Research Article

Delaram Kahrobaei, Carmine Monetta, Ludovic Perret, Maria Tota*, and Martina Vigorito

Investigation of metabelian platform groups for protocols based on (simultaneous) conjugacy search problem

https://doi.org/10.1515/jmc-2023-0036
received October 11, 2023; accepted October 15, 2024

Abstract: There are many group-based cryptosystems in which the security is related to the conjugacy search problem or the simultaneous conjugacy search problem in their underlying platform groups. In this article, we show that some metabelian groups do not provide strong security for these cryptosystems and so they cannot be chosen as platform groups.

Keywords: metabelian groups, conjugacy search problem, non-commutative public key cryptography

MSC 2020: 20F14, 94A60

1 Introduction

The field of group-based cryptography began with the seminal work of Anshel et al., in 1999, when they proposed the commutator key-exchange protocol whose security is related to the simultaneous conjugacy search problem (SCSP) [1]. The search for a platform group for this protocol has been an active area including several cryptanalysis. In general, to be a suitable platform group, there are certain properties that the group must satisfy, for instance, we need fast computation of the products and fast comparison of the elements. Indeed, to allow fast computation, the group has to be finitely presented. Also, the problem on which the protocol is based should be hard in the underlying platform. Then, a good candidate could be a polycyclic group. In fact, polycyclic groups are natural generalizations of cyclic groups and cyclic groups have been used in the classical cryptosystems such as Rivest Shamir Adleman [2]. Moreover, polycyclic groups own more complex algorithmic problems and a finite presentation, which provide suitable platforms for cryptography. Actually, Eick and Kahrobaei [3] introduced a new line of investigation for cryptography which has been called polycyclic group-based cryptography. More precisely, they proposed such groups as platform groups for the Commutator Key-Exchange protocol, also known as the Anshel–Anshel–Goldfeld (AAG) Key-Exchange protocol, as well as for the non-commutative Diffie–Hellman (a.k.a. Ko-Lee) Key-Exchange protocol [4]. The security of these protocols is related to SCSP and the conjugacy search problem (CSP) in the platform group G. This means that solving the SCSP in G implies breaking the AAG protocol and that solving the CSP in G implies breaking the Ko-Lee protocol in G. The argument used in the study of Eick and Kahrobaei [3] is based on experimental results for the CSP in certain metabelian polycyclic groups arising from field extensions.

*** Corresponding author: Maria Tota,** Department of Mathematics, University of Salerno, IT, Fisciano SA, Italy, e-mail: mtota@unisa.it
Delaram Kahrobaei: Department of Computer Science, University of York, York, United Kingdom; Departments of Computer Science and Mathematics, Queens College, City University of New York, New York, United States of America; Department of Computer Science and Engineering, Tandon School of Engineering, New York University, New York, United States of America; Initiative for the Theoretical Sciences, Graduate Center, City University of New York, New York, United States of America
Carmine Monetta, Martina Vigorito: Department of Mathematics, University of Salerno, IT, Fisciano SA, Italy
Ludovic Perret: Laboratoire d'Informatique de Paris 6 (LIP6), Sorbonne University, CNRS, LIP6, PolSys, Paris, France

These groups are not virtually nilpotent; hence, the CSP cannot be solved using the analysis provided in the study of Monetta and Tortora [5]. Nevertheless, some of these groups must be avoided as platform since, Kotov and Ushakov [6] cryptanalyzed the AAG Key-Exchange protocol for some metabelian groups of this type, using the so-called Field-Based-Attack (FBA). Then, Gryak et al. [7] investigated a particular class of metabelian groups, i.e., the split metabelian groups of finite Prüfer rank. Indeed, they obtained a complexity result concerning the CSP, which is proved to be at most exponential for the class of groups under investigation. The methods used to test the CSP include experiments conducted with machine learning algorithms, as done by Gryak et al. [8], but also Length-Based-Attack (LBA) [9].

In this context, we got inspired by previous studies [6,7] and, in this study, we extend the arguments in the study of Kotov and Ushakov [6] to a more general class of metabelian groups, including non-polycyclic ones. The same class of groups as the one considered in the study of Gryak et al. [7] and we use a different approach to study the CSP in these groups. Actually, we address the AAG protocol and the Ko-Lee protocol and investigate the possibility of using some particular metabelian groups as platform groups for these protocols. It turns out that the groups under consideration are not suitable because do not provide security for the corresponding Key-Exchange protocols. In fact, we prove the following theorems.

Theorem 4.1: Let $G = M \ltimes N$, where $M \cong \mathbb{Z}^n$ and $N = \mathbb{Z}[m_1^{\pm 1}, \ldots, m_n^{\pm 1}]$ (as additive groups), with $m_1, \ldots, m_n$ positive integers. Then, there exists a polynomial-time algorithm that solves the SCSP, which implies that it is possible to break the Commutator Key-Exchange protocol for such a group G.

Theorem 4.2: Let $G = M \ltimes N$, where $M \cong \mathbb{Z}^n$ and $N = \mathbb{Z}[m_1^{\pm 1}, \ldots, m_n^{\pm 1}]$ (as additive groups), with $m_1, \ldots, m_n$ positive integers. Then, there exists a polynomial-time algorithm that solves the CSP, which implies that it is possible to break the non-commutative Diffie-Hellmann Key-Exchange protocol for such a group G.

This article is structured as follows: in Section 2, we provide the reader with some needed definitions from group theory, and we recall the definitions of some algorithmic problems. Then, we describe the Key-Exchange protocols we are interested in, i.e., the non-commutative Diffie–Hellman and the AAG. Section 3 introduces the family of metabelian groups of interest, along with some examples. In Section 4, we prove Theorems 4.1 and 4.2. The conclusions of our work are in Section 5.

Related works: For an overview of group-based cryptography in the quantum era, refer to [10,11]. Garberet al. [9] demonstrated that the LBA is inefficient for certain classes of metabelian polycyclic groups. However, Kotov and Ushakov [6] explored the security properties of polycyclic groups used as the platform for the Commutator Key-Exchange protocol and showed that, despite the limited success of the LBA, the protocol can still be broken by a deterministic polynomial-time algorithm. This approach, called the FBA, was implemented in groups, algorithms and programming to compare the performance of LBA and FBA.

2 Background

2.1 Algorithmic problems

We start clarifying the difference between a decision problem and a search problem, as this distinction will form the basis for the next definitions. In general, decision problems are problems of the following nature: given a property P and an object O, find out whether or not the object O has the property P. On the other hand, search problems are of the following nature: given a property P and an object O with the property P, find a witness for the property P; for example, given two conjugate elements, find a conjugator. Therefore, in the next definitions, we will assume that the related decision problems are solvable; otherwise, if you cannot even determine whether a solution exists, then it does not make sense to search for that solution.

Here and in the following, if x and g are group elements, the conjugate of g by x is denoted by g^x and it is the element $x^{-1}gx$.

CSP: Let G be a finitely presented group such that the conjugacy decision problem is solvable. Given $g, h \in G$ such that $h = g^x$ for some $x \in G$, the CSP asks to find such an element $x \in G$.
SCSP: Given a finitely presented group G and $g_1,\dots, g_n$, $h_1,\dots, h_n$ elements of G such that $h_i = g_i^x$, for all $i \in \{1, \dots,n\}$ and some $x \in G$, the SCSP asks to recover such a conjugator $x \in G$.

Note that the CSP and the SCSP are always solvable since we assume that the corresponding decision problems are solvable. Also, a solution of $g^x = h$ is not unique. In fact, given a solution x, the set of solutions is $\{ax : a \in C_G(g)\}$, being $C_G(g) = \{a : g^a = g\}$ the centralizer in G of g.

2.2 Polycyclic and metabelian groups

As mentioned in Section 1, the search for the platform group for non-commutative cryptographic protocols is a very active area. In particular, in this context, polycyclic groups show up. Here, we recall the definition.

Definition 2.1. A group G is said to be *polycyclic* if it has a descending chain of subgroups

$$G = G_1 \geq G_2 \geq \dots \geq G_{n+1} = 1,$$

in which each G_{i+1} is a normal subgroups of G_i and the quotient G_i/G_{i+1} is non-trivial cyclic for $i \in \{1, \dots,n\}$.

Related definitions are the following.

Definition 2.2. A group G is called *nilpotent* if it has a central series, i.e., a normal series

$$G = G_1 \geq G_2 \geq \dots \geq G_{n+1} = 1$$

such that G_i/G_{i+1} is contained in the center of G/G_{i+1} for all $i \in \{1, \dots,n\}$.

For example, nilpotent groups have been suggested for multilinear cryptography in previous studies [12,13].

Definition 2.3. A group G is said to be *virtually nilpotent* if it has a non-trivial normal subgroup H such that H is nilpotent and G/H is finite.

Definition 2.4. A group G is said to be *metabelian* if it has a non-trivial normal subgroup H such that both H and G/H are abelian.

In the case of finitely generated metabelian groups, the conjugacy decision problem is solvable [14].

2.3 Non-commutative Diffie-Hellman (a.k.a. Ko-Lee) key-exchange protocol

Originally proposed by Ko et al. in [4] using braid groups, the non-commutative Diffie-Hellman Key-Exchange protocol can be generalized to work over other platform groups. Let G be a finitely presented group, with $A, B \leq G$ such that all elements of A and B commute.

An element $g \in G$ is chosen, and g, G, A, and B are made public. A shared secret can then be constructed as follows:

- Alice chooses a random element $a \in A$ and sends g^a to Bob.
- Bob chooses a random element $b \in B$ and sends g^b to Alice.
- The shared key K is then g^{ab}, as Alice computes $(g^b)^a$, which is equal to Bob's computation of $(g^a)^b$ as a and b commute.

Hence, in order to find K, an attacker needs to obtain a or b, which are private, from g, g^a, and g^b, which are public. In fact, the security of this protocol is related to the CSP.

2.4 AAG Key-Exchange protocol

The AAG Key-Exchange protocol [1] is a two-party protocol performed as follows:

- Fix a finitely presented group G, called the platform group, a set of generators $g_1,\dots,g_k$ for G and some positive integers n_1, n_2, l, m. All this information is made public.
- Alice prepares a tuple of elements $\bar{a} = (a_1, \dots, a_{n_1})$ called Alice's public tuple. Each a_i is generated randomly as a product of g_i's and their inverses.
- Bob prepares a tuple of elements $\bar{b} = (b_1, \dots, b_{n_2})$ called Bob's public tuple. Each b_i is generated randomly as a product of g_i's and their inverses.
- Alice generates a random element A as a product $a_{s_1}^{\varepsilon_1} \dots a_{s_l}^{\varepsilon_l}$, where $a_{s_j} \in \{a_1, \dots, a_{n_1}\}$ and $\varepsilon_j \in \{+1, -1\}$, for all $j \in \{1, \dots, l\}$. The element A (or more precisely its factorization) is called the Alice's private element.
- Bob generates a random element B as a product $b_{t_1}^{\delta_1} \dots b_{t_m}^{\delta_m}$, where $b_{t_j} \in \{b_1, \dots, b_{n_2}\}$ and $\delta_j \in \{+1, -1\}$, for all $j \in \{1, \dots, m\}$. The element B (or more precisely its factorization) is called the Bob's private element.
- Alice publishes the tuple of conjugates $\bar{b}^A = (A^{-1}b_1A, \dots, A^{-1}b_{n_2}A)$.
- Bob publishes the tuple of conjugates $\bar{a}^B = (B^{-1}a_1B, \dots, B^{-1}a_{n_1}B)$.
- Finally, Alice computes the element K_A as a product:

$$A^{-1}(B^{-1}a_{s_1}^{\varepsilon_1}B \dots B^{-1}a_{s_l}^{\varepsilon_l}B) = A^{-1}B^{-1}AB = [A, B]$$

  using the elements of Bob's conjugate tuple $\bar{a}^B$.
- Similarly, Bob computes the element K_B as a product:

$$(A^{-1}b_{t_1}^{\delta_1}A \dots A^{-1}b_{t_m}^{\delta_m}A)^{-1}B = A^{-1}B^{-1}AB = [A, B]$$

  using the elements of Alice's conjugate tuple $\bar{b}^A$.
- It is easy to check that $K = K_A = K_B = [A, B]$ in G. The obtained commutator is the shared key.

The security of this protocol is based on the computational hardness of computing the commutator $[A, B]$ based on the intercepted public information, the tuples $\bar{a}$ and $\bar{b}$ and their conjugates $\bar{b}^A$ and $\bar{a}^B$. In practice, it is often achieved by solving systems of conjugacy equations for A or B, i.e., finding X and Y satisfying the following systems:

$$\begin{cases} X^{-1}b_1X = b_1', \\ \dots \\ X^{-1}b_{n_2}X = b_{n_2}', \end{cases} \tag{1}$$

$$\begin{cases} Y^{-1}a_1Y = a_1', \\ \dots \\ Y^{-1}a_{n_1}Y = a_{n_1}'. \end{cases} \tag{2}$$

This means that the security of this protocol is related to the SCSP.

3 Examples of metabelian groups

Here, we describe the families of groups we are interested in. To be more precise, we consider groups G of the form $G = M \ltimes N$, with both groups M and N abelian. This implies that N is a normal abelian subgroup of G with abelian quotient, i.e., G is metabelian. We use multiplicative notation for the whole group G but additive notation for N. So if $s \in M$ and $c \in N$, the action of the element s maps c to

$$c \cdot s \text{ with additive notation or,}$$
$$c^s = s^{-1}cs \text{ with multiplicative notation.}$$

These kinds of groups arise quite naturally in linear algebra and ring theory, as we will show in more details in the following examples.

Example 3.1. The first example we want to mention is the one introduced by Kotov and Ushakov [6]. Given an irreducible monic polynomial $f(x) \in \mathbb{Z}[x]$, they put:

$$F = \mathbb{Q}[x]/(f). \tag{3}$$

Then, they considered the group $G = F^* \ltimes F$, where F^* is the multiplicative group of F. If $a \in F^*$ and $b \in F$, the action of a maps b to $b \cdot a$. This is a metabelian group, which is also polycyclic, and Kotov and Ushakov [6] showed that in such a group, it is possible to reduce systems (1) and (2) to two systems of linear equations over the field F.

More generally, we can start with a vector space and we obtain the following example.

Example 3.2. Let $V(+,\cdot)$ be a vector space over a field F, and consider the group $G = F^* \ltimes V$, where F^* is the multiplicative group of F. If $\lambda \in F^*$ and $v \in V$, the action of λ maps v to $v \cdot \lambda$. Hence, G has the same structure of the general group we introduced at the beginning of the section. Note that, for $V = F$, if F is of the form described in (3), then we obtain Example 3.1. Similarly, we could start with a module over a commutative unitary ring.

Such examples are interesting from a mathematical point of view but more examples, as they have been described in [7, Section 2], follow.

Example 3.3. *Split metabelian groups of finite Prüfer rank.*

Let m_{ij} be integers, and consider the following group presentation:

$$G = \langle q_1, \dots, q_n, b_1, \dots, b_s | [q_i, q_j] = 1, [b_l, b_t] = 1, b_j^{q_l} = b_1^{m_{1j}} b_2^{m_{2j}} \dots b_s^{m_{sj}} \rangle.$$

Let M the subgroup of G generated by $q_1, \dots, q_n$ and N the normal subgroup of G generated by $b_1, \dots, b_s$. Then, $G = M \ltimes N$ is a split metabelian group of finite Prüfer rank (meaning that the number of generators needed to generate any finitely generated subgroup is bounded). For such a group, it is possible to see that there exists an embedding $N \to \mathbb{Q}^s$. Observe that the action of M on N can be described using integer matrices: the action of q_l is encoded by the $(s \times s)$-matrix M_l with columns $m_{1j}, \dots, m_{sj}$. Moreover, G is polycyclic if and only if the matrices M_l can be taken to be integer matrices with integral inverses [15].

One of the main advantages of these groups is that they admit the following fairly simple set of normal forms:

$$q_1^{\alpha_1} \dots q_n^{\alpha_n} b_1^{\beta_1} \dots b_s^{\beta_s} q_1^{\gamma_1} \dots q_n^{\gamma_n},$$

with $\gamma_1, \dots, \gamma_n > 0$. Moreover, there is an efficient algorithm (collection) to transform any word in the generators to the corresponding normal form: given an arbitrary word in the generating system, move all of the instances of q_i with negative exponent to the left and all the instances of q_i with positive exponents to the right.

Example 3.4. *Generalized metabelian Baumslag–Solitar groups.*

Let $m_1, \dots, m_n$ be the positive integers. We call the group given by the following presentation a generalized metabelian Baumslag–Solitar group:

$$G = \langle q_1, \dots, q_n, b | [q_i, q_j] = 1, b^{q_i} = b^{m_i}, i, j = 1, \dots, n \rangle.$$

It is a split metabelian group of finite Prüfer rank and $G \cong M \ltimes N$ with $M = \langle q_1, \dots, q_n \rangle \cong \mathbb{Z}^n$ and $N = \mathbb{Z}[m_1^{\pm 1}, \dots, m_n^{\pm 1}]$ (as additive groups).

This group represents a natural platform for group-based protocols. Indeed, it possesses good algorithmic properties, as it admits a finite presentation that allows us to work within several proposed cryptosystems.

4 Investigation of platform groups for the commutator and non-commutative Diffie–Hellman key-exchange protocols

We begin studying the CSP and the SCSP in a metabelian group of the form $G = M \ltimes N$, as described in Section 3. Following [7, Section 3], assume that we have conjugated elements $g, h \in G$ and we want to solve the CSP for g, h, i. e., we want to find $x \in G$ such that

$$g^x = h.$$

We put $g = sc$, $h = s'c'$, and $x = td$, where $s, s', t \in M$ and $c, c', d \in N$. Then,

$$g^x = x^{-1}gx = d^{-1}t^{-1}sctd = d^{-1}st^{-1}ctd = s(d^{-1})^s c^t d.$$

Now, $g^x = h$ implies $s' = s$ and $c' = (d^{-1})^s c^t d$. Since the element $(d^{-1})^s c^t d$ belongs to N, we can write it additively as

$$-d \cdot s + c \cdot t + d = d \cdot (1 - s) + c \cdot t.$$

This means that the CSP above is equivalent to the problem of finding $t \in M$ and $d \in N$ such that

$$d \cdot (1 - s) + c \cdot t = c', \tag{4}$$

where $s \in M$ and $c, c' \in N$ are given.

Similarly, (see [6, Section 3]) assume that we have conjugated elements $b_i^X = b_i'$ in G and we want to solve the SCSP, i.e., we want to find such a conjugator $X \in G$. If we put $b_i = s_i c_i$, $b_i' = s_i' c_i'$ with $s_i, s_i' \in M$ and $c_i, c_i' \in N$, for all $i \in \{1, \ldots, n_2\}$, and $X = td$ with $t \in M$, $d \in N$ we can apply the reduction process described above and we end up with the following system of equations:

$$\begin{cases} d \cdot (1 - s_1) + c_1 \cdot t = c_1', \\ \ldots \\ d \cdot (1 - s_{n_2}) + c_{n_2} \cdot t = c_{n_2}', \end{cases} \tag{5}$$

where $s_i \in M$ and $c_i, c_i' \in N$ are given and we need to find $t \in M$ and $d \in N$.

Then, we apply the discussion in [6, Section 3] to the groups described in Example 3.4. In particular, we will show that the generalized metabelian Baumslag–Solitar group is not a suitable platform group for the AAG protocol.

In fact, we have the following theorem.

Theorem 4.1. *Let $G = M \ltimes N$, where $M \cong \mathbb{Z}^n$ and $N = \mathbb{Z}[m_1^{\pm 1}, \ldots, m_n^{\pm 1}]$ (as additive groups), with $m_1, \ldots, m_n$ positive integers. Then, there exists a polynomial-time algorithm that solves SCSP, which implies that it is possible to break the Commutator Key-Exchange protocol for such a group G.*

Proof. In the AAG protocol, the attacker knows $b_1^X, b_2^X, \ldots, b_{n_2}^X$ for some $b_1, \ldots, b_{n_2}$ (which are public) and $n_2 > 1$. To find $X = td$, with $t \in M$ and $d \in N$, the attacker has to solve a system of equations such as (5). Let us consider two equations from the system

$$d \cdot (1 - s) + c \cdot t = c',$$
$$d \cdot (1 - \tilde{s}) + \tilde{c} \cdot t = \tilde{c}',$$

were $s, \tilde{s}, c, \tilde{c}, c'$, and $\tilde{c}'$ are known and the attacker has to find t and d. Recall that $c', \tilde{c}', c, \tilde{c}$, and d lie in N which is a subring of $\mathbb{Q}$. If we identify s and t with the integer they act by, then they also lie in N. So the above can be seen as a system of two equations in N; moreover, we know *a priori* that the system has a solution. This means that unless the second equation is a multiple of the first one, this solution is unique and the standard procedure to solve the system yields then the suitable value of t and d in polynomial time. □

The argument in the previous proof applies also when G is as described in Example 3.2, choosing $V = F^n$ with $n \in \mathbb{N}$. Moreover, if F is as defined in (3), it gives back the FBA described in [6].

Next, we move to the non-commutative Diffie–Hellmann Key-Exchange protocol.

Theorem 4.2. *Let $G = M \ltimes N$, where $M \cong \mathbb{Z}^n$ and $N = \mathbb{Z}[m_1^{\pm 1}, \ldots, m_n^{\pm 1}]$ (as additive groups), with $m_1, \ldots, m_n$ positive integers, then there exists a polynomial-time algorithm that solves CSP, which implies that it is possible to break the Diffie–Hellmann Key-Exchange protocol for such a group G.*

Proof. In the Ko-Lee protocol, the main problem is that Alice and Bob must agree on a set Ω of pairwise commuting elements and then choose their conjugators a and b from that set. Let $a = td$, where $t \in M$ and $d \in N$. As M is abelian, a possible choice would be $\Omega = M$, and if a lies in M, then the attacker can find a from g^a in polynomial time. Another possibility would be to choose $a \in N$. But then $a = d$ and the equation (4) for g^a is

$$d \cdot (1 - s) + c = c',$$

and the only unknown is d, which can be easily found in polynomial time, since we can see it as a linear equation over $\mathbb{Q}$. In the case when a is an arbitrary element lying neither in M nor in N, Ω must be a subset of the centralizer $C_G(a)$ of a in G.

Things are particularly easy when the element a belongs to M^r for some $r \in N$, which happens if and only if

$$a = td = t_1^r = r^{-1}t_1 r = t_1 t_1^{-1} r^{-1} t_1 r = t_1 r^{-t_1} r,$$

for some $t, t_1 \in M$ and $d \in N$. Additively, this is equivalent to

$$d = r - r \cdot t = r \cdot (1 - t).$$

It is a standard fact that $M^r = \{x\delta(x) | x \in M\}$, where δ is the inner derivation given by $\delta(x) = r \cdot (1 - x)$. In this case, it is easy to check that

$$\Omega \subseteq C_G(a) = M^r.$$

If the attacker has the extra information that a belongs to M^r for some r, then the equation that he has to solve is

$$r \cdot (1 - t)(1 - s) + c \cdot t = c'.$$

Equivalently,

$$(c - r + r \cdot s) \cdot t = c' - r + s \cdot r.$$

This can be seen as an equation in $\mathbb{Q}$ and only requires to perform the quotient of $c' - r + r \cdot s$ by $c - r + r \cdot s$ thus can be solved in polynomial time.

Moreover, we are going to see now that by embedding our group G in a bigger group, we may always assume that a lies in some conjugated subgroup of M. Let $\tilde{G} = M \ltimes \tilde{N}$ where $\tilde{N} = N \otimes_{\mathbb{Z}} \mathbb{Q} = \mathbb{Q}$. Then, $a = td$ lies in M^r for some $r \in \tilde{N}$ if and only if $d = r \cdot (1 - t)$. This can always be solved in $\mathbb{Q}$; in other words, we can always find a suitable $r \in \mathbb{Q}$. Then, one proceeds as we did before with this r. The fact that r might not belong to N does not create any troubles: recall that we are dealing not with the conjugacy problem but with the conjugacy *search* problem, meaning that we know a priory that our equations have a solution so the aforementioned procedure yields the right values of t, d even if r does not belong to N.

Observe that behind what we said above is the fact that for the group $\tilde{G}$, the first cohomology group $H^1(M, \tilde{N})$ is zero; thus, all the complements of $\tilde{N}$ in $\tilde{G}$ are conjugated. □

Note that exactly the same argument in the previous proof applies for any group $G = M \ltimes N$ with $N \subseteq \mathbb{Q}^n$ for some n, so it can be extended to the groups described in Example 3.3.

5 Conclusion

We have shown that the generalized metabelian Baumslag–Solitar groups are not suitable platforms neither for the AAG nor for the Ko-Lee protocol.

Actually, our results could also be generalized to all protocols whose security is related to the CSP and the SCSP. For example, the Kahrobaei–Koupparis Digital Signature Scheme [16] and the Khan–Kahrobaei non-commutative ElGamal Key-Exchange [17] (see also [18]).

Nevertheless, we want to point out that our results do not exclude the possibility for a metabelian group to be a suitable platform group for the protocols under consideration.

Acknowledgements: DK thanks the University of Salerno (Italy), where most of this article was discussed and written. We thank Professor Conchita Martinez-Perez for fruitful discussions. MV thanks Initiative for the Theoretical Sciences at CUNY GC, which hosted her Fall 2022. CM, MT, and MV are members of the "National Group for Algebraic and Geometric Structures, and their Applications" (GNSAGA - INdAM). These results have been presented in September 2024 at CIFRIS24, the Italian congress of De Cifris (www.decifris.it/cifris24). The authors acknowledge support of the Institut Henri Poincaré (UAR 839 CNRS-Sorbonne Université), and LabEx CARMIN (ANR-10-LABX-59-01).

Funding information: This research was funded by GNSAGA - INdAM and by the European Union - Next Generation EU, Missione 4 Componente 1 CUP B53D23009410006, PRIN 2022- 2022PSTWLB - Group Theory and Applications. DK has conducted this work partially with the support of ONR Grant 62909-24-1-2002.

Author contributions: All authors have accepted responsibility for the entire content of this manuscript and approved its submission.

Conflict of interest: Dr. Delaram Kahrobaei is a member of the Editorial Board of the Journal of Mathematical Cryptology but was not involved in the review process of this article.

Ethical approval: The conducted research is not related to either human or animal use.

Data availability statement: Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

References

[1] Anshel I, Anshel M, Goldfeld D. An algebraic method for public-key cryptography. Math Res Let. 1999;6:287–91.
[2] Rivest R, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Commun ACM. 1978;2:463–74.
[3] Eick B, Kahrobaei D. Polycyclic groups: A new platform for cryptology?. 2004. arXiv:math/0411077v1 [math.GR].
[4] Ko KH, Lee SJ, Cheon JH, Han JW, Kang J, Park C. New public-key cryptosystem using braid groups. Adv Cryptol CRYPTO 2000;1880:166–83.
[5] Monetta C, Tortora A. The multiple conjugacy search problem in virtually nilpotent polycyclic groups. Adv Group Theory Appl. 2022;13:61–70.
[6] Kotov M, Ushakov A. Analysis of a certain polycyclic-group-based cryptosystem. J Math Cryptol. 2015;9:161–7.
[7] Gryak J, Kahrobaei D, Martinez-Perez C. On the conjugacy problem in certain metabelian groups. Glasgow Math J. 2019;61(2):251–69.
[8] Gryak J, Haralick R, Kahrobaei D. Solving the Conjugacy Decision Problem via Machine Learning. Experiment Math. 2020;29:66–78.
[9] Garber D, Kahrobaei D, Lam HT. Length based attack for polycyclic groups. J Math Cryptol. 2015;9:33–44.
[10] Kahrobaei D, Flores R, Noce M Group-based cryptography in the quantum era. Notices of the American Mathematical Society. 2023. pp. 752–63.

[11] Kahrobaei D, Flores R, Noce M, Habeeb M, Battarbee C. Applications of group theory in cryptography. in: Mathematical Surveys and Monographs. Vol. 278. Providence, Rhode Island: American Mathematical Society; 2024.
[12] Kahrobaei D, Tortora A, Tota M. Multilinear cryptography using nilpotent groups. Elementary theory of groups and group rings and related topics. Proceedings of the Conference Held on 1-2 November 2018 at Fairfield University and Graduate Center, CUNY. De Gruyter Proceedings in Mathematics, Boston, 2018. p. 127–34.
[13] Kahrobaei D, Tortora A, Tota M. A closer look at the multilinear cryptography using nilpotent groups. Int J Comput Math Comput Syst Theory. 2022;7(1):63–7.
[14] Noskov GA. Conjugacy problem in Metabelian groups. Math Notes Acad Sci USSR. 1982;31:252–8.
[15] Auslander L. On a problem of Philip Hall. Ann Math. 1967;86(1):112–6.
[16] Kahrobaei D, Koupparis C. Non-commutative digital signatures using non-commutative groups. Groups Complexity Cryptol. 2012;4:377–84.
[17] Kahrobaei D, Khan B. Nis05-6: A non-commutative generalization of ElGamal key exchange using polycyclic groups. IEEE Global Telecommunications Conference (GLOBECOM ’06). New York: IEEE Press; 2006.
[18] Gryak J, Kahrobaei D. The status of the polycyclic group-based cryptography: A survey and open problems. Groups Complexity Cryptol. 2016;8(2):171–86.