_2 = a_1 \tilde{g} a_2;$$
$$c^{-1}b_1 g b_2 = b_1 c^{-1} g b_2 = b_1 \tilde{g} b_2.$$

Apply the given algorithm to $\tilde{g}, g_1, \ldots, g_k, h_1, \ldots, h_k, a_1\tilde{g}a_2, b_1\tilde{g}b_2$, to obtain $a_1 b_1 \tilde{g} a_2 b_2$. Multiply by c to obtain $a_1 b_1 g a_2 b_2$.

We may, in addition, assume that

$$\inf(g_1), \ldots, \inf(g_k), \inf(h_1), \ldots, \inf(h_k) \in \{0, 1\},$$

since when we apply Algorithm 14 in the image of our group in a matrix group, we have in Problem 13 that

$$\big\langle \{h_1, \ldots, h_k\} \cup Z(G) \big\rangle = \big\langle \{\tilde{h}_1, \ldots, \tilde{h}_k\} \cup Z(G) \big\rangle;$$
$$\big\langle \{g_1, \ldots, g_k\} \cup Z(G) \big\rangle = \big\langle \{\tilde{g}_1, \ldots, \tilde{g}_k\} \cup Z(G) \big\rangle.$$

As in Sect. 3, it follows that

$$a_2, b_1 \in \big[-m(\ell+1), m(\ell+1)\big].$$

Let $u = a_1 g a_2$ and $v = b_1 g b_2$. Decompose $u = c\tilde{u}$ and $v = d\tilde{v}$ with c, d central and $\inf(\tilde{u}), \inf(\tilde{v}) \in \{0, 1\}$. As $g \in [0, \ell+1]$ and $a_1 \in [\inf(a_1), \inf(a_1) + \ell]$,

$$u = a_1 g a_2 \in \big[\inf(a_1), \inf(a_1) + (m+1)(\ell+1) + \ell\big],$$

and thus

$$a_1 g\big(c^{-1}a_2\big) = \tilde{u} \in \big[0, (m+1)(\ell+2)\big];$$
$$c^{-1}a_1 = \tilde{u} a_2^{-1} g^{-1} \in \big[-(m+1)(\ell+1), (m+1)(2\ell+3)\big].$$

Similarly,

$$\left(d^{-1}b_1\right)gb_2 = \tilde{v} \in \left[0, (m+1)(\ell+2)\right].$$

Finally,

$$\begin{aligned} K' &:= a_1\left(d^{-1}b_1\right)gb_2\left(c^{-1}a_2\right) = a_1\tilde{v}\left(c^{-1}a_2\right) \\ &= \left(c^{-1}a_1\right)\tilde{v}a_2 \in \left[-(m+2)(\ell+1), (m+1)(4\ell+6)\right]. \end{aligned}$$

Let $M = (m+2)(4\ell+6)$. Continue as in Sect. 3.

By Theorem 4, we have

$$\left(2^{2NM}t^M\right)\cdot \mathrm{LK}\left(K'\right) \in \mathrm{GL}_n\left(\mathbb{Z}[t]\right),$$

the absolute values of the coefficients in this matrix are bounded by $2^{N^2(M+1)}$, and the maximal degree of t in this matrix is bounded by $2M$. Let p be a prime slightly greater than 2^{N^2M}, and $f(t)$ be an irreducible polynomial over $\mathbb{Z}_p$, of degree d slightly larger than $2M$. Then

$$\left(2^{2NM}t^M\right)\cdot \mathrm{LK}\left(K'\right) = \left(2^{2NM}t^M\right)\cdot \mathrm{LK}\left(K'\right) \bmod \left(p, f(t)\right) \in \mathrm{GL}_n\left(\mathbb{Z}[t]/\left\langle p, f(t)\right\rangle\right),$$

under the natural identification of $\{-(p-1)/2, \ldots, (p-1)/2\}$ with $\{0, \ldots, p-1\}$. Let $\mathbb{F} = \mathbb{Z}[t]/\langle p, f(t)\rangle = \mathbb{Z}[t^{\pm 1}, \frac{1}{2}]/\langle p, f(t)\rangle$. $\mathbb{F}$ is a finite field of cardinality p^d, where d is the degree of $f(t)$. It follows that the complexity of field operations in $\mathbb{F}$ is, up to logarithmic factors, of order

$$d^2 \log p = O\left(M^3N^2\right) = O\left(m^3\ell^3N^2\right).$$

Thus, the key K can be recovered as follows:

1. Apply the composed function $\mathrm{LK}(x) \bmod (p, f(t))$ to

$$g, g_1, \ldots, g_k, h_1, \ldots, h_k, \tilde{u} = a_1 g\left(c^{-1}a_2\right), \tilde{v} = \left(d^{-1}b_1\right)gb_2,$$

   to obtain an input to Problem 13.
2. Solve the problem there, to obtain $\mathrm{LK}(K') \bmod (p, f(t))$.
3. Compute $(2^{2NM}t^M)\cdot \mathrm{LK}(K') \bmod (p, f(t)) = (2^{2NM}t^M)\cdot \mathrm{LK}(K')$.
4. Divide by $(2^{2NM}t^M)$ to obtain $\mathrm{LK}(K')$.
5. Compute K' using the Cheon–Jun inversion algorithm.
6. Multiply by cd to obtain $a_1b_1ga_2b_2$.

8. Further Applications

8.1. *The Braid Diffie–Hellman KEP*

Figure 3 illustrates the well known Diffie–Hellman KEP. Here, G is a cyclic group of prime order, generated by a group element g, and exponentiation denotes ordinary exponentiation.

Interpreting exponentiation in noncommutative groups as conjugation leads to the Ko–Lee–Cheon–Han–Kang–Park *Braid Diffie–Hellman KEP* [21]. For subsets A, B of a group G, $[A, B] = 1$ means that a and b commute, $ab = ba$, for all $a \in A, b \in B$. The Braid Diffie–Hellman KEP is illustrated in Fig. 4. Since, in the Braid Diffie–Hellman

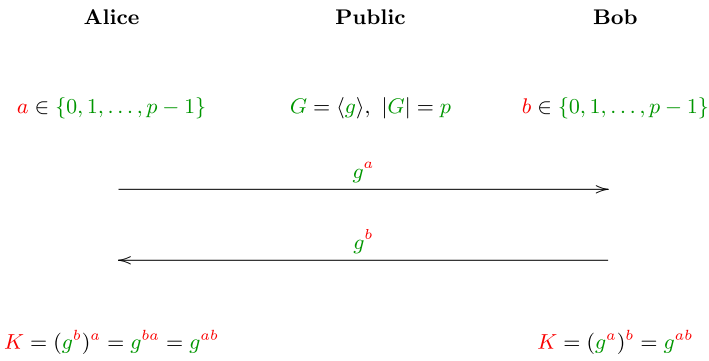

Fig. 3. The Diffie–Hellman KEP.

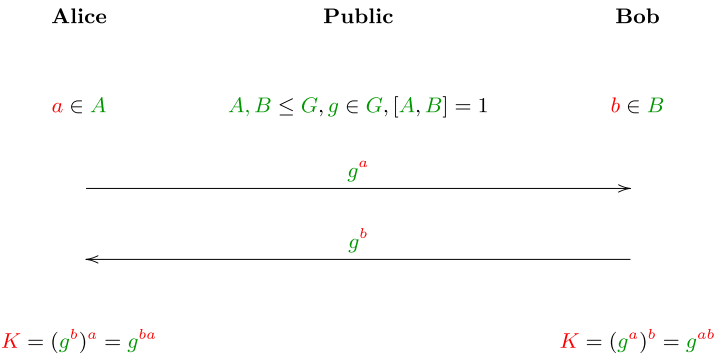

Fig. 4. The braid Diffie–Hellman KEP.

KEP, the subgroups A and B of G commute element-wise, the keys computed by Alice and Bob are identical. It is proposed in [21] to use Artin's braid group $\mathbf{B}_N$ as the platform group G for the Braid Diffie–Hellman KEP, hence the term *Braid* in the name of this KEP.

In the passive adversary model, the security of the Braid Diffie–Hellman KEP for a platform group G (Fig. 4) is captured by the following problem.

Problem 16 (Diffie–Hellman Conjugacy Problem). *Let A and B be subgroups of G with $[A, B] = 1$, and let $g \in G$, be given. Given a pair (g^a, g^b) where $a \in A$ and $b \in B$, find g^{ab}.*

The Cheon–Jun attack on the Braid Diffie–Hellman KEP [8] forms a solution to the Diffie–Hellman Conjugacy Problem in the case where G is the braid group $\mathbf{B}_N$. Their solution can be described, roughly, as follows. Using the methods described in Sect. 4, the problem is reduced to the case where $G \leq \mathrm{GL}_n(\mathbb{F})$, a matrix group over a finite field. Since we are dealing with solutions that are supposed to work for all problem instances, *this problem is not harder than that where* $G = \mathrm{GL}_n(\mathbb{F})$, the group of all

invertible matrices in $\mathrm{M}_n(\mathbb{F})$. However, the latter problem is easy: Assume that $B = \langle b_1, \ldots, b_k \rangle \leq G$. Solve the system

$$\begin{aligned} xg^a &= gx, \\ xb_1 &= b_1 x, \\ &\vdots \\ xb_k &= b_k x \end{aligned}$$

of $(k+1)n^2$ linear equations in the n^2 entries of the unknown matrix x. There is an invertible solution to this system, namely, a. Now, any invertible solution $\tilde{a}$ of this system can be used to compute g^{ab}: By the first equation,

$$g^{\tilde{a}} = \tilde{a}^{-1}(g\tilde{a}) = \tilde{a}^{-1}\left(\tilde{a}g^a\right) = g^a.$$

By the remaining equations, $\tilde{a}$ commutes with the generators of B, and consequently with all elements of B. Thus, we can compute

$$\left(g^b\right)^{\tilde{a}} = g^{b\tilde{a}} = g^{\tilde{a}b} = \left(g^{\tilde{a}}\right)^b = \left(g^a\right)^b = g^{ab}.$$

This essentially establishes that the Diffie–Hellman Conjugacy Problem in this scenario can be solved in time $O(kn^{2\omega})$.

Comparison with Our Approach The reason why the above-mentioned approach of Cheon and Jun is not applicable, as is, to the Commutator KEP or to the Centralizer KEP is that, in either case, there is no prescribed set of generators with which it suffices that the solution commutes: In the Commutator KEP (Fig. 1) it is not clear that a has to commute with anything. In the Centralizer KEP (Fig. 2), we need a_2 to commute with b_2, but b_2 is secret. The main ingredient in our solution, in both cases, is the replacement of membership in a subgroup with membership in the double centralizer (in the full matrix algebra) of that subgroup, and the observation that the latter is efficiently computable. In other words, *instead of adding equations that guarantee that the solution commutes with prescribed elements, we enlarge the set of solutions by moving to the double centralizer*, and prove the increase in the set of solutions is not too large.

The secondary ingredients of our approach also have something to contribute to the Cheon–Jun attack: First, in [8] the dependence of the complexity on the infimum i is exponential (in the bit-length of i). This can be eliminated using the infimum reduction methods of Sects. 3 and 7.2. Second, the fact that the solution to the above-mentioned system of equations is invertible with overwhelming probability is not proved in [8].[10] This gap may be filled using the Invertibility Lemma (Lemma 9). Third, our approach may be used to push most of the work to the offline phase.

Theorem 17. *Assume that $|\mathbb{F}|/n \geq c > 1$. The Diffie–Hellman Conjugacy Problem for a matrix group $G \leq \mathrm{GL}_n(\mathbb{F})$ and $B = \langle b_1, \ldots, b_k \rangle$ is solvable in $O(kn^{2\omega})$ offline time and $O(n^{2\omega})$ online Las Vegas time. More precisely, the running time of the online phase is $O(n^{2\omega})$, plus $O(n^{\omega})$ Las Vegas time.*

[10] An argument involving Zariski density is provided in [8], but this seems to be a heuristic argument; not one intended to be a rigorous proof.

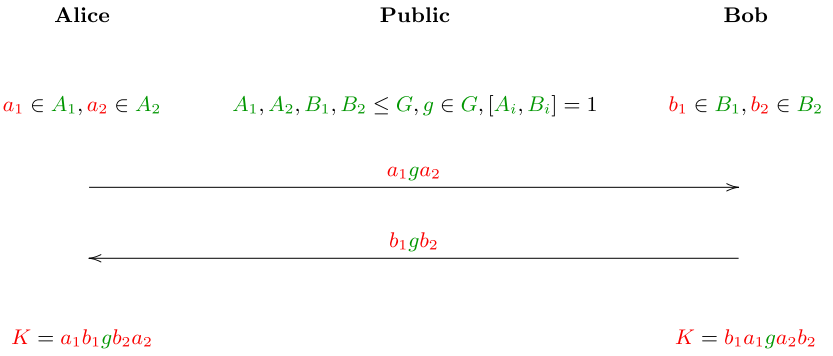

Fig. 5. The double coset KEP.

Proof. Offline phase: Compute a basis for the centralizer $C(B)$ in the matrix algebra $\mathrm{M}_n(\mathbb{F})$, a solution space of a system of kn^2 linear equations in the n^2 entries of the variable matrix x. Since $C(B)$ is a subspace of the vector space $\mathrm{M}_n(\mathbb{F})$, its dimension d is at most n^2. Let $c_1, \ldots, c_d$ be a basis for $C(B)$.

Online phase: Given g^a, solve $xg^a = gx$ subject to $x \in C(B)$, a linear system of n^2 equations in d scalar variables. Let H be the solution space. Let $h_1, \ldots, h_{\tilde{d}}$ be a basis for H. Then $\tilde{d} \leq d$.

There is an invertible element in H, namely: a. By the Invertibility Lemma (Lemma 9), if $t_1, \ldots, t_{\tilde{d}}$ are chosen uniformly and independently from a large subset of $\mathbb{F}$, then the matrix $\tilde{a} = t_1 h_1 + \cdots + t_{\tilde{d}} h_{\tilde{d}}$ is invertible with probability at least $1/c$. Having found such invertible $\tilde{a}$, compute

$$g^{\tilde{a}} = \tilde{a}^{-1}(g\tilde{a}) = \tilde{a}^{-1}\left(\tilde{a}g^a\right) = g^a.$$

The running time of the online phase is $O(n^{2\omega})$, plus $O(n^\omega)$ Las Vegas time for the expected constant number of $n \times n$ matrix inversions. □

Remark 18. In the complexity of the offline phase in Theorem 17, k can be taken to be the minimum among the number of generators of A and the number of generators of B, by exchanging the roles of A and B.

8.2. *Double Coset KEPs*

In 2001, Cha, Ko, Lee, Han and Cheon [7] proposed a variation of the Braid Diffie–Hellman KEP (Fig. 4). For this variation, Cheon and Jun [8] described a convincing variation of their attack. Another variation of this protocol was proposed in 2005, by Shpilrain and Ushakov [34]. Both variations, as well as the Braid Diffie–Hellman KEP, are special cases of the protocol illustrated in Fig. 5.

The methods of Theorem 17 extend to the Double Coset KEP. Here too, the restriction to matrix groups is with no loss of generality, and we obtain an expected polynomial-time solution of the underlying problem in the braid group $\mathbf{B}_N$.

Theorem 19. *Assume that* $|\mathbb{F}|/n \geq c > 1$. *Let* $A_1, A_2, B_1 = \langle b_1, \ldots, b_k \rangle, B_2 = \langle b'_1, \ldots, b'_l \rangle \leq G \leq \mathrm{GL}_n(\mathbb{F})$, *with* $[A_1, B_1] = [A_2, B_2] = 1$, *and* $g \in G$. *After an of-*

fline computation of complexity $O((k+l)n^{2\omega})$, *one can, given* a_1ga_2, b_1gb_2, *compute* $a_1b_1ga_2b_2$ *in time* $O(n^{2\omega})$, *plus* $O(n^{\omega})$ *Las Vegas time.*

Proof. Offline phase: Compute a basis for the centralizers $C(B_1), C(B_2)$ in the matrix algebra $\mathrm{M}_n(\mathbb{F})$, by solving one system of kn^2 linear equations in n^2 variables, and another system of ln^2 linear equations in n^2 variables. Let $c_1, \ldots, c_{d_1}$ be a basis for $C(B_1)$, and $c'_1, \ldots, c'_{d_2}$ be a basis for $C(B_2)$. $d_1, d_2 \le n^2$.

Online phase: Given a_1ga_2, solve $x(a_1ga_2) = gy$ subject to $x \in C(B_1)$, $y \in C(B_2)$, a system of n^2 equations in $d_1 + d_2 \le 2n^2$ scalar variables. Let H be the solution space,

$$H = \left\{(x, y) \in C(B_1) \times C(B_2) : x(a_1ga_2) = gy\right\},$$

and let $(h_1, g_1), \ldots, (h_d, g_d)$ be a basis for H. $d \le d_1 + d_2 \le 2n^2$.

Let $H_1 = \{x : (xy) \in H\}$ be the projection of H on the first coordinate. Then $\{h_1, \ldots, h_d\}$ spans H_1. There is an element $(x, y) \in H$ with x (and y) invertible, namely: (a_1^{-1}, a_2). Thus, there is an invertible element in H_1. By Lemma 9, if $t_1, \ldots, t_d$ are chosen uniformly and independently from a large subset of $\mathbb{F}$, then the matrix $x = t_1h_1 + \cdots + t_dh_d$ is invertible with probability at least $1/c$. Let $\tilde{a}_2 = t_1g_1 + \cdots + t_dg_d$. Then $(x, \tilde{a}_2) \in H$. Compute $\tilde{a}_1 = x^{-1}$. Then

$$\tilde{a}_1g\tilde{a}_2 = x^{-1}(g\tilde{a}_2) = x^{-1}(xa_1ga_2) = a_1ga_2.$$

As $x \in C(B_1)$, $\tilde{a}_1 \in C(B_1)$. Compute

$$\tilde{a}_1b_1gb_2\tilde{a}_2 = b_1\tilde{a}_1g\tilde{a}_2b_2 = b_1a_1ga_2b_2 = a_1b_1ga_2b_2.$$

□

An interesting further application is to Stickel's KEP [36]. This KEP was cryptanalyzed by Shpilrain in [33], describing a heuristic cryptanalysis and supporting it by experimental results. Stickel's KEP is a special case of the Double Coset KEP, where $G = \mathrm{GL}_n(\mathbb{F})$, $A_1 = B_1 = \langle\{a\} \cup Z(G)\rangle$, and $A_2 = B_2 = \langle b\rangle$ (a, b public). By Theorem 19, Shpilrain's cryptanalysis can be turned into a provable Las Vegas algorithm that runs in expected polynomial time, i.e., one supported by a rigorous mathematical proof. In particular, in this way, it is guaranteed that changing the distributions according to which the protocol chooses the involved group elements would not defeat the mentioned polynomial-time cryptanalysis.

9. Additional Comments

Ignoring logarithmic factors, the overall complexity of the algorithms presented here is $n^{2\omega+2} = N^{4\omega+4}$ field operations that are of complexity $m^3\ell^3N^2$. Thus, the complexity of our algorithms is

$$N^{4\omega+6}m^3\ell^3,$$

ignoring logarithmic factors. While polynomial, this complexity is practical only for braid groups of small index N. However, these algorithms constitute the first provable polynomial-time cryptanalyses of the Commutator KEP and of the Centralizer KEP.

The main novelty of our approach lies in the usage of linear centralizers (and double centralizers). However, also the secondary ingredients of our analysis may be of

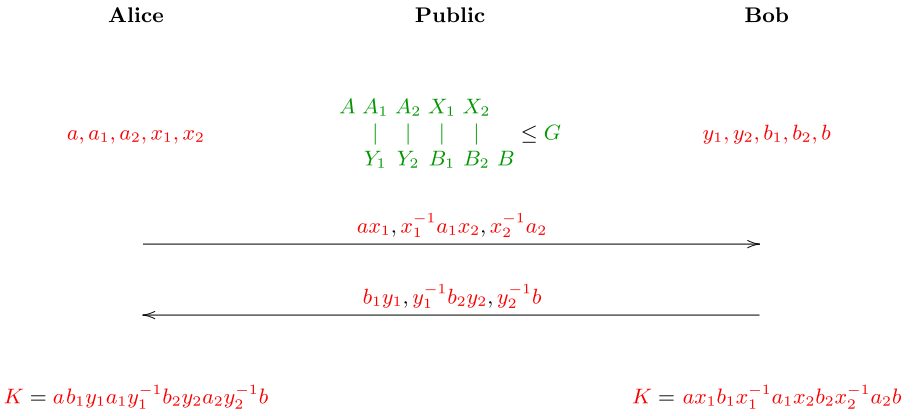

Fig. 6. The triple decomposition KEP.

interest. In particular, we have shown that the Invertibility Lemma can be used to turn the Cheon–Jun cryptanalysis of the Braid Diffie–Hellman KEP [8] and the Shpilrain cryptanalysis of Stickel's KEP [33] into provable Las Vegas algorithm that runs in expected polynomial time, and that the infimum reduction method can be applied to the Cheon–Jun attack to eliminate the exponential dependence on the bit-length of the infimum.

The major challenge is to reduce the degree of N in the polynomial-time cryptanalyses. By Chinese Remaindering or p-adic lifting methods, it may be possible to reduce the complexity contributed by the field operations. Apparently, this may reduce the power of N by 1. It should be possible to make sure that the Invertibility Lemma is still applicable when these methods are used. Much of the complexity comes from the Lawrence–Krammer representation having dimension quadratic in N. Unfortunately, it is conjectured that there are no faithful representations of $\mathbf{B}_N$ of smaller dimension. A more careful analysis of the Lawrence–Krammer representation may yield finer estimates. However, it does not seem that any of these directions would make the attacks practical for, say, $N = 100$.

One may wonder whether, from the *complexity theoretic* point of view, this paper may be the end of braid-based cryptography. Our belief is that this is not the case. For example, consider Kurt's *Triple Decomposition KEP* [28, 4.2.5], described in Fig. 6. In this figure, an edge between two subgroups means that these subgroups commute element-wise. This ensures that the keys computed by Alice and Bob are both equal to $ab_1a_1b_2a_2b$.

We do not, at present, know whether the Triple Decomposition KEP can be cryptanalyzed using the methods presented here, or whether there is a provable, efficient cryptanalysis at all. Additional KEPs to which the present methods do not seem to be applicable are introduced by Kalka in [19] and [20]. There are additional types of braid-based schemes (e.g., authentication schemes) that cannot be attacked using the methods presented here. Some examples are reviewed in the monograph [28].

Changing the platform group in any of the studied KEPs is a very interesting option. There are efficiently implementable, infinite groups with no faithful representations as matrix groups (e.g., the braided Thompson group).[11]

Acknowledgements

I worked on the Commutator KEP, from various other angles, since I was introduced to it at the Hebrew University CS Theory seminar, by Alex Lubotzky [32]. I thank Oleg Bogopolski for inviting me, earlier this year (2012), to deliver a minicourse [37] in the conference *Geometric and Combinatorial Group Theory with Applications* (Düsseldorf, Germany, July 25–August 3, 2012). Preparing this minicourse, I discovered the linear centralizer attack. Initially, I addressed the Centralizer KEP (Sect. 7). When I moved to consider the Commutator KEP, Arkadius Kalka pointed out an obstacle, mentioned by Shpilrain and Ushakov, that struck me as solvable by linear centralizers. I am indebted to Kalka for making the right comment at the right time.

I also thank David Garber, Arkadius Kalka, and Eliav Levy, and the referees, for comments leading to improvements in the presentation of this paper.

References

[1] B. An, K. Ko, A family of pseudo-Anosov braids with large conjugacy invariant sets. arXiv:1203.2320 (2012)

[2] I. Anshel, M. Anshel, D. Goldfeld, An algebraic method for public-key cryptography. *Math. Res. Lett.* **6**, 287–291 (1999)

[3] I. Anshel, M. Anshel, B. Fisher, D. Goldfeld, New key agreement protocols in braid group cryptography, in *CT-RSA 2001*. Lecture Notes in Computer Science, vol. 2020 (2001), pp. 13–27

[4] L. Babai, R. Beals, Á. Seress, Polynomial-time theory of matrix groups, in *ACM STOC* (2009), pp. 55–64

[5] S. Bigelow, Braid groups are linear. *J. Am. Math. Soc.* **14**, 471–486 (2001)

[6] J. Birman, T. Brendle, Braids: a survey, in *Handbook of Knot Theory*, ed. by W. Menasco, M. Thistlethwaite (Elsevier, Amsterdam, 2005), pp. 19–103

[7] J. Cha, K. Ko, S. Lee, J. Han, J. Cheon, An efficient implementation of braid groups, in *ASIACRYPT 2001*. Lecture Notes in Computer Science, vol. 2248 (2001), pp. 144–156

[8] J. Cheon, B. Jun, A polynomial time algorithm for the braid Diffie–Hellman conjugacy problem, in *CRYPTO 2003*. Lecture Notes in Computer Science, vol. 2729 (2003), pp. 212–224

[9] P. Dehornoy, Braid-based cryptography. *Contemp. Math.* **360**, 5–33 (2004)

[10] D. Garber, Braid group cryptography, in *Braids: Introductory Lectures on Braids, Configurations and Their Applications*, ed. by J. Berrick, F.R. Cohen, E. Hanbury, Y.L. Wong, J. Wu. IMS Lecture Notes Series, vol. 19 (National University of Singapore, Singapore, 2009), pp. 329–403

[11] D. Garber, S. Kaplan, M. Teicher, B. Tsaban, U. Vishne, Probabilistic solutions of equations in the braid group. *Adv. Appl. Math.* **35**, 323–334 (2005)

[12] V. Gebhardt, A new approach to the conjugacy problem in Garside groups. *J. Algebra* **292**, 282–302 (2005)

[11] As for *finite* nonabelian groups, we are pessimistic. For example, finite simple groups tend to be linear of small dimension, by the classification of finite simple groups, and our method would reduce the cryptanalysis to the problem of finding an *efficient* linear representation of small dimension. There are at present no signs that such representations must be harder to evaluate (or invert) than, say, computing discrete logarithms in $\mathbb{Z}_p^*$. Indeed, results of Babai, Beals, and Seress seem to indicate otherwise [4].

[13] V. Gebhardt, Conjugacy search in braid groups. *Appl. Algebra Eng. Commun. Comput.* **17**, 219–238 (2006)
[14] R. Gilman, A. Miasnikov, A. Miasnikov, A. Ushakov, New developments in commutator key exchange, in *Proceedings of the First International Conference on Symbolic Computation and Cryptography*, Beijing (2008), pp. 146–150. http://www-calfor.lip6.fr/~jcf/Papers/scc08.pdf
[15] D. Hofheinz, R. Steinwandt, A practical attack on some braid group based cryptographic primitives, in *PKC 2003*. Lecture Notes in Computer Science, vol. 2567 (2002), pp. 187–198
[16] J. Hughes, A. Tannenbaum, Length-based attacks for certain group based encryption rewriting systems, in *SECI02: Sécurité de la Communication sur Internet* (2002). www.ima.umn.edu/preprints/apr2000/1696.pdf
[17] J. Hughes, A linear algebraic attack on the AAFG1 braid group cryptosystem, in *Information Security and Privacy*. Lecture Notes in Computer Science, vol. 2384 (2002), pp. 107–141
[18] A. Kalka, Representation attacks on the braid Diffie–Hellman public key encryption. *Appl. Algebra Eng. Commun. Comput.* **17**, 257–266 (2006)
[19] A. Kalka, Representations of braid groups and braid-based cryptography. PhD thesis, Ruhr-Universität Bochum (2007). www-brs.ub.ruhr-uni-bochum.de/netahtml/HSS/Diss/KalkaArkadiusG/
[20] A. Kalka, Non-associative public key cryptography. 1210.8270 (2012)
[21] K. Ko, S. Lee, J. Cheon, J. Han, J. Kang, C. Park, New public-key cryptosystem using braid groups, in *CRYPTO 2000*. Lecture Notes in Computer Science, vol. 1880 (2000), pp. 166–183
[22] K. Ko, J. Lee, T. Thomas, Towards generating secure keys for braid cryptography. *Des. Codes Cryptogr.* **45**, 317–333 (2007)
[23] D. Krammer, Braid groups are linear. *Ann. Math.* **155**, 131–156 (2002)
[24] S. Lee, E. Lee, Potential weaknesses of the commutator key agreement protocol based on braid groups, in *EUROCRYPT 2002*. Lecture Notes in Computer Science, vol. 2332 (2002), pp. 14–28
[25] S. Maffre, A weak key test for braid-based cryptography. *Des. Codes Cryptogr.* **39**, 347–373 (2006)
[26] A. Miasnikov, V. Shpilrain, A. Ushakov, A practical attack on some braid group based cryptographic protocols, in *CRYPTO 2005*. Lecture Notes in Computer Science, vol. 3621 (2005), pp. 86–96
[27] A. Miasnikov, V. Shpilrain, A. Ushakov, Random subgroups of braid groups: an approach to cryptanalysis of a braid group based cryptographic protocol, in *PKC 2006*. Lecture Notes in Computer Science, vol. 3958 (2006), pp. 302–314
[28] A. Miasnikov, V. Shpilrain, A. Ushakov, *Non-commutative Cryptography and Complexity of Group-Theoretic Problems*. American Mathematical Society Surveys and Monographs, vol. 177 (2011)
[29] A. Miasnikov, A. Ushakov, Length based attack and braid groups: cryptanalysis of Anshel–Anshel–Goldfeld key exchange protocol, in *PKC 2007*. Lecture Notes in Computer Science, vol. 4450 (2007), pp. 76–88
[30] A. Myasnikov, A. Ushakov, Random subgroups and analysis of the length-based and quotient attacks. *J. Math. Cryptol.* **2**, 29–61 (2008)
[31] D. Micciancio, O. Regev, Lattice-based cryptography, in *Post-quantum Cryptography*, ed. by D. Bernstein, J. Buchmann (Springer, Berlin, 2008)
[32] A. Lubotzky, Braid group cryptography, in *CS Theory Seminar*, Hebrew University, March (2001). http://www.cs.huji.ac.il/theorys/2001/Alex_Lubotzky
[33] V. Shpilrain, Cryptanalysis of Stickel's key exchange scheme, in *Computer Science in Russia*. Lecture Notes in Computer Science, vol. 5010 (2008), pp. 283–288
[34] V. Shpilrain, A. Ushakov, Thompson's group and public key cryptography, in *ACNS 2005*. Lecture Notes in Computer Science, vol. 3531 (2005), pp. 151–164
[35] V. Shpilrain, A. Ushakov, A new key exchange protocol besed on the decomposition problem, in *Algebraic Methods in Cryptography*, ed. by L. Gerritzen, D. Goldfeld, M. Kreuzer, G. Rosenberger, V. Shpilrain. Contemporary Mathematics, vol. 418 (2006), pp. 161–167
[36] E. Stickel, A new method for exchanging secret keys, in *Proceedings of the Third International Conference on Information Technology and Applications (ICITA05)* (2005), pp. 426–430
[37] B. Tsaban, *The conjugacy problem: cryptoanalytic approaches to a problem of Dehn*, Minicourse, Düsseldorf University, Germany, July–August 2012. http://reh.math.uni-duesseldorf.de/~gcgta/slides/Tsaban_minicourses.pdf