On Inverse Protocols of Post Quantum Cryptography Based on Pairs of Noncommutative Multivariate Platforms Used in Tandem

Vasyl Ustymenko\textsuperscript{1,2}

\textsuperscript{1} Royal Holloway University of London, United Kingdom.
\textsuperscript{2} Institute of telecommunications and global information space, Kyiv, Ukraine

Abstract
Non-commutative cryptography studies cryptographic primitives and systems which are based on algebraic structures like groups, semigroups and noncommutative rings. We continue to investigate inverse protocols of Non-commutative cryptography defined in terms of subsemigroups of Affine Cremona Semigroups over finite fields or arithmetic rings $\mathbb{Z}_m$ and homomorphic images of these semigroups as possible instruments of Post Quantum Cryptography. This approach allows to construct cryptosystem which are not public keys, when protocol finish correspondents have mutually inverse transformations on affine space $\mathbb{K}^n$ or variety $(\mathbb{K}^*)^n$ where $\mathbb{K}$ is the field or arithmetic ring. The security of such inverse protocol rests on the complexity of word problem to decompose element of Affine Cremona Semigroup given in its standard form into composition of given generators. We discuss the idea of usage combinations of two cryptosystems with cipherspaces $(\mathbb{K}^*)^n$ and $\mathbb{K}^n$ to form a new cryptosystem with the plainspace $(\mathbb{K}^*)^n$, ciphertext $\mathbb{K}^n$ and nonbijective highly nonlinear encryption map.

Keywords: Multivariate Cryptography, Noncommutative Cryptography, stable transformation groups and semigroups, semigroups of monomial transformations, word problem for nonlinear multivariate maps, hidden tame homomorphisms, key exchange protocols, cryptosystems, linguistic graphs

This research is supported by British Academy Fellowship for Researchers at Risk 2022

1. Introduction

Post-Quantum Cryptography (PQC) is an answer to a threat coming from a full-scale quantum computer able to execute Shor’s algorithm. With this algorithm implemented on a quantum computer, currently used public key schemes, such as RSA and elliptic curve cryptosystems, are no longer secure. The U.S. NIST made a step toward mitigating the risk of quantum attacks by announcing the PQC standardisation process\textsuperscript{[1]}. In June 2020, NIST published a list of candidates qualified to the third round of the PQC process. Some public key candidates are implemented like PQC Round 2 candidate called Round 5 (see [2]) or code based classic Mc Eliece algorithm (see [3]). The unique third round candidate defined via Multivariate Cryptography was selected in the category of digital signatures schemes. Noteworthy that during the following NIST project steps an interesting results on cryptanalysis of this candidate known as Unbalanced Rainbow Oil and Vinegar digital signatures schemes were found (see [34], [35], [36]). This scheme is defined via quadratic multivariate public rule, which refers to MiniRank problem Already selected in July of 2022 four cryptosystems are developed not in the area of Applide Algebra. This fact motivates algebraist to continue design of new cryptographical primitives in areas of Noncommutative Cryptography and Multivariate Cryptography.

In March 2021 it was announced that prestigious Abel prize will be shared by A. Wigderson and L.Lovasz. They contribute valuable applications of theory of Expanding...
graphs to Theoretical Computer Science (see [1], [2] and further references). We have been working on applications of these graphs to Cryptography. This paper is dedicated to the usage of geometrical expanders in sense of N. Alon [3] as encryption tools.

In this paper we discuss the development of new cryptosystems within alternative approach ([4], [5], [6]) to construct cryptosystems without usage of public rules. The idea is based on modifications of Diffie-Hellman protocols on the case of multiple generators to construct procedures which output is a pair of mutually inverse multivariate transformations of affine space $K^n$ defined over finite commutative ring $K$. Security of these algorithms rests on the complexity of word problem to decompose given multivariate map into generators of affine Cremona semigroup. The first usage of the complexity of word problem for groups was considered in [8].

In the algorithms of this paper the encryption rule is not given publicly. We introduce new cryptosystems defined in terms of stable semigroups of transformations of affine $K^n$ which consist on transformations of degree bounded by small constant. Main instruments are following. Let $K$ be a commutative ring, $K[x_1, x_2,...,x_n]$ be a ring of polynomials in $n$ variable. Semigroup of endomorphisms $End(K[x_1, x_2,...,x_n])=S(K^n)$ of $K[x_1, x_2,...,x_n]$ is known as Affine Cremona Semigroup, element $f$ of $S(K^n)$ acts naturally on affine space $K^n$ and can be given its standard form $x_1\rightarrow f_1(x_1,x_2,...,x_n), x_2\rightarrow f_2(x_1,x_2,...,x_n), ... , x_n\rightarrow f_n(x_1,x_2,...,x_n)$, where $f_i\in K[x_1, x_2,...,x_n]$.

We assume that $K$ is a finite commutative ring. Symbol $C(K^n)$ stands for Affine Cremona Group of all invertible elements from $S(K^n)$.

Density of the map $f$ is total number of monomial term in all $f_i$.

The computations in subgroups and subsemigroups of $S(K^n)$ are computationally costly because for transformations $g$ and $h$ their "general position" degree of $g(h(x))$ coincides with degree of $h(g(x))$ and equals $deg(g) deg(h)$. The density of $g$ is growing fast when $x$ grows. So special conditions on subsemigroup $S \subset S(K^n)$ needed to make computations feasible.

We know two such conditions

1. **stability condition**, group $G$ such for $g \in G$ maximal degree $deg(g)$ is $d$ (the cases $d=2$ or $d=3$ are probably the most important).

2. **minimality of density condition** (transformation $g \in G$ has to be toric, i.e. its standard form is written as $x_i \rightarrow t_i(x_1, x_2,...,x_n)$, where $t_i$ are monomial expressions. We refer to $g$ as Eulerian map if coefficients are regular and map $g$ is bijective one on the variety $(K^n)^*$. Correspondents use this variety as the plaintext. Let $EG(K)$ be Eulerian group of all such transformations.

PLATFORMS. We discover classes of subgroups of kind (1) or (2) and fast algorithm to generate pairs $g$ and $g^{-1}$. Look at cryptography e-print archive papers [9] and [6] and further references.

Notice that security of Diffie-Hellman algorithm for groups depends not only on abstract group $G$ but on the way of its generation in computer memory. For instance if $G=Z_p^{*}$ is multiplicative group of large prime field then discrete logarithm problem (DLP) is difficult one and guarantees the security of the protocol, if the same abstract group is given as additive group of $Z_{p,1}$ protocol is insecure because DLP will be given by linear equation.

If $G$ is noncommutative group correspondents can use conjugations of elements involved in protocol, some algorithms of this kind were suggested in [10], [11], [12], [13], where group $G$ is given with the usage of generators and relations. Security of such algorithms is connected with Conjugacy Search Problem (CSP) and Power Conjugacy Search Problem (PCSP), which combine CSP and Discrete Logarithm Problem and their generalisations.

This direction belongs to **Non-commutative cryptography** which is active area of cryptology, where the cryptographic primitives and systems are based on algebraic structures like groups, semigroups and noncommutative rings (see [14], [15], [16], [17], [18], [19], [20], [23], [24]). Semigroup based cryptography consist of general cryptographical schemes defined in terms of wide classes of semigroups and their implementations for chosen semigroup families (so called platform semigroups).

Since 2015 several important cryptanalytic results have been obtained in this area ([37]-[42])

As we already mentioned we work with subsemigroups of affine Cremona semigroup $S(K^n)$ on generalisations and modifications of...
Diffie – Hellman protocols for the case of several generators. Elements of the subsemigroup are presented in their standard form of multivariate cryptography.

2. Some schemes of noncommutative cryptography with multivariate platforms

Let $\mathcal{S} < \mathcal{S}(K^n)$ be a subsemigroup of affine Cremona semigroup and $\varphi$ be a homomorphism from $\mathcal{S}$ onto $G < \mathcal{S}(K^n)$, $n > m$.

2.1. Additionally we consider a stable subsemigroup $S' < \mathcal{S}(K^n)$ and assume that $H$ is a stable group $H$, $G < \mathcal{S}(K^n)$. Alice selects elements $s_1, s_2, \ldots, s_r$, $r > 1$ of subsemigroups and computes $\varphi(s_i) = u_i$. She takes invertible elements $hs_i h^{-1}$ and computes $\varphi(s_i) = u_i$. She forms word $u = (a_1)\varphi(i)(a_2)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)$ and sends it to Bob.

He forms word $w = (a_1)\varphi(i)(a_2)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)$ and keeps the word $u = (a_1)\varphi(i)(a_2)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)\varphi(i)$.

Alice computes $u^{-1}$ as $\varphi(h)^{-1} \varphi(h) w f^{-1}$.

So Alice and Bob when the protocol ends have mutually inverse encryption/decryption tools $u^{-1}$ and $u$ for the plainspace $K^n$.

Examples of the implementation of this algorithm can be found in [6].

2.2. Let us consider above algorithms in the case when subsemigroup $S$ consists on toric elements and $H < \mathcal{S}(K^n)$.

Alice forms $h$ and $h^{-1}$ from $\mathcal{S}(K)$ together with pair $f f^{-1}$ from $\mathcal{S}(K)$ and proceed with the modification of algorithm.

Alice selects elements $s_1, s_2, \ldots, s_r$, $r > 1$ of semigroups and computes $\varphi(s_i) = u_i$. She takes invertible elements $h$ and $f$ to form pairs $(a_s h s h^{-1}, b f u f^{-1})$ and sends them to Bob. The rest of the algorithm is identical to case of 2.1.

After the completion of inverse protocol Alice and Bob have bijective maps $u^{-1}$ and $u$ on the plainspace $K^n$.

Security base: The adversary has to solve the word problem for the subsemigroup $S'$, i.e., find the decomposition of $w$ from $S'$ into combinations $a_i, i = 1, 2, \ldots, t$. The general algorithm to solve this problem in polynomial time for the variable $n$ is unknown, as well as a procedure to get its solution in terms of quantum computations. The problem depends heavily on the choice of group.

Remark. Of course in each case alternative ways of computation of the value $\varphi(w)$ of antiisomorphism between semigroup $<a_1, a_2, \ldots, a_r>$ and group $<b_1, b_2, \ldots, b_r>$ given by the rule $\varphi(a_i) = b_i$ have to be investigated.

2.3. On platforms acting in tandem.

2.3.1. Alice and Bob use algorithm 2.1 with output $u^{-1}$ and $u$ on $K^n$ as leading procedure. Supporting procedure is algorithm of kind 2.2 with the same commutative ring $K$ and parameter $m$. Alice (or Bob) deforms the input of 2.2 for her/his correspondent via the change $a_i h$ for $a_i h^2$, $i = 1, 2, \ldots, r’$ where $v = u^{-1}$ or $u$.

Notice that the maps $h_{2V}$ are well defined injective mapsof $\mathcal{S}(K^n)_{\text{pol}}$ into $K^n$, they have polynomial density.

Remark. Encryption and decryption functions of the above algorithm can be treated as polynomial maps of $K^n$ to $K^n$ because elements of $\mathcal{S}(K^n)$ act naturally on $K^n$. Between encryption and decryption functions there is a density gap because decryption map is not a transformation of polynomial density. Such pairs can be used as non-bijective stream ciphers in a spirit of [25]. In the tandem procedure interception of plaintexts with corresponding ciphertexts are unfeasible without the computation of $\varphi(w)$.

2.3.2 Alice and Bob can use algorithm 2.2 with output $u^{-1}$ and $u$ on $(K^n)_{\text{pol}}$ as leading procedure. Supporting procedure is algorithm of
3. On groups and semigroups defined in terms of linguistic graphs

3.1. On linguistic graphs over commutative rings and skating on them.

The missing definitions of graph-theoretical concepts which appear in this paper can be found in [28]. All graphs we consider are simple graphs, i.e. undirected without loops and multiple edges. Let $V(G)$ and $E(G)$ denote the set of vertices and the set of edges of $G$ respectively.

When it is convenient we shall identify $G$ with the corresponding anti-reflexive binary relation on $V(G)$, i.e. $E(G)$ is a subset of $V(G) \times V(G)$.

We refer to $\{ (x,y) \mid x \in V(G), y \in V(G) \}$ as degree of the vertex $v$.

The incidence structure is the set $V$ with partition sets $P$ (points) and $L$ (lines) and symmetric binary relation $I$ such that the incidence of two elements implies that one of them is a point and another one is a line. We shall identify $I$ with the simple graph of this incidence relation or bipartite graph. The pair $x, y, x \in P, y \in L$ such that $x I y$ is called a flag of incidence structure $I$.

Let $K$ be a finite commutative ring. We refer to an incidence structure with a point set $P=P_{s,m}=K^{s+m}$ and a line set $L=L_{s,m}=K^{s+m}$ as linguistic incidence structure, if point $x=(x_1, x_2, \ldots, x_s, y_{r+1}, y_{r+2}, \ldots, y_{r+m})$ is incident to line $y=(y_1, y_2, \ldots, y_s, y_{r+1}, y_{r+2}, \ldots, y_{r+m}) I$ if and only if the following relations hold:

$$a_1x_1+y_1y_{r+1}=f_1(x_1, x_2, \ldots, x_s, y_1, y_2, \ldots, y_r)$$

$$a_2x_2+y_2y_{r+2}=f_2(x_1, x_2, \ldots, x_s, y_{r+1}, y_1, y_2, \ldots, y_{r+1})$$

$$\vdots$$

$$a_mx_m+b_my_{r+m}=f_m(x_1, x_2, \ldots, x_s, x_{r+1}, \ldots, x_{r+m}, y_1, y_2, \ldots, y_r, y_{r+1}, \ldots, y_{r+m})$$

where $a_j$ and $b_j, j=1,2,\ldots,m$ are not zero divisors, and $f_j$ are multivariate polynomials with coefficients from $K$ [29]. Brackets and parenthesis allow us to distinguish points from lines.

The colour $\rho(x)=\rho((x))$ ($\rho(y)=\rho([y])$) of point $x$ (line $[y]$) is defined as projection of an element $(x)$ (respectively $[y]$) from a free module on its initial $s$ (relatively $r$) coordinates. As it follows from the definition of linguistic incidence structure for each vertex of incidence graph there exists unique neighbour of a chosen colour.

We refer to $\rho((x))=(x_1, x_2, \ldots, x_s)$ for $(x)=(x_1, x_2, \ldots, x_{s+m})$ and $\rho([y])=(y_1, y_2, \ldots, y_r)$ for $[y]=[y_1, y_2, \ldots, y_{r+m}]$ as the colour of the point and the colour of the line respectively. For each $b \in K$ and $p=(p_1, p_2, \ldots, p_{s+m})$ there is a unique neighbour of the point $[l]=N_0(p)$ with the colour $b$. Similarly for each $c \in K$ and line $l=[l_1, l_2, \ldots, l_{r+m}]$ there is a unique neighbour of the line $(p)=N(l)$ with the colour $c$. The triples of parameters $s, r, m$ defines type of linguistic graph.

We consider also linguistic incidence structures defined by infinite number of equations.

Linguistic graphs are defined up to isomorphism. We refer to written above equations as canonical equations of linguistic graph.

In the case of linguistic graph defined over commutative ring the walk consisting of its vertices $v_0, v_1, v_2, \ldots, v_k$ is uniquely defined by initial vertex $v_0$ and colours $\rho(v_i)$, $i=1,2,\ldots,k$ of other vertices from the path. We consider the equivalence relations on partition sets such that $(p)=\rho([l]) \equiv [l']$ if $p_{i+r}=p'_{i+r}$, $(l_{i+r}=l'_{i+r})$ for $i=1,2,\ldots,m$.

We define jump operator $J(p,a)$, $aK'$ on partitions set $P(J[l,a], aK'$ on partition set $L)$ by conditions $J(p,a)=(p)$ and $\rho(J(p,a))=a$ ($J([l],a)=a$ and $\rho(J([l],a))=a$).

Already defined neighbour computation operator (or ground moving operator) $N(v,a)$ acts on $PUL$ by rules $N(p,a)=\{l\}$ where $(p)[l]\approx [l']$ and $\rho([l],a)=a$ and $N([l],a)=(p)$ where $(p)[l]$, $\rho((p))=a$.

Let us consider skating chain of the linguistic graph with starting point $p$ which is a sequence $(p, p_0, l_1, l_2, p_3, p_4, \ldots, l_b, l_{b+1}, p_{b+1})$, $t=4k, k \geq 0$ such that $p_{i+4}=p_{i+1}$, $i \geq 0, p_{i+2}=p_{i+1}$, $i \geq 0, p_{i+2}=p_{i+1}$ and $p_{i+2}=p_{i+1}$ for $k \geq 0$.

Colours of elements from the skating chain and the starting point determine the sequence. Obviously sequence of alternating jump operators $J_{l}$ and ground moving operators form the skating chain from starting point $(p)$. In fact term skating chain is selected because of the similarity of computation the sequence with competitions on skating boards, roller skates,
figure skating (various jumps and skate surface moves).

### 3.2. Semigroups of infinite symbolic strings and linguistic compression maps.

Let us consider semigroup \( S(K') \) and the totality \( S^t(K') \) of maps of kind \( G: (y_1, y_2, ..., y_t) \rightarrow (f_1(y_1, x_1, x_2, ..., x_{r_1}), f_2(x_1, x_2, ..., x_{r_2}), ..., f_s(x_1, x_2, ..., x_{r_s})), \) where \( H \in S(K') \), then \( G(H) \) for \( G \in S^t(K') \) is the map \( (y_1, y_2, ..., y_t) \rightarrow (f_1(H(x_1), x_2, ..., x_{r_1}), f_2(H(x_1), x_2, ..., x_{r_2}), ..., f_s(H(x_1), x_2, ..., x_{r_s})). \)

When it is convenient we will identify elements of \( S(K') \) with tuples from \( K[x_1, x_2, ..., x_s] \) and elements of \( S^t(K') \) with tuples of \( K[x_1, x_2, ..., x_s] \).

Let us consider a to totality \( 'BS_t(K) \) as sequences of kind \( u=(H_0, G_1, G_2, H_3, ..., H_{t-1}, H_t) \), \( t=4i \) where \( H \in S(K') \).

We define a product of \( u \) with \( u'=(H'_0, G'_1, G'_2, H'_3, G'_4, ..., H'_{t-1}, H'_t) \) as \( w=(H_0, G_1, G_2, H_3, ..., H_{t-1}, H_t), H'_0, H'_1, H'_2, H'_3, G'_4, ..., H'_{t-1}, H'_t) \).

Elements of kind are \( \{H_0, G_1, G_2, H_3, H_4 \} \) are generators of the semigroup.

The refer to generator with \( H_r=H_0 \) as loop element. Let \( L=L_0(H) \) be the totality of loop elements. The semigroup generated by loop elements is isomorphic to the free semigroup \( F(L)=F_t(K) \) of words in the alphabet \( L \). We refer to \( F(L) \) as group of loop strings.

It is easy to see that \( 'BS_t(K) \) is isomorphic to semidirect product of \( F(L) \) and affine Cremona semigroup \( S(K') \).

Let us consider the homomorphism of the group \( 'BS_t(K) \) into Cremona semigroup \( S(K') \).

Semigroup \( S(K_t^{*n}) \) defined in terms of linguistic graph \( I_t=I_t^n(K) \). Notice that one can consider graph \( I_t(K) \) over the extension \( K' \) of \( K \) with the usage of the same equations. Let us take \( K=[x_1, x_2, ..., x_{m+s}] \) where \( x_i \) are formal variables and consider an infinite graph \( I_t^m(K(x_1, x_2, ..., x_{m+s}), n=m+s \) with partition sets \( P'=K[x_1, x_2, ..., x_{m+s}], \) and \( L'=K(x_1, x_2, ..., x_{m+s}) \). After that we take a bipartite string \( u=(H_0, G_1, G_2, H_3, H_4, G_5, G_6, ..., H_{t-1}, H_t) \) formed by a totality of multivariate polynomials from the subring \( K[x_1, x_2, ..., x_n] \) of \( K[x_1, x_2, ..., x_n] \) and the point \( (x)=(x_1, x_2, ..., x_n) \).

We refer to \( 'BS_t(K) \rightarrow S(K') \) as a chain transitions semigroup of linguistic graph \( I(K) \) and to map \( \psi \) as linguistic compression map. Notice that in the case of finite commutative ring \( \psi \) maps infinite semigroup into finite set of chain transitions.

### 3.3. Some subsemigroups of symbolic strings and their homomorphic linguistic graphs over commutative rings and skating on them.

We define subsemigroup \( GS_t(K) \) of symbolic ground strings as a totality of bipartite strings \( u=(H_0, G_1, G_2, H_3, G_4, G_5, ..., H_{t-1}, H_t) \) in \( 'BS_t(K) \) with \( H_0=H_0, G_1=G_2, H_3=H_4, G_5=..., H_{t-1}=H_{t-1} \) and refer to \( \psi(GS_t(K))=GCT_t(K) \) as semigroup of ground chain transitions on linguistic graph \( I \).

Let us assume that \( H \) is its bijective map and its inverse is a polynomial map (in the case of infinite ring \( K \)). Then we can consider a reverse bigraded string \( Rev(u)=H, G_2, H_3, G_4, ..., G_{t-1}, H_t \) and refer to \( \psi(GS_t(K))=GCT_t(K) \) as semigroup of ground chain transitions on linguistic graph \( I \).

Let us consider the homomorphism of the group \( 'BS_t(K) \) into Cremona semigroup \( S(K') \).

Semigraph \( S(K_t^{*n}) \) defined in terms of linguistic graph \( l_t=I_t(K) \). Notice that one can consider graph \( I_t(K) \) over the extension \( K' \) of \( K \) with the usage of the same equations. Let us take \( K=[x_1, x_2, ..., x_{m+s}] \) where \( x_i \) are formal variables and consider an infinite graph \( I_t^m(K(x_1, x_2, ..., x_{m+s}), n=m+s \) with partition sets \( P'=K[x_1, x_2, ..., x_{m+s}], \) and \( L'=K(x_1, x_2, ..., x_{m+s}) \). After that we take a bipartite string \( u=(H_0, G_1, G_2, H_3, H_4, G_5, G_6, ..., H_{t-1}, H_t) \) formed by a totality of multivariate polynomials from the subring \( K[x_1, x_2, ..., x_n] \) of \( K[x_1, x_2, ..., x_n] \) and the point \( (x)=(x_1, x_2, ..., x_n) \).

We refer to \( 'BS_t(K) \rightarrow S(K') \) as a chain transitions semigroup of linguistic graph \( I(K) \) and to map \( \psi \) as linguistic compression map. Notice that in the case of finite commutative ring \( \psi \) maps infinite semigroup into finite set of chain transitions.

**Lemma 1 [44].** The map \( \psi: 'BS_t(K) \rightarrow S(K') \) is a homomorphism of semigroups.

We refer to \( \psi('BS_t(K))=CT_t(K) \) as a chain transitions semigroup of linguistic graph \( I(K) \) and to map \( \psi \) as linguistic compression map. Notice that in the case of finite commutative ring \( \psi \) maps infinite semigroup into finite set of chain transitions.

### 3.3. Some subsemigroups of symbolic strings and their homomorphic linguistic graphs over commutative rings and skating on them.

We define subsemigroup \( GS_t(K) \) of symbolic ground strings as a totality of bipartite strings \( u=(H_0, G_1, G_2, H_3, G_4, G_5, ..., H_{t-1}, H_t) \) in \( 'BS_t(K) \) with \( H_0=H_0, G_1=G_2, H_3=H_4, G_5=..., H_{t-1}=H_{t-1} \) and refer to \( \psi(GS_t(K))=GCT_t(K) \) as semigroup of ground chain transitions on linguistic graph \( I \).

Let us assume that \( H \) is its bijective map and its inverse is a polynomial map (in the case of infinite ring \( K \)). Then we can consider a reverse bigraded string \( Rev(u)=H, G_2, H_3, G_4, ..., G_{t-1}, H_t \) and refer to \( \psi(GS_t(K))=GCT_t(K) \) as semigroup of ground chain transitions on linguistic graph \( I \).

**Lemma 2 [44].** The homomorphic image \( \psi('BR_t(K))=BCT_t(K) \) is a subgroup of affine Cremona group \( C(K') \).

Really \( \psi(u-Rev(u), u \in 'BR_t(K)) \) is an identity map.

We refer to \( BCT_t(K) \) as subgroup of bijective chain transitions of linguistic graph \( I \).
4. On semigroups and groups related to Double Schubert graphs and corresponding inverse protocols


We define Double Schubert Graph $DS(k,K)$ over commutative ring $K$ as incidence structure defined as disjoint union of partition sets $PS=K^{(k+1)}$ consisting of points which are tuples of kind $x=(x_1, x_2, ..., x_k, x_{k+1}, x_{k+2}, ..., x_{2k})$ and $LS=K^{(k+1)}$ consisting of lines which are tuples of kind $y=[y_1, y_2, ..., y_k, y_{k+1}, y_{k+2}, ..., y_{2k}]$, where $x$ is incident to $y$, if and only if $x_jy_k=x_iy_i$ for $i=1$, $2$, ..., $k$ and $j=1$, $2$, ..., $k$. It is convenient to assume that the indices of kind $i, j$ are placed for tuples of $K^{(k+1)}$ in the lexicographical order.

**Remark.**

The term Double Schubert Graph is chosen, because points and lines of $DS(k,F_p)$ can be treated as subspaces of $F_{p^{(k+1)}}$ of dimensions $k+1$ and $k$, which form two largest Schubert cells. Recall that the largest Schubert cell is the largest orbit of group of untrangular matrices acting on the variety of sets of given dimensions. We will consider these connection in details in the next section.

We define the colour of point $x=(x_1, x_2, ..., x_k, x_{k+1}, x_{k+2}, ..., x_{2k})$ from $PS$ as tuple $(x_1, x_2, ..., x_k)$ and the colour of a line $y=[y_1, y_2, ..., y_k, y_{k+1}, y_{k+2}, ..., y_{2k}]$ as the tuple $(y_1, y_2, ..., y_k)$. For each vertex $v$ of $DS(k,K)$, there is the unique neighbouring $N_v(v)$ of a given colour $a=(a_1, a_2, ..., a_k)$. It means the graphs $DS(k,K)$ form a family of linguistic groups.

Let us consider the subsemigroup $\mathcal{Y}(d,K)$ of $BS_2(K)$ consisting of strings $u=(H_0, G_1, G_2, H_3, H_4, G_5, G_6, ..., H_{i_1}, H_{i_2})$ such that maximum of parameters $deg(H_0)+deg(G_1)$, $deg(G_1)+deg(G_2)+deg(H_3)$, $deg(H_2)+deg(G_3)$, $deg(G_3)+deg(H_4)$, $deg(G_4)+deg(H_5)$, $deg(H_5)+deg(H_6)$, $deg(H_6)$ equals to $d$, $d>1$.

**Theorem 1.** Let $I(K)$ be an incidence relation of Double Schubert graph $DS(k,K)$. Then $\mathcal{Y}(d,K)\mathcal{U}(d,K)$ form a family of stable semigroups of degree $d$.

The proof is based on the fact that chain transition $u$ from $\mathcal{U}(d,K)$ moves $x_i$ into expression $x_{i+T(u)}$, where $T(u)$ is a linear combination of products $\mathcal{F}(x_1, x_2, ..., x_k)$, $\mathcal{G}(y_1, y_2, ..., y_k)$ where $deg(f)+deg(g)<d$.

New semigroup $\mathcal{U}(d,K)$ consists of transformations of the free module $K^r$, $r=(k+1)k$. If $d=2$ then $\mathcal{U}(d,K)$ contains semigroup of quadratic transformations defined in [9], which consists of ground chain transitions.

Let $J$ be subset of the Cartesian square of $M=\{1,2, ..., k\}$. We can identify its element $(i,j)$ with the index $i$ of Double Schubert Graph $DS(k,K)$.

**Proposition 1 [44].** Each subset $J$ of $M^2$ defines symplectic homomorphism $\delta$ of $DS(k,K)$ onto linguistic graph $DS^J(k,K)$.

It is easy to see that in the case of empty set corresponds to complete bipartite graph with the vertices sets $K^2UK^2$.

**Corollary 1.** Let $I(J,K)$ be an incidence relation of linguistic graph $DS^J(k,K)$. Then $\mathcal{Y}(d,K)^\mathcal{U}(d,K)$ forms a family of stable semigroups of degree $d$.

4.2. Implementation of inverse protocols and their extensions with double Schubert graphs and their symplectic homomorphisms.

Let us consider the implementation of algorithm 2.1 in the case of $S=S^\ast$ and $G=H$. We consider the family of graphs $DS(k,K)$ and form the family $DS_{\mathcal{U}}(k,K)$. We assume that $j(k)=|J(k)|$ and $c(k)=c'(k)<c(k)$ for some constants $0<c'<c<1$. We set $S=\mathcal{U}(\mathcal{Y}(d,K))^\mathcal{U}(d,K)$ which is a subgroup of affine Cremona group $C(k^n)$, $n=k^2+2$ and $G=k^2+2$. Alice selects elements $u=(H_0, G_1, G_2, H_3, H_4, G_5, G_6, ..., H_{i_1}, H_{i_2})$, $i_1=1,2, ..., r, r>1$ of subsemigroup $\mathcal{Y}(d,K)$ and computes $ Rev(u) $. Alice has $ h^k Y(d,K) $ together with $ Rev(h) $. Alice forms elements $ u_i $ and $ Rev(u_i)=v_i $ and computes $ \psi(hu_i Rev(h))=v_i' $ for $ i=1,2, ..., r $. She takes $ f $ from $ Y(d,K) $ and forms strings $ Rev(\psi(u_i)) $, $ Rev(f) $. Alice computes $ h^{\psi(u_i) Rev(\psi(u_i))} f $. She takes invertible affine $ j=1,2, ..., t $ transformations $ T $ and $ L $ of free modules $ K^m $ and $ K^n $ of kind and forms pairs $ (a_i: T^i, b_i: L, L^i) $ and sends them to Bob.

He forms word $ w=(a_{i_1})^{\psi(u_1)}(a_{i_2})^{\psi(u_2)}(a_{i_3})^{\psi(u_3)} ... (a_{i_t})^{\psi(u_t)} $, $ t=r-1, i(jk) \{1,2, ..., r\}, a(j_i)>0 $ and sends it to Alice. Bob changes alphabet via the substitution of $ b_i $ instead of $ a_i $ and keeps the reverse word $ \psi(a_{i_1})^{\psi(u_1)}(a_{i_2})^{\psi(u_2)}(a_{i_3})^{\psi(u_3)} ... (a_{i_t})^{\psi(u_t)} $.
Alice computes $u'$ as $L_{\phi}(f) \circ \phi(h)(T^i w T)^{-1} \cdot (T^j) = L_{\phi}(f) \circ \phi(h) \circ (T^i w T)^{-1} \cdot (T^j)$ where $\phi = \beta_{i,j}$ and $\sigma$ homomorphism of $\mu(U(d, K))$ onto $U(d, K)$ induced by graph homomorphism $\beta_{i,j}$. So Alice and Bob when the protocol ends have mutually inverse encryption/decryption tools $u'$ and $u$ for the plaintext $K$.

The algorithm is implemented in the cases of $K = \mathbb{Z}_p$, $p=2^t$ and $K = \mathbb{Z}_p^2$, $p=2^t$ for $d=2$.

4.3. Remarks on complexity.

Let us estimate the complexity of computations for Bob. He need to create two words of finite lengths in corresponding affine Cremona semigroup via several compositions of quadratic polynomials in $n=k^2+2k$ variables. It takes him $O(n^2)$ elementary ring operations. Computation of quadratic map in given point of $K^n$, $n=k^2+2k$ takes time $O(k^n)$. Thus the total complexity of computations for Bob is $O(n^2)$.

Let us estimate the complexity of decryption process for Alice. She need computation of product of linear and quadratic maps, product of two quadratic maps of densities $O(k^2)$ and $O(k^2)$, product of two quadratic maps of densities $O(k^2)$. It requires $O(k^{10})$ operations.

5. On Eulerian semigroups and corresponding inverse protocols

Let $K$ be a finite commutative ring with the multiplicative group $K^*$ of regular elements of the ring. We take Cartesian power $K^n = (K^*)^n$ and consider an Eulerian semigroup $\text{ES}(K)$ of transformations of kind $x_1 \rightarrow M_{x_1} x_1^{a(1,1)} x_2^{a(1,2)} \ldots x_n^{a(n,1)}$, $x_2 \rightarrow M_{x_2} x_1^{a(2,1)} x_2^{a(2,2)} \ldots x_n^{a(n,2)}$, $\ldots$, $x_n \rightarrow M_{x_n} x_1^{a(n,1)} x_2^{a(n,2)} \ldots x_n^{a(n,n)}$, where $a(i,j)$ are elements of arithmetic ring $\mathbb{Z}_d$, $d=|K^*|$, $m \in K^*$.

Let $\text{EG}(K)$ stand for Eulerian group of invertible transformations from $\text{ES}(K)$. It is easy to see that the group of monomial linear transformations $M_*$ is a subgroup of $\text{EG}(K)$. So semigroup $\text{ES}(K)$ is a highly noncommutative algebraic system. Each element from $\text{ES}(K)$ can be considered as transformation of a free module $K^n$.

The problems of constructions of large subgroups $G$ of $\text{EG}(K)$, pairs $(g, g^{-1})$, $g \in G$, and tame Eulerian homomorphisms $\mu : G \rightarrow H$, i.e. computable in polynomial time $t(n)$ homomorphisms of subgroup $G$ of $\text{EG}(K)$ onto $H < \text{EG}(K)$ are motivated by tasks of Nonlinear Cryptography.

Each element of the semigroup $\text{ES}(K)$ is written in the chosen basis $e_1, e_2, \ldots, e_n$.

Let $J = \{i(1), i(2), \ldots, i(k)\}$ be a subset of $\{1, 2, \ldots, n\}$ and $W_j = e_{i(j)}$, $e_{i(j)}$, $\ldots$, $e_{i(k)}$ be a corresponding symplectic subspace. We refer to totality $\text{PS}_j$ of maps $F \circ \text{ES}(K)$ preserving $W_j$ as parabolic semigroup of $\text{ES}(K)$. The map $F$ from $\text{PS}_j$ $(K)$ transforms tuple $(x_{i(1)}, x_{i(2)}, \ldots, x_{i(n)})$ according to the rule $x_{i(1)} \rightarrow M_{x_{i(1)}} x_{i(1)}^{a(1,1)} x_{i(2)}^{a(1,2)} \ldots x_{i(k)}^{a(1,k)}, x_{i(2)} \rightarrow M_{x_{i(2)}} x_{i(2)}^{a(2,1)} x_{i(2)}^{a(2,2)} \ldots x_{i(k)}^{a(2,k)}, \ldots, x_{i(k)} \rightarrow M_{x_{i(k)}} x_{i(k)}^{a(k,1)} x_{i(k)}^{a(k,2)} \ldots x_{i(k)}^{a(k,k)}$.

Let $\pi_j$ be the restriction of element $F$ from $\text{PS}_j$ $(K)$ onto $W_j$. The map $\pi_j$ defines canonical homomorphism of $\text{PS}_j$ $(K)$ onto $\text{ES}(K)$.

6. Conclusion

The usage of stable inverse platforms was discussed in [4]. For instance correspondents can use cubical collision rules keeping in mind attacks by adversary with the interception of plaintext – ciphertext pairs. In the case of plaintext space $K$ adversary has to intercept $O(n^2)$ pairs to conduct successful linearization attack in time $O(n^{10})$. Thus correspondents can follow natural recommendation to start a new session of the inverse protocol after the exchange of $O(n^2)$ messages. Instead of a new protocol Alice can use idea of deformation rule. She can use same platform to generate its element $g$ together with its inverse $g^{-1}$, combine $g$ with two affine bijective maps $T_1$ and $T_2$, use her encryption map $e_A$ already elaborated during the session of inverse protocol and send $e_A(T_1 g T_2)$ or $T_2 g T_2(e_A)$ to Bob. He can restore $T_1 g T_2$ and use it as the new encryption rule. Alice can decrypt because of her knowledge of the inverse map.

We believe that the case of single toric inverse algorithm has similarity with the case of stable protocol. Adversary has to intercept set of pairs plaintext /ciphertext of polynomial cardinality to interpolate encryption function.

Research on finding of exact upper bounds is an interesting task. Other interesting question is about the existence of polynomial algorithm to find the inverse of element $g$ from $\text{ES}(K)$ or $\text{EG}(K)$. Similarly to the problem of finding the inverse of bijective multivariable map a polynomial algorithm to invert $g$ is currently unavailable.
Despite the difference in interpolation of encryption functions, security of both toric and stable inverse protocols rests on the same difficult word decomposition problem for the large semigroup, which is intractable with ordinary Turing machine and Quantum Computer.

The usage of tandem which consists of toric and stable the protocol allows to create “eternal” encryption rule similar to public key but not given publicly. Let us assume that toric and stable protocols of tandem algorithm elaborate pairs of maps (\(e_A, e_{A}\)) and (\(e_B, e_{B}\)) for Alice and Bob. The problem to interpolate composition \(e_A(e_{A})\), which is non-bijective map of \((K^*)^n\) to \(K^n\) of unbounded degree and polynomial density is unfeasible task and decryption function has non polynomial density.

Example of inverse protocols based on toric and stable platforms with outputs acting on \((K^*)^n\) and \(K^n\) gives algorithms 5.3 with arbitrary parameter \(k\) and \(l+|J|=n\) together with algorithm 4.4 with usage graphs \(DS(k', K)\) and \(DS(l',K)\) where \(l'+|J'|=n\) and \(K\) is a finite field or arithmetic ring. Implementation of different from 4.4 stable algorithms is given in [31], [32], [33], alternative to procedure of 5.3 is given in [6].

Notice that in all mentioned above platforms group enveloped inverse Diffie – Hellman protocol [4] can be used instead of inverse protocols 2.1 and 2.2. Recently some new platforms which are formed by families of stable subsemigroups of affine Cremona semigroups have been constructed (see [43], [45], [46]). They can be also used in the combinations with the subsemigroups of Eulerian transformations.

References


