Cryptanalysis of the cryptosystems based on the generalized hidden discrete logarithm problem

Yanlong Ma

Abstract

In this paper, we will show the hidden discrete logarithm problem(HDLP) and the generalized form of HDLP(GHDLP) over non-commutative associative algebras (FNAAs) can be reduced to discrete logarithm problem(DLP) in a finite field through analyzing the eigenvalues of the representation matrix. Through the analysis of computational complexity, we will show that HDLP and GHDLP is not are not good improvements of DLP.With all the instruments in hand, we will show how some schemes based on GHDLP can be broken. Thus we can conclude that, all ideas of constructing cryptographic schemes based on the two problem are of no practical significance.

Keywords: hidden discrete logarithm; generalized hidden discrete logarithm; digital signature; matrix representation.

MSC 2020: 68P25, 68Q12, 68R99, 68W30, 94A60, 16Z05, 14G50.

ACM CCS 2020: F.2.2, E.3

1 Introduction

1.1 Backgrounds

We first recall the integer factorization problem(IFP) and the discrete logarithm problem(DLP):

IFP *Given a big number of the form $n = pq$, find the two prime divisors p and q.*

doi:10.xxxxxxxxxx

DLP *Having known g, h in a cyclic group G, find $t \in N$ such that $g^t = h$.*

IFP and DLP have been used as mathematical base of cryptography for a long time. RSA [1] and ElGamal [2] may be the most famous two. Until now, a lot of digital signatures and cryptosystems are still using the two difficult problems. They are proved by time to be usable for a long time since there is still no polynomial methods to break them in classical computers.

However, quantum computers can solve the two difficult problems in very short time. [3] So new difficult mathematical problems are in urgent need. In this background, a great many problems are proposed and announced to be safe under quantum computers. A possible try is to construct equations with several variables in stead of one, for example, Rainbow in [4] is built based on the difficulty of solving multivariable polynomial systems. The hidden discrete logarithm problem(HDLP) proposed in [5] is also one try to extend the one variable problem DLP to an equation of several variables. The units(or just local units) are used to hide the initial element. This is similar in spirits to the methods in [6]–[10]. Also, in [11]–[13], matrix are used in multivariate schemes to diffuse the initial functions or change the basis to hide the initial one.

1.2 HDLP and GHDLP

HDLP is defined in a finite non-commutative associative algebra (FNAA) in [5], [14].

HDLP *Suppose A is an FNAA, $B \subset A$ is a given subspace. Given two elements x, y in A, find a unit (invertible element) $u \in B$, and an integer t, such that $ux^t u^{-1} = y$, if they exist.*

The answer may not be unique, but any one may be okay since they will give an equivalent key in the schemes. But in most cases, t is unique in $Z/o(x)$, where $o(x)$ is the order or local order of x. This is to say, if (u, t) and (u', t') are solutions to $ux^t u^{-1} = y$, then $x^t = x^{t'}$.

The generalized form of hidden discrete logarithm(GHDLP) was proposed in [15], [16]. It is still defined in an FNAA.

GHDLP *Given two elements x, y in A, $B \subset A$, compute a triple*

(t, u, v), *such that* $u, v \in B$,$ux^t v = y$, *and* $xvu = x$.

Still, the solutions may not be unique, and we need just any one of them.

In the definition above, $xvu = x$ can be replaced with $vux = x$, then a similar problem is proposed, which is still called GHDLP.

1.3 Systems

1.3.1 The KEA

Moldovyan constructed a key exchange agreement(KEA) [5] based on HDLP:

Publicly choose a big prime number p, an positive integer θ, an FNAA A of dimension m over $GF(p^\theta)$, a big commutative subalgebra B of A, and a element $x \notin B$. Now (p, A, m, B, x) are known to all people.

To exchange secrets, Alice choose secretly a unit $g \in B$ together with a secret integer t while Bob choose secretly another unit $h \in B$ and integer s. Now, only Alice knows (g, t) and only Bob knows (h, s). Then Alice compute $k_1 = gx^t g^{-1}$ and send it to Bob. Bob compute $k_2 = hx^s h^{-1}$ and send it to Alice. Now Alice knows (g, t, k_2) and computes $k_A = gk_2^t g^{-1} = ghx^{st} h^{-1} g^{-1}$. Bob knows (h, s, k_1) and computes $k_B = hk_1^s h^{-1} = hgx^{ts} g^{-1} h^{-1}$. Now since g and h are chosen in a commutative subalgebra B of A, $gh = hg$, thus $k_A = k_B$ and they share a common secret $k = k_A = k_B$.

1.3.2 The digital signature

In [5], a digital signature(DS) is constructed.

Suppose H is a publicly known hash function. (p^θ, A, B, m) are also publicly known as above. Now if Alice wants to sign something, she secretly choose a number t, $x, h, g, v \in A$, such that x, g, h do not commute with one another, x is only invertible in B with a local unit v. This is to say, $x \in B$ and there exist some $x' \in B$ such that $xx' = x'x = v$, with $vb = bv = b$, for all $b \in B$. Now (z, y, w) where $z = hxh^{-1}, y = gx^t g^{-1}, w = gvh^{-1}$ are published as public keys, (x, h, g, t) are kept as secret keys.

Now if Alice wants to sign a message M, she will first randomly choose k, compute $u = gx^k h^{-1}, e = H(M, u)$, and then sign M with e and $s = k - te \bmod o(x)$ where $o(x)$ is the order of x.

Suppose Bob get (M, e, s) from Alice, He can verify it in the following procedure. He will compute $u' = y^e wz^s, e' = H(M, u')$ and then check whether $e = e'$. If the signature is valid, then $u' = y^e wz^s = gx^{te} g^{-1} gvh^{-1} hx^s h^{-1} = gx^k h^{-1} = u$. Here we have used that $s + te = k$ and that $xv = vx = x$ because v is a local unit to the subalgebra containing x.

1.4 Existing attacks towards HDLP and GHDLP

There are many attacks towards the two problems.

Moldovyan [17] gives an attack towards HDLP in special cases where x admits a nonzero and non-identity determinant. But this attack can be avoided by taken other elements.

Kuzmin [18] gives a classic attack towards HDLP with time $O(|x|^{1/2})$, where $|x|$ is the multiplication order of the base element x. The invertible element u is also computed. But he did not analyze all the issues. His algorithm is not of high-efficiency. In this paper, all situations are analyzed in detail and better algorithm will be proposed.

In [19], the author gives a quantum attack towards two concrete signatures [20] using HSP(Hidden subgroup problem). This method also works for the signatures in [16], [21], [22]. But it can not deal with all the systems based on HDLP and GHDLP.

1.5 Methods and objectives of this paper

We have two important propositions.

Proposition1 Eigenvalues of a power of a matrix are just the powers of the original eigenvalues.

Proposition2 Conjugation of a matrix do not change the eigenvalues.

So if we use matrix representation of algebra to change HDLP into matrix forms, everything will become clear when we compare the eigenvalues of both sides. For GHDLP, we will show, with a small modification, similar algorithm still works.

Using the method above, we will solve HDLP and GHDLP completely by reducing them to DLP in finite fields, which is of polynomial time using quantum computers [3]. Once HDLP and GHDLP is solved, all the schemes based on it are broken. We will break several representative ones.

2 Cryptanalysis of the cryptosystems

In this section, suppose we can solve HDLP and GHDLP. We will show how to use HDLP and GHDLP to break the cryptosystems based on them.

2.1 Cryptanalysis of the systems in the introduction

2.1.1 Cryptanalysis of the KEA

Suppose Carol wants to obtain Alice and Bob's common secret. By listening to the internet, he can obtain $(p, A, m, B, x, k_1, k_2)$. Now he can solve the HDLP of $k_1 = gx^t g^{-1}$, and then he will know a pair t', g' with $g' \in B$, such that $k_1 = g' x'^{t'} g'^{-1}$. Carol can calculate h' and s' in the same way. Now Carol can compute the secret $g'h'x^{s't'}h'^{-1}g'^{-1} = ghx^{st}h^{-1}g^{-1}$, as one can easily check.

The schemes in [23], [24] can be similarly broken as well.

2.1.2 Cryptanalysis of the DS

Suppose now Carol wants to forge Alice to sign messages. He knows $z = hxh^{-1}, y = gx^t g^{-1}, w = gvh^{-1}$ for some h, x, t, g, v. So if $d = gh^{-1}$, then $y = dz^t d^{-1}$. By solving HDLP, Carol now knows (t', d'), such that $y = d' z^{t'} d'^{-1}$. Then Carol can compute $wz^{k'} = gvh^{-1}hx^{k'}h^{-1} = gx^{k'}h^{-1}$, where k' is randomly chosen. Furthermore, he can compute $e = H(M, u), s = k' - t'e$ and thus he can sign as if he was Alice.

2.2 Cryptanalysis of other cryptosystems

2.2.1 Cryptanalysis of a zero zero-knowledge protocol

In [23], a zero-knowledge protocol based on the general form of HDLP is constructed: Suppose x is locally invertible, a, b, x is known to all, $gab = g$.

Now Alice wants to prove to Bob that she knows the private key (t, s) corresponding to the public key y, where $y = b^t x^s a^t$, they can do as follows:

Step 1 Bob randomly chooses t', s', compute $y' = b^{t'} x^{s'} a^{t'}, z = b^{t'} y^{s'} a^{t'}, h = H(z)$ and sends y', h to Alice.

Step 2 Alice compute $z' = b^t y'^s a^t, h' = H(z')$ and sends z' to Bob if $h = h'$.

Step 3 Bob verifies that $z = z'$.

Now we will disguise ourselves as Alice. This is to say, we can go through the verification of the three steps above.

Now we know (x, y, a, b, H). Solve GHDLP about x and y with linear constraint, we get (u, v, s'') such that $y = ux^{s''}v, ub = bu, va = av$. Then for the (y', h) from Bob, we can compute $z'' = uy'^{s''}v, h'' = H(z'')$,then we send z'' to Bob. Then because $z'' = uy'^{s''}v = ub^{t'}x^{s's''}a^{t'}v = b^{t'}ux^{s''s'}va^{t'} = b^{t'}y^{s'}a^{t'} = z$, we can go through the verification.

Similar arguments goes to the cryptosystems in [24].

2.2.2 Cryptanalysis of the new DS of Moldovyan D.

Moldovyan D. [25] recently proposed a new digital signature.

We still use the system (V, x, p^θ, m, H) Suppose Alice is a signer, she first generate the keys:

Step 1 Select two commutative elements $g, h \in V$, some units a, b, d, f, and some positive integers x, w, s, t.

Step 2 Compute and publish the six elements as public keys:

$$\begin{aligned} y_1 &= ag^x b, \quad z_1 = fh^w a^{-1}, \\ y_2 &= dh^s b, \quad z_2 = fg^t d^{-1}, \\ y &= ahb, \qquad z = fgd^{-1}. \end{aligned} \tag{1}$$

To sign the message M, Alice will randomly choose two integers k, j and compute:(The third line means to divide the bit string e evenly into two parts e_1 and e_2)

$$\begin{aligned} r &= ag^k h^j d^{-1}; \\ e &= H(M, r); \\ e &= (e_1, e_2); \\ u &= \frac{k - xe_1 - te_2 - 1}{e_1 + e_2 + 1}; \\ v &= \frac{j - we_1 - se_2 - 1}{e_1 + e_2 + 1}; \\ s &= b^{-1} g^u h^v f^{-1}. \end{aligned} \tag{2}$$

Finally, the message will be signed as (M, e, s).

To verify the signed message (M, e, s), one can compute

$$\begin{aligned} r' &= (y_1 s z_1)^{e_1} (ysz)(y_2 s z_2)^{e_2}; \\ e' &= H(M, r'). \end{aligned} \tag{3}$$

and verify if $e = e'$.

Now we forge Alice using the public keys $(y_1, z_1, y_2, z_2, y, z)$.

Step 1 Set $m = df^{-1}, n = fb, l = fa^{-1}, \eta = db = mn, g' = fgf^{-1}, h' = fhf^{-1}$.

Step 2 From

$$\begin{aligned} z^{-1} z_2 &= dg^{-1} f^{-1} f g^t d^{-1} = dg^{t-1} d^{-1}; \\ z_2 z^{-1} &= f g^t d^{-1} d g^{-1} f^{-1} = f g^{t-1} f^{-1}. \end{aligned} \tag{4}$$

we have $z^{-1} z_2 m = m z_2 z^{-1}$ and we can compute a m', such that $z^{-1} z_2 m' = m' z_2 z^{-1}$.

Step 3 We have

$$\begin{aligned} m' z_1 y y_2^{-1} &= df^{-1} f h^w a^{-1} a h b b^{-1} h^{-s} d^{-1} = dh^{w-s+1} d^{-1}; \\ y_2^{-1} m' z_1 y &= b^{-1} h^{-s} d^{-1} d f^{-1} f h^w a^{-1} a h b = b^{-1} h^{w-s+1} b. \end{aligned} \tag{5}$$

so $m' z_1 y y_2^{-1} \eta = \eta y_2^{-1} m' z_1 y$ and we can compute a η' such that $m' z_1 y y_2^{-1} \eta' = \eta' y_2^{-1} m' z_1 y$

Step4 By $\eta' = m'n$ we can solve an n' such that $\eta' = m'n'$

Step5 Now

$$\begin{aligned} z_1 y n^{-1} &= f h^w a^{-1} a h b b^{-1} f^{-1} = f h^{w+1} f^{-1}; \\ y n^{-1} z_1 &= a h b b^{-1} f^{-1} f h^w a^{-1} = a h^{w+1} a^{-1}. \end{aligned} \tag{6}$$

we have $z_1 y n'^{-1} l = l y n'^{-1} z_1$, so we can solve an l' such that $z_1 y n'^{-1} l' = l' y n'^{-1} z_1$.

Step6 Rewrite the public key equations, we get:

$$\begin{aligned} l' y_1 n'^{-1} &= f a^{-1} a g^x b b^{-1} f^{-1} &&= f g^x f^{-1} = g'^x; \\ z_1 l'^{-1} &= f h^w a^{-1} a f^{-1} &&= f h^w f^{-1} = h'^w; \\ m'^{-1} y_2 n'^{-1} &= f d^{-1} d h^s b b^{-1} f^{-1} &&= f h^s f^{-1} = h'^s; \\ z_2 m' &= f g^t d^{-1} d f^{-1} &&= f g^t f^{-1} = g'^t; \\ l' y n'^{-1} &= f a^{-1} a h b b^{-1} f^{-1} &&= f h f^{-1} = h'; \\ z m' &= f g d^{-1} d f^{-1} &&= f g f^{-1} = g'. \end{aligned} \tag{7}$$

Then (g', h') is known and we can compute a group of equivalent keys $(g', h', m', n', l', x', w', s', t')$.

The following steps shows how we can sign M as if we were Alice.

Step7 Choose randomly two integers k', j' and compute:

$$\begin{aligned} R &= l^{-1} g'^{k'} h'^{j'} m^{-1}; \\ E &= H(M, R); \\ E &= E_1 || E_2; \\ U &= \frac{k' - x' E_1 - t' E_2 - 1}{E_1 + E_2 + 1}; \\ V &= \frac{j' - w' E_1 - s' E_2 - 1}{E_1 + E_2 + 1}. \end{aligned} \tag{8}$$

Step8 Sign the message M as (M, E, S).

This signature can be verified because

$$\begin{aligned}
R' =&(y_1Sz_1)^{E_1}(ySz)(y_2Sz_2)^{E_2}\\
=&(ag^xbn^{-1}g'^Uh'^Vfh^wa^{-1})^{E_1}(ahbn^{-1}g'^Uh'^Vfgd^{-1})\\
&\cdot(dh^sbn^{-1}g'^Uh'^Vfg^td^{-1})^{E_2}\\
=&(ag^{x+U}h^{w+V}a^{-1})^{E_1}(ag^{U+1}h^Vd^{-1})(dh^{s+V}g^{U+t}d^{-1})^{E_2}\\
=&ag^{(x+U)E_1+U+1+(U+t)E_2}h^{(w+V)E_1+V+(s+V)E_2}d^{-1}\\
=&ag^{U(E_1+E_2+1)+xE_1+tE_2+1}h^{V(E_1+E_2+1)+wE_1+sE_2+1}d^{-1}\\
=&ag^{k'}h^{j'}d^{-1}=R.
\end{aligned} \tag{9}$$

where we have used the equivalency of the keys:

$$\begin{aligned}
&g^x=g^{x'},g^t=g^{t'},h^w=h^{w'},h^s=h^{s'};\\
&g'^x=l'y_1n'^{-1},h'^w=z_1l'^{-1},\\
&h'^s=m'^{-1}y_2n'^{-1},g'^t=z_2m'.
\end{aligned} \tag{10}$$

Signatures in [22], [26]–[28] can be broken similarly.

3 Reduction of HDLP and GHDLP

In this section, we will reduce HDLP and GHDLP in any FNNA into HDLP and GHDLP in matrix form.

3.1 Structure constants

To describe multiplication in an FNNA A, we choose a basis(as a vector space) $\{e_1,...,e_m\}$ of A, then if all the multiplications of any two elements of B is given:$e_i\cdot e_j=\sum_{k=1}^m\Gamma_{i,j}^ke_k$, then we can know all the multiplications of any two elements of V, just by the bi-linearity of the multiplication.

The coefficients of $e_i\cdot e_j$, say, $\Gamma_{i,j}^k$ is called the structure constants of A corresponding to the basis $\{e_1,...,e_m\}$. Clearly, for a given basis, the structure constants and the multiplication determine each other.

3.2 Algebraic representation

A representation of an associative algebra A is by definition an algebraic homomorphism ϕ from A to $End(W)$, the algebra of all linear transformations of W, with trivial addition and composition as multiplication.

Now we consider the left regular representation L, with $L(a) = L_a \in End(A)$, where $L_a(r) = a \cdot r$,for all $r \in A$). Respectively, we can also consider the right regular representation R,with $R(a) = R_a \in End(A)$, where $R_a(r) = r \cdot a$, for all $r \in A$). In most cases the left regular representation is enough, but sometimes the right regular representation is more convenient.

Then L is a homomorphism:$(L(a)L(b))(r) = abr = L(ab)(r)$, so $L(a)L(b) = L(ab)$. The same arguments goes to R, the only difference is that R is an antihomomorphism: $R(ab) = R(b)R(a)$.

Besides the left and right representation, other presentations can also be used, when it is convenient or more natural. For example, when an FNAA is constructed from a group, then the irreducible representations can always extend to the FNAA, which is often of less dimension than the regular representations.

3.3 Representation described as structure constants

Now suppose we are given an algebra A with a basis $\{e_1, ..., e_m\}$, together with the structure constants $\{\Gamma^k_{i,j}\}$. we will determine explicitly the representation, using matrix language.

For any vector $v \in A$, we have $v = \sum_{s=1}^{m} v^s e_s$, for some $v^s \in$

$GF(p^\theta)$, then

$$\begin{aligned} L_v(e_j) &= v \cdot e_j \\ &= (\sum_{s=1}^{m} v^s e_s) \cdot e_j \\ &= \sum_{s=1}^{m} v^s e_s \cdot e_j \\ &= \sum_{s=1}^{m} v^s (\sum_{i=1}^{m} \Gamma_{s,j}^i e_i) \\ &= \sum_{i=1}^{m} (\sum_{s=1}^{m} v^s \Gamma_{s,j}^i) e_i. \end{aligned} \tag{11}$$

Let $c_j^i = \sum_{s=1}^m v^s \Gamma_{s,j}^i$, then we have $L_v(e_j) = \sum_{i=1}^m c_j^i e_i$. So the matrix of L_v is $\{c_j^i\}$,that is, c_j^i lies on the ith row crossing the jth column.

If we identify $(v^1, ...v^m)^T$ with v, and rename L as ϕ, then we get the homomorphism from an FNAA to the matrix algebra: $\phi(v^1, ...v^m)^T \mapsto \{\sum_{s=1}^m v^s \Gamma_{s,j}^i\}_{i,j}$.

3.4 Reduction of HDLP and GHDLP to matrix algebra

In the above subsection, we have shown that any FNAA can be mapped to some matrix algebra. With a ϕ action to both side of $ux^tu^{-1} = y$, one can get $\phi(u)(\phi(x))^t(\phi(u))^{-1} = \phi(y)$. This new HDLP is in matrix algebra. Any solution (u, t) of the initial HDLP will give a solution $(\phi(u), \phi(t))$ to the new HDLP. So if we can compute all possible t in the matrix form, one of them must be a solution to the initial HDLP.

For GHDLP, the equation is $\phi(u)(\phi(x))^t(\phi(v)) = \phi(y)$, with $\phi(x)\phi(v)\phi(u) = \phi(x)$. Any solution (u, v, t) of the initial GHDLP will give a solution $(\phi(u), \phi(v), \phi(t))$ to the new HDLP. So we can still find the t for the initial one if we can find all t for the matrix form.

3.5 Computation of the conjugation element

In this subsection, suppose we have known t for the HDLP and GHDLP.

For HDLP $ux^tu^{-1} = y$, we have $ux^t = yu$. Since t, x, y is known, this is a linear system for the coefficients of u, and thus can be computed quickly.

For GHDLP $ux^tv = y, xvu = x$, we have $ux^t = yu$. Since t, x, y is known, u can be computed quickly. In this case, $xvu = x$ is a linear system for v, and so v can be computed easily.

4 Solving HDLP and GHDLP in matrix algebra

In this section, we will reduce HDLP and GHDLP in matrix algebra into DLP in finite field, and thus solve HDLP and GHDLP in any FNAA considering the last section.

4.1 Solving HDLP in matrix algebra

Rewrite HDLP in matrix form, we get:

HDLP(M) Given two matrix X, Y of dimension m over the field $F = GF(p^\theta)$, find a tuple $(U, t) \in GL(m, F) \times Z/o(X)$, such that $UX^tU^{-1} = Y$, where $o(X)$ is the multiplication order of X.

Our objective is to find all possible t.

Suppose $J_{\lambda,k}$ is the Jordan block with eigenvalue λ of dimension k. Then We have the next lemma.

Lemma 1 The Jordan form of $J^t_{\lambda,k}$ is $J_{\lambda^t,k}$ if $\lambda \neq 0$.

Proof $J^t_{\lambda,k}$ is similar to $J_{\lambda^t,k}$ if and only if $J^t_{\lambda,k} - \lambda^t E$ is similar to $J_{\lambda^t,k} - \lambda^t E$. We can compute $J^t_{\lambda,k} - \lambda^t E = (J_{\lambda,k} - \lambda E)Q$, where $Q = J^{t-1}_{\lambda,k} + \lambda J^{t-2}_{\lambda,k} + ... + \lambda^{t-1}E$ is invertible because it is a sum of nilpotent matrix $J^{t-1}_{\lambda,k} + \lambda J^{t-2}_{\lambda,k} + ... + \lambda^{t-2}J_{\lambda,k}$ and an invertible matrix $\lambda^{t-1}E$ considering that $\lambda \neq 0$. Q commutates with $J_{\lambda,k} - \lambda E$ because they are both polynomials of $J_{\lambda,k}$.

So $(J^t_{\lambda,k} - \lambda^t E)^i = (J_{\lambda,k} - \lambda E)^i Q^i = (J_{\lambda^t,k} - \lambda^t E)^i Q^i$ and thus the two nilpotent matrix $A = J^t_{\lambda,k} - \lambda^t E$ and $B = J_{\lambda^t,k} - \lambda^t E)^i$ satisfies $rank(A^k) = rank(B^k)$ for all $k = 1, ..., m$. So the two matrix are similar.

4.1.1 All eigenvalues of X are 0 or 1

One can easily check that if all eigenvalues of X are 0 or 1, then for $t \geq m$, X^t is similar to X^m, So m is always a suitable solution for t. One will never use this case in cryptosystems.

4.1.2 Other cases

Since $UX^tU^{-1} = Y$, the eigenvalues of X to the power of t will match the eigenvalues of Y.

4.1.3 The procedure of solving HDLP

We give the following steps:

Step 1 Extend the field by the roots of the characteristic polynomial of X.

Step 2 Compute all the eigenvalues of X and Y, and Rewrite them as a vector in the reverse order of multiplicities.

Step 3 Select an eigenvalue λ of X and an eigenvalue σ of Y, whose multiplicity no less than that of λ.

Step 4 Compute the DLP $\lambda^t = \sigma$.

Step 5 If the eigenvalues of X to the power of t match the eigenvalues of Y, keep this t and go to the initial FNNA to compute u. Otherwise, Select another eigenvalue σ of Y and go to **Step 4**.

The steps are of high efficiency because we can find the root of a polynomial in polynomial time. [29]

One can show that using these steps, we can solve HDLP by computing at most m DLPs, which can be done in sub-exponential time [29] with classical computer, or be done in polynomial time with quantum computer [3].

4.2 Solving GHDLP in matrix algebra

4.2.1 Analysis of GHDLP

Rewrite GHDLP in matrix form, we get:

GHDLP(M) Given two matrix X, Y of dimension m over the field $F = GF(p^\theta)$, find a tuple $(U, V, t) \in M(m \times m, F)^2 \times Z/o(X)$, such

that $UX^tV = Y, XVU = X$, where $o(X)$ is the local multiplication order of X. Similar to HDLP,for $t \geq m$, X^t is similar to X^m, So m is always a suitable solution for t in such GHDLP. For other cases, recall the root space decomposition of vector space. We have

$$V = \underset{\lambda \in \mathrm{Spec(X)}}{\oplus} N((X - \lambda E)^{r_\lambda}). \tag{12}$$

where $r_\lambda + 1$ equals to the dimension of Jordan block of eigenvalue λ.

Suppose $v \in N((X - \lambda E)^{r_\lambda})$, then $(X - \lambda E)^{r_\lambda} v = 0$, so

$$\sum_{j=0}^{r_\lambda} (-\lambda)^{r_\lambda - j} \binom{r_\lambda}{j} X^j v_\lambda = 0. \tag{13}$$

Then

$$U(\sum_{j=0}^{r_\lambda} (-\lambda)^{r_\lambda - j} \begin{pmatrix} r_\lambda \\ j \end{pmatrix} X^j V U v_\lambda) = 0. \tag{14}$$

or

$$(UXV - \lambda E)^{r_\lambda}(U v_\lambda) = 0. \tag{15}$$

This is to say, if v_λ is a generalized eigenvector of eigenvalue λ, then Uv_λ is either a generalized eigenvector of eigenvalue λ, or a zero vector. But it can not always be zero, or $Y = UX^tV$ will become zero because all the generalized eigenvectors generate the column space of X. So there is always an eigenvalue λ of X, such that λ^t is an eigenvalue of Y.

4.2.2 Steps for solving GHDLP

We give the following Steps:

Step 1 Extend the field by the roots of the characteristic polynomial of X.

Step 2 Compute all the eigenvalues of X and Y, and Rewrite them as a vector in the reverse order of multiplicities.

Step 3 Select a nonzero eigenvalue λ of X and a nonzero eigenvalue σ of Y.

Step 4 Compute the DLP $\lambda^t = \sigma$.

Step 5 If the eigenvalues of X(some replaced by zero if necessary), to the power of t match the eigenvalues of Y, keep this t and go to the initial FNNA to compute u and v. Otherwise, Select another eigenvalue tuple λ' of X and σ' of Y and go to **Step 4**.

One can show that using these steps, we can solve HDLP by computing at most m^2 DLPs.

5 Conclusion

Now we have completely solve HDLP and GHDLP. We have also break several schemes based on them. Our methods do not use the features of the specific FNAA. So the steps are independent of the fancy designs [30], [31] of the FNAAs.

As we have analyzed, for classical cryptography, there is little improvement from DLP to HDLP and GHDLP considering the efficiency and length of keys; for post-quantum cryptography, HDLP and GHDLP can be solved in polynomial time. Therefore, constructing cryptosystems based on HDLP and GHDLP is of no practical significance in any sense.

References

[1] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[2] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE transactions on information theory*, vol. 31, no. 4, pp. 469–472, 1985.

[3] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[4] J. Ding and D. Schmidt, "Rainbow, a new multivariable polynomial signature scheme," in *ACNS*, vol. 5. Springer, 2005, pp. 164–175.

[5] D. Moldovyan, "Non-commutative finite groups as primitive of public key cryptosystems," *Quasigroups and Related Systems*, vol. 18, no. 2, pp. 165–176, 2010.

[6] C. Tao, A. Diene, S. Tang, and J. Ding, "Simple matrix scheme for encryption," in *Post-Quantum Cryptography: 5th International Workshop, PQCrypto 2013, Limoges, France, June 4-7, 2013. Proceedings 5*. Springer, 2013, pp. 231–242.

[7] J. Ding, A. Petzoldt, and L.-c. Wang, "The cubic simple matrix encryption scheme," in *Post-Quantum Cryptography: 6th International Workshop, PQCrypto 2014, Waterloo, ON, Canada, October 1-3, 2014. Proceedings 6*. Springer, 2014, pp. 76–87.

[8] A. Petzoldt, J. Ding, and L.-C. Wang, "Eliminating decryption failures from the simple matrix encryption scheme," *Cryptology ePrint Archive*, 2016.

[9] A. Szepieniec, J. Ding, and B. Preneel, "Extension field cancellation: a new central trapdoor for multivariate quadratic systems," in *Post-Quantum Cryptography: 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings 7*. Springer, 2016, pp. 182–196.

[10] L.-C. Wang, B.-Y. Yang, Y.-H. Hu, and F. Lai, "A "medium-field" multivariate public-key encryption scheme," in *Topics in Cryptology–CT-RSA 2006: The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13-17, 2005. Proceedings*. Springer, 2006, pp. 132–149.

[11] T. Matsumoto and H. Imai, "Public quadratic polynomial-tuples for efficient signature-verification and message-encryption," in *Advances in Cryptology—EUROCRYPT'88: Workshop on the Theory and Application of Cryptographic Techniques Davos, Switzerland, May 25–27, 1988 Proceedings 7*. Springer, 1988, pp. 419–453.

[12] A. Casanova, J.-C. Faugere, G. Macario-Rat, J. Patarin, L. Perret, and J. Ryckeghem, "Gemss: a great multivariate short signature,"

Ph.D. dissertation, UPMC-Paris 6 Sorbonne Universités; INRIA Paris Research Centre, MAMBA Team ..., 2017.

[13] J. Patarin, "The oil and vinegar signature scheme," in *Dagstuhl Workshop on Cryptography September, 1997*, 1997.

[14] D. N. Moldovyan and N. A. Moldovyan, "A new hard problem over non-commutative finite groups for cryptographic protocols," in *Computer Network Security: 5th International Conference on Mathematical Methods, Models and Architectures for Computer Network Security, MMM-ACNS 2010, St. Petersburg, Russia, September 8-10, 2010. Proceedings 5*. Springer, 2010, pp. 183–194.

[15] D. Moldovyan, "New form of the hidden logarithm problem and its algebraic support," *Buletinul Academiei de Ştiinţe a Moldovei. Matematica*, vol. 93, no. 2, pp. 3–10, 2020.

[16] D. Moldovyan, A. Moldovyan, and N. Moldovyan, "An enhanced version of the hidden discrete logarithm problem and its algebraic support," *Quasigroups and Related Systems*, vol. 28, no. 2, pp. 269–284, 2020.

[17] D. Moldovyan and N. Moldovyan, "Cryptoschemes over hidden conjugacy search problem and attacks using homomorphisms," *Quasigroups and Related Systems*, vol. 18, no. 2, pp. 177–186, 2010.

[18] A. Kuzmin, V. Markov, A. Mikhalev, A. Mikhalev, and A. Nechaev, "Cryptographic algorithms on groups and algebras," *Journal of Mathematical Sciences*, vol. 223, pp. 629–641, 2017.

[19] V. Roman'kov, A. Ushakov, and V. Shpilrain, "Algebraic and quantum attacks on two digital signature schemes," *Journal of Mathematical Cryptology*, vol. 17, no. 1, p. 20220023, 2023.

[20] D. N. Moldovyan, A. A. Moldovyan, and N. A. Moldovyan, "Post-quantum signature schemes for efficient hardware implementation," *Microprocessors and Microsystems*, vol. 80, p. 103487, 2021.

[21] A. Moldovyan and N. Moldovyan, "Post-quantum signature algorithms based on the hidden discrete logarithm problem," *Computer Science Journal of Moldova*, vol. 78, no. 3, pp. 301–313, 2018.

[22] D. Moldovyan, A. Moldovyan, and N. Moldovyan, "Digital signature scheme with doubled verification equation," *Computer Science Journal of Moldova*, vol. 82, no. 1, pp. 80–103, 2020.

[23] D. Moldovyan, "Post-quantum public key-agreement scheme based on a new form of the hidden logarithm problem," *Computer Science Journal of Moldova*, vol. 79, no. 1, pp. 56–72, 2019.

[24] D. N. Moldovyan, N. A. Moldovyan, and A. A. Moldovyan, "Commutative encryption method based on hidden logarithm problem," *Bulletin of the South Ural State University. Series Mathematical Modeling and Programming*, vol. 13, no. 2, pp. 54–68, 2020.

[25] D. Moldovyan, "A new type of digital signature algorithms with a hidden group," *Computer Science Journal of Moldova*, vol. 91, no. 1, pp. 111–124, 2023.

[26] N. Moldovyan, "Finite non-commutative associative algebras for setting the hidden discrete logarithm problem and post-quantum cryptoschemes on its base," *Buletinul Academiei de Ştiinţe a Moldovei. Matematica*, vol. 89, no. 1, pp. 71–78, 2019.

[27] D. Moldovyan, A. Moldovyan, and N. Moldovyan, "A new design of the signature schemes based on the hidden discrete logarithm problem," *Quasigroups and Related Systems*, vol. 29, no. 1, pp. 97–106, 2021.

[28] M. N. Hieu, A. A. Moldovyan, N. A. Moldovyan, and C. H. Ngoc, "A new method for designing post-quantum signature schemes," *Journal of Communications*, vol. 15, no. 10, pp. 747–754, 2020.

[29] E. R. Berlekamp, "Factoring polynomials over large finite fields," *Mathematics of computation*, vol. 24, no. 111, pp. 713–735, 1970.

[30] A. A. Kostina, A. Y. Mirin, D. N. Moldovyan, and R. S. Fahrutdinov, "Method for defining finite noncommutative associative algebras of arbitrary even dimension for development of cryptoschemes," *Informatika i Ee Primeneniya [Informatics and its Applications]*, vol. 14, no. 1, pp. 94–100, 2020.

[31] A. Moldovyan, N. Moldovyan, and V. Shcherbacov, "Noncommutative 6-dimensional associative algebras of two different types," in *Conference on Mathematical Foundations of Informatics*, 2018, pp. 154–163.

Yanlong Ma Received MMM DD, YYYY

Yanlong Ma
Department of Mathematical Sciences Tsinghua University
No.30, Shuangqing Road, Haidian District
Beijing, China
Postcode: 100084
E–mail: 1543537831@qq.com