On Symbolic Computations over Arbitrary Commutative Rings via Temporal Jordan-Gauss Graphs and Multivariate Cryptosystems*

Vasyl Ustimenko[1,2,†] and Oleksandr Pustovit[2,*,†]

[1] Royal Holloway University of London, United Kingdom, Egham Hill, Egham TW20 0EX, United Kingdom

[2] Institute of Telecommunications and Global Information Space, NAS of Ukraine, 13 Chokolivsky ave., 02000 Kyiv, Ukraine

Abstract

The paper is dedicated to Multivariate Cryptography over general commutative ring K and protocols of symbolic computations for safe delivery of multivariate maps. We consider the iterative algorithm of generation of multivariate maps of prescribed degree or density with the trapdoor accelerator, i.e. piece of information which allows to compute the reimage of the map in polynomial time. The concept of Jordan-Gauss temporal graphs is used for the obfuscation of known graph based public keys and constructions of new cryptosystems. We suggest use of the platforms of Noncommutative Cryptography defined in terms of Multivariate Cryptography over K for the conversion of Multivariate Public Keys into El Gamal type Cryptosystems. Some new platforms are introduced.

Keywords

public key, multivariate cryptography, general commutative ring, noncommutative cryptography, key exchange protocols, semigroups of transformations

1. Introduction

The paper is dedicated to the constructions of special multivariate maps on affine space K^n over finite commutative ring with the unity. We are interested in maps of prescribed bounded by constant degree or unbounded degree but prescribed density which has a trapdoor accelerator, i.e. pieces of information such that its knowledge allows us to compute the reimage of the map in polynomial time.

One of the applications of these maps is the following scheme of access control to the resources of Information System. Administrator A of the Information System (IS) possesses the map F in n-variables and its trapdoor accelerator T. He/she is going to give secure access to the resources of IS to trusted user U. So A and U executes selected protocol of Noncommutative Cryptography in terms of special subsemigroup S of the affine Cremona semigroup of all multivariate maps of K^n into itself. The output of the protocol X can be used by A and U for the creation of its *deformation* $G(X)$ which is a transformation of K^n.

The administrator sends $F+G(X)$ to U. User restores F. Now A is able to create pseudorandom or genuinely random passwords $(p_1, p_2, \ldots, p_n) = p$ as the condition to enter the system. The administrator solves the equation $F(x) = b$ and sends the solution $x = (d_1, d_2, \ldots, d_n)=d$ to the user together with the link for entering the password. User U gets the password as $F(d_1, d_2, \ldots, d_n)$.

The administrator has the option to change the password several times working with the same map F with the trapdoor accelerator. He/she is able to change F via a new session of the protocol and delivery scheme.

The security of this scheme rests on the security of selected Postquantum Protocol on Noncommutative Cryptography. We describe Twisted Diffie-Helman protocol which use the complexity of Conjugation Power Problem of the semigroup ${}^nES(K)$ of Eulerian endomorphisms of

* CPITS 2025: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, February 28, 2025, Kyiv, Ukraine

[*] Corresponding author.

[†] These authors contributed equally.

Vasyl.Ustymenko@rhul.ac.uk (V. Ustimenko); sanyk_set@ukr.net (O. Pustovt)

0000-0002-2138-2357 (V. Ustimenko); 0000-0002-3232-1787 (O. Pustovit)

$K[x_1, x_2, ..., x_n]$ which sends each variable x_i, $i = 1, 2, ..., n$ to a monomial term. Some other protocols of Noncommutative Cryptography with the platform ${}^nES(K)$ are given in [1].

For each positive integer d, $d≥2$ we present the multivariate map of degree d with the trapdoor accelerator. In fact we present the iterative process of expansion of initial map $\boldsymbol{F_0}$ which can be a bijective multivariate nonlinear map of degree at most d on K^n with the trapdoor accelerator T or an element of general affine group $AGL_n(K)$. The input parameters are positive integers $m(1)$, $m(2)$, ..., $m(k)$, $k≥2$. The step i, $i = 1, 2, ..., k$ of the algorithm produces the multivariate map G_i of degree d on the $K^{n+m(1)+m(2)+...+m(i)}$ with the trapdoor accelerator T_i.

Similarly we can take polynomial surjective map F_0 of K^n onto K^r of degree at most d with the trapdoor accelerator T and get the sequence of surjective polynomial multivariate maps of $K^{n+m(1)+m(2)+...+m(i)}$ onto $K^{r+m(1)+m(2)+...+m(i)}$ of degree d with the trapdoor accelerators.

So we can use known construction of multivariate cryptography over the general maps with trapdoor accelerators or linear maps on affine spaces for the construction of new maps together with the polynomial algorithm to compute reimage.

We define the density of the multivariate polynomial in n variables as the number of its monomial terms. The density of multivariate map $F: (x_1, x_2, ..., x_n) \longrightarrow (f_1(x_1, x_2, ..., x_n), f_2(x_1, x_2, ..., x_n), ..., f_2(x_1, x_2, ..., x_n), ..., f_m(x_1, x_2, ..., x_n))$ is the maximal value of densities of f_i for $i = 1, 2, ..., m$.

We also will work with the multivariate maps in n variable of unbounded degree and prescribed density $O(n^\lambda)$. Let K^* stands for the multiplicative group of K. Assume that K^* is nontrivial. We say that multivariate map F of K^n to itself has multiplicative trapdoor accelerator T if the restriction of F onto $(K^*)^n$ is injective map and the knowledge of T allows to compute the reimage of the element from $F((K^*)^n)$ in a polynomial time.

For each nonnegative rational number λ we present the explicit constructions of multivariate maps of density λ with unbounded degree and multiplicative trapdoor accelerator. Additionally we present the iterative process of the expansion of the selected initial map F_0 which is a multivariate nonlinear map of density $O(1)$ on K^n with unbounded degree and the multiplicative trapdoor accelerator T. The input consists of positive integers $m(1)$, $m(2)$, ..., $m(k)$, $k≥2$ and some internal parameters which are nonnegative rational numbers.

The step i, $i = 1, 2, ..., k$ of the algorithm produces the multivariate map G_i of polynomial density on the $K^{n+m(1)+m(2)+...+m(i)}$ with the multiplicative trapdoor accelerator T_i. Appropriate choice of internal parameters allows us to construct G_k of prescribed density $O((n+m(1)+m(2)+...+m(k))^\lambda)$.

We can use multivariate maps of unbounded degree and prescribed polynomial density with the multiplicative trapdoor accelerator instead of maps of bounded degree in the presented above scheme of access control. We can use the same protocol of Noncommutative Cryptography and the same platform ${}^nES(K)$ of Eulerian transformations. The modification of the deformation rule will be presented.

Let us consider the case of finite commutative ring K of the cardinality $O(1)$ with nontrivial multiplicative group. In the case of the map F of unbounded by constant degree of size $O(n)$ and of density $O(1)$ with the multiplicative trapdoor accelerator we use term pseudolinear map. The complexity of computation of $F(p)$, $p\epsilon(K^*)^n$ is $O(n^2)$. In the case of density $O(n^\lambda)$, $\lambda<1$ we use the term of sub quadratic map. The complexity of computation of $F(p)$, $p\epsilon\ (K^*)^n$ is $O(n^{2+\lambda})$.

It is better then in the case of quadratic map on the space K^n. If density is $O(n)$ we say that we have pseudo quadratic map.

We hope that defined in the paper wide variety of the quadratic or cubic maps with the trapdoor accelerators and the varieties of pseudo-linear, sub quadratic and pseudo quadratic maps with the multiplicative trapdoor accelerators can be effectively used in the presented above scheme of the access control of Information System.

These varieties are defined via the symbolic computations in terms of algebraic graphs defined by the systems of nonlinear algebraic equations over the finite commutative ring K with unity or temporal analogue of these graphs for which generic equations are changeable with the change of time. The sequences of pseudorandom or genuinely random graphs can be used for the change of coefficients in time dependent algebraic equations.

For the design of maps we use Jordan-Gauss graphs which are bipartite graph with partition sets K^n and K^m given via quadratic equations such that the neighbourhood of the vertex is the solution set of linear system of equations written in its row-echelon form.

Subsection 2.1 of Section 2 contains basic definitions of affine Cremona semigroup and group of endomorphism of multivariate ring $K[x_1, x_2, ..., x_n]$, endomorphisms with the trapdoor accelerators. It contains the discussion of the area of Multivariate Cryptography over the general finite commutative ring.

In the subsection 2. 2 we define linguistic graphs over the general commutative ring and their temporal analogue. Algorithm 1.2 allows us to construct the variety of elements of Cremona semigroup with the trapdoor accelerator defined in terms of selected linguistic graph or its temporal analogue. Simple conditions insure that the constructive map is bijective transformation of K^n. The method allows us to construct surjective maps of K^n onto K^m, $n>m\geq 2$ with the trapdoor accelerator. For practical implementation of the algorithm we need select special classes of linguistic graphs which allow us to control the degrees and densities of the outputs. We define the special class of Jordan-Gauss graphs and consider flexible families of generalised Double Schubert graphs $DS_{s,r}(K)$ *and truncated Double Schubert graphs* ${}^{Q}DS_{s,r}(K)$ which are convenient instruments for generating of families of multivariate maps of prescribed degree on the affine space K^n.

Assume that (F, T) stands for pair multivariate function F of degree d, $d\geq 2$ on K^n and its trapdoor accelerator. We suggest the method of construction of new pair (F', T') of degree d on $K^{n'}$, $n'>n$ from the known (F, T). It can be used iteratively. Many constructions of pairs (F, T) over fields can be found in the recent papers on Classical Multivariate Cryptography [2–13].

In Section 3 we introduce semigroup of ${}^{n}ES(K)$ of Eulerian endomorphisms of $K[x_1, x_2, ..., x_n]$ and consider iterative method of construction of multivariate maps of prescribed density $O(n^d)$ with the trapdoor accelerators or multiplicative trapdoor accelerators. These maps are constructed in terms of temporal truncated Schubert graphs.

In Section 4 we consider twisted Diffie-Hellman protocol implemented with the platform ${}^{n}ES(K)$ of Eulerian transformations. We introduce several *deformation rules* convenient for the safe delivery of multivariate maps of prescribed degree or density from one correspondent to his/her partner. We discuss the use of stable subsemigroups of Cremona semigroup ${}^{n}CS(K)$ as a platform for the protocol. Stability means that the maximal degree of endomorphisms from the semigroup is a constant d.

Section 5 contains conclusive remarks. We have to note that last talk at Eurocrypt conferences was delivered in 2021. It is paper [14] dedicated to cryptanalytic studies. Some studies on Multivariate Cryptography were presented during PQCrypt workshops [15–18].

2. Analysis of the last research and publications

2.1. General remarks

Let K be a finite commutative ring. It is possible to say that Multivariate Cryptography in a wide sense is about the use of polynomial maps F of affine spaces K^n to itself for cryptographical purposes.

In classical case $K=F_q$ the map F is an element of affine Cremona semigroup ${}^{n}CS(K)$ of endomorphisms of multivariate ring $K[x_1, x_2, ..., x_n]$. Endomorphism F can be given by its values $F(x_1) = f_1$, $F(x_2)=f_2$, ..., $F(x_n)=f_n$ on the variables x_i, $i = 1, 2,..., n$.

We can assume that polynomials f_i are given in their standard form i.e. sum of monomial terms ordered in lexicographical order.

Endomorphism F induces the map F': $x_1 \longrightarrow f_1(x_1, x_2, ..., x_n)$, $x_2 \longrightarrow f_1(x_1, x_2, ..., x_n)$, ..., $x_n \longrightarrow f_n(x_1, x_2, ..., x_n)$ of the affine space K^n into itself.

We define degree $deg(F)$ as maximal value of $deg\ (f_i)$. The density $den\ f_i(x_1, x_2, ..., x_n)$ is its number of monomial terms. We define density $den(F)$ of F as maximal value of $den\ (f_i)$, $i = 1, 2, ..., n$ and identify endomorphism F with the tuple $(f_1(x_1, x_2, ..., x_n), f_2(x_1, x_2, ..., x_n), ..., f_m(x_1, x_2, ..., x_n))$.

The image *Im F'* is isomorphic to K^m for some *m, n≥m.* We can treat *F'* as surjective map of K^n onto K^m.

We say that piece of information *T* is *trapdoor accelerator* of surjective nonlinear polynomial map *F'* of K^n onto K^m, $n \geq m$ if the knowledge of ***T*** allows to compute a reimage of given element $b \epsilon K^m$ in a polynomial time.

New multivariate cryptosystem "TUOV: Triangular Unbalanced Oil and Vinegar" was officially submitted to NIST recently see https://csrc.nist.gov/csrc/media/Projects/pqc-digsig/documents /round-1/spec-files/TUOV-spec-web.pdf). It is based on the quadratic map defined over finite fields with the trapdoor accelerator.

He hopes that this is the example of one way function, i.e. the reimage of this quadratic map is not possible to compute in a polynomial time without the knowledge of given trapdoor accelerator.

As you know the existence of one way function is not proven. Anyway there is a chance of NIST certification of TOUV as first representative from the class of Multivariate Public Keys.

As you know Multivariate cryptography uses the *gap between linearity and nonlinearity.* We know that the system of linear equations written over the field *F* can be solved in time $O(n^3)$ via Jordan-Gauss elimination method.

The complexity of solving a nonlinear system of constant degree *d, d>1* is subexponential.

Despite the convenience of Groebner basis method for the implementation the complexity of this algorithm is equivalent to old Gauss elimination method for solution of the system of nonlinear equation.

Recall that the standard way to transform of nonlinear system of equation of degree *d, d>2* to an equivalent quadratic system via introduction of additional variables and substitutions is well known [19].

So if we have a nonlinear map *F* of bounded degree *d* in "general position" which has a trapdoor accelerator *T* then corresponding cryptosystem is secure. This status is insure the fact that *F* is given as one way function i.e. reimage of *F* is impossible to compute in a polynomial time without knowledge of the secret *T.*

The map *F* is not in "*general position"* if some additional specific information is known. For instance, if *F* is bijective cubic map and F^{-1} is also cubic. Then public user can generate $O(n^3)$ pairs of kind plaintext *p*/corresponding ciphertext *c* and approximate inverse map in time $O(n^{10})$.

Known computer tests and cryptanalytic methods are attempts to justify that map *F* is "in general position". Noteworthy that the existence of one way function is not proven yet even under the *main complexity conjecture* that *P≠NP.*

Note that the investigation of nonlinear systems of equations over the commutative ring *K* with *zero divisors* is essentially harder case in comparison the case of a field.

Multivariate Cryptography over rings with zero divisors can be an interesting direction of cryptographic research.

2.2. Linguistic graphs and multivariate maps over commutative rings

Below we present the method of construction of nonlinear representatives of affine Cremona semigroup $End\ K[x_1, x_2,..., x_n]$ where *K* is a finite commutative ring.

The incidence structure is the set *V* with the 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 which is of course a bipartite graph. The pair *x, y, x ϵ P, yϵ L* such that *x I y* is called a flag of incidence structure *I.*

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

$$
\begin{aligned}
&a_1x_{s+1}-b_1y_{r+1} = f_1(x_1, x_2, \ldots, x_s, y_1, y_2, \ldots, y_r),\\
&a_2x_{s+2}-b_2y_{r+2} = f_2(x_1, x_2, \ldots, x_s, x_{s+1}, x_{s+1}, y_1, y_2, \ldots, y_r, y_{r+1}),\\
&\ldots \\
&a_mx_{s+m}-b_my_{r+m} = f_m(x_1,x_2,\ldots ,x_s, x_{s+1}, \ldots, x_{s+m-1}, y_1, y_2, \ldots, y_r, y_{r+1}, \ldots,y_{r+m-1})
\end{aligned}
\qquad (1)
$$

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 [20, 21]. 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 the incidence graph there exists a 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 $b \epsilon K^r$ and $p = (p_1, p_2, \ldots,p_{s+m})$ there is a unique neighbour of the point $[l] = N_b(p)$ with the colour b. Similarly for each $c \epsilon K^s$ and line $l = [l_1, l_2, \ldots, l_{r+m}]$ there is a unique neighbour of the line $(p) = N_c([l])$ with the colour c. The triples of parameters s, r, m defines *type of linguistic graph.*

Let $J_a(v)$ stands for the operator of change colour of vertex v (point or line) for $a = (a_1, a_2, \ldots, a_t)$ where $t=s$ or $t=r$.

We consider also linguistic incidence structures defined by infinite number of equations. Let $I(K)$ and $I'(K')$ be two linguistic graphs of the same type (s, r, m) with governing polynomials f_i and f'_i written in their standard forms. We refer to them as symbolically equivalent structures if monomial terms of f_i and f'_i for each i are the same up to their nonzero coefficients.

We refer to family $I(K)^t$, $t = 1, 2, \ldots$ of symbolically equivalent linguistic graphs as *temporal linguistic graph.*

Algorithm 1.2. (Generation of multivariate map F with the trapdoor accelerator [22])

Let us consider linguistic graph ${}^mI_{s,r}(K)$ given by equations (1) of type $s, r, m, s \geq r$ together with graph ${}^mI_{s,r}(R)$ where R is the commutative ring of multivariate polynomials $K[z_1, z_2, \ldots, z_s, z_{s+1}, z_{s+2}, \ldots, z_{s+m}]$ given by the same equations (1) with coefficients from K but with variables x_i, y_j from R. So infinite graph ${}^mI_{s,r}(R)$ has the point set R^{s+m} and the line set R^{r+m}.

Let us conduct the following symbolic computation. We consider the special point $z = (z) = (z_1, z_2, \ldots, z_s, z_{s+1}, z_{s+2}, \ldots, z_{s+m})$ which coordinates are variables, positive integer l and colours $a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l)$ and c such that $a(1), a(3), \ldots, a(l), b(2), b(4), \ldots, b(l-1) \epsilon K[z_1, z_2, \ldots, z_s]^s$, elements $a(2), a(4), \ldots, a(l-1), b(1), b(3), \ldots, b(l) \epsilon K[z_1, z_2, \ldots, x_s]^r$.

So, we compute recurrently $v_1 = J_{a(1)}(z)$, $u_1 = N_{b(1)}(v_1)$, $v_2 = J_{a(2)}(u_1)$, $u_2 = N_{b(2)}(v_2)$, $\ldots$, $v_l = J_{a(l)}(u_{l-1})$, $u_l = N_{b(l)}(v_l)$ and finally $J_c(u_l) = v$. If l is odd then $v = (f_1, f_2, \ldots, f_r, f_{1+r}, f_{2+r}, \ldots, f_{m+r})$. Thus we construct the map $F = F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c)$ from K^{s+m} to K^{r+m} sending the tuple $(z_1, z_2, \ldots, z_s, x_{s+1}, x_{s+2}, \ldots, x_{s+m})$ to $(f_1, f_2, \ldots, f_r, f_{r+1}, f_{r+2}, \ldots, f_{r+m})$. In the case of even k we construct the transformation $F = F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c)$ of K^{s+m} given by the tuple $(f_1, f_2, \ldots, f_s, f_{1+s}, f_{2+s}, \ldots, f_{m+s})$.

Note that f_i, $i = 1, 2, \ldots, s$ are elements of $K[z_1, z_2, \ldots, z_s]$ but $f_i \epsilon K[z_1, z_2, \ldots, z_s, z_{1+s}, z_{2+s}, \ldots, z_{m+s}]$.

Assume that map L_1 is an element of $AGL_{s+m}(K)$ and L_2 is taken from $AGL_{r+m}(K)$ in the case of odd l and $L_2 \epsilon AGL_{s+m}(K)$ if l is even. The bijective polynomial maps L_1 and L_2 have degree 1. Then we can compute the standard form of the map $G=L_1FL_2$.

Proposition 1. 2. [22] *Assume that constant l is odd the tuple c defines surjective multivariate map C from K^s to K^r with trapdoor accelerator T and parameters $a(i)$, $b(i)$ and c have degrees of size $O(1)$. Then polynomial surjective map G* from *K^{s+m} to K^{r+m} has the trapdoor accelerator T' which is the knowledge on l, $a(i)$, $b(i)$, $i = 1, 2, \ldots, l$, C, T, L_1, L_2 and equations (1).*

Remark 1.2. If $K = F_q$ we can take the pair C, T defined by J. Ding and his team and get a new surjective map G from larger vector space with the trapdoor accelerator.

Proposition 2.2. [22] *Assume that l is even or $r = s$ and the tuple c defines bijective multivariate map C from K^s to K^s with trapdoor accelerator T. Assume that $a(i)$, $b(i)$, c are of size $O(1)$. Then the map G is bijective, it has trapdoor accelerator T' which is the knowledge on l, $a(i)$, $b(i)$, $i =1, 2, \ldots, l$, C, T, L_1, L_2 and equations (1).*

Remark 2.2. Under the condition of Proposition 2 in the case of even **l** it could be that $r>s$.

Procedure 1.2 (reimage computation).

Alice gets the image $\boldsymbol{e} = (e_1, e_2, \ldots, e_t, e_{t+1}, e_{t+2}, \ldots, e_{t+m})$, $t = r$ or $t = s$ of the map G. She creates intermediate vector $(z_1, z_2, .., z_s, z_{s+1}, z_{s+2}, \ldots, z_{s+m})$. Alice computes $(L_2)^{-1}(e) = (d_1, d_2, \ldots, d_t, d_{t+1}, d_{t+2}, \ldots, d_{s+m}) = d$. She investigates the system of equations $c_1(z_1, z_2, \ldots, z_s) = d_1$, $c_2(z_1, z_2, \ldots, z_s) = d_2$, …, $c_t(z_1, z_2, \ldots, z_s) = d_t$. The knowledge of T allows her to take some solution $z_1 = \alpha_1$, $z_2 = \alpha_2$, …, $z_s = \alpha_s$. Alice calculates values $\beta(i) = b(i)(\alpha_1, \alpha_2, \ldots, \alpha_s)$, $\gamma(i) = a(i)(\alpha_1, \alpha_2, \ldots, \alpha_s)$, $i = 1, 2, \ldots, l$.

She computes $J_{\beta(l)}(d) = v_l$, $N_{a(l)}(v_l) = u_l$, $J_{\beta(l-1)}(u_l) = \mathrm{v}_{l-1}$, $N_{a(l-1)}(v_{l-1}) = u_{l-1}$, …, $J_{\beta(1)}(u_2) = v_1$, $N_{a(l)}(v_1) = u_1$, $J_a(u_1) = u$ for $a = (\alpha_1, \alpha_2, \ldots, \alpha_s)$.

Alice computes the reimage as $(L_1)^{-1}(u)$.

Remark. 3.2. We can define $F = F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c) = F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c, I_1, I_2, \ldots, I_l)$ *in the case of temporal linguistic graph* ${}^mI_{s,r}(K)^t$ via simple assumption that operators $N_{b(j)}$ *of the algorithm* are executed in the graph I_j $(K[z_1, z_2, \ldots, z_s, z_{s+1}, z_{s+2}, \ldots, z_{s+m}])$ formed as expansion of *momentum* graph $I_j = {}^mI_{s,r}(K)^j/t = j$, $j = 1, 2, \ldots, l$. Proposition 1.1 and 2.2 hold for temporal graphs as well.

To control the degrees and densities of $F = F(a_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l, c)$ we need a special class of linguistic graphs over K.

Jordan-Gauss graphs are linguistic graphs given by special quadratic equations over the commutative ring K with unity such that the neighbour of each vertex is defined by the system of linear equation given in its row-echelon form [23–25].

Generalised Double Schubert graph $DS_{s,r}(K)$ (see [22] and [26] and further references) is a bipartite graph with the points of kind $(x) = (x_1, x_2, \ldots, x_s, x_{11}, x_{12}, \ldots, x_{sr})$ and lines $[y] = [y_1, y_2, \ldots, y_r, y_{11}, y_{12}, \ldots y_{st}]$ such that point (x) is incident to $[y]$ if and if the conditions

$$x_{ij}\text{-}y_{ij} = x_iy_j \tag{2}$$

hold for $i = 1, 2, \ldots, s$ and $j = 1, 2, \ldots, r$.

Temporal graph $DS_{s,r}(K)^t$ is given by equations

$${}^{i,j}\alpha(t)x_{ij}\text{-}{}^{i,j}\beta(t)y_{ij} = {}^{i,j}\gamma(t)x_iy_j \tag{2′}$$

where ${}^{i,j}\alpha(t)$ and ${}^{i,j}\beta(t)$ are elements of multiplicative group K^* and ${}^{i,j}\gamma(t)$ are elements of $K\text{-}\{0\}$.

To form momentum graphs $D_1 = DS_{s,r}(K)^t/t = 1$, $D_2 = DS_{s,r}(K)^t/t = 2$, … we can use pseudorandom or random sequences of elements from K^* or $K\text{-}\{0\}$ respectively. For the constructions genuinely random sequences Quantum Computer can be used.

Remark 4.2. Graph $DS_{s,r}(K)$, $K = F_q$ is formed by spaces of dimension s and $s+1$ from two corresponding largest Schubert cells of projective geometry $PG_{s+r}(F_q)$.

In fact many other temporal Jordan-Gauss graphs and their configurations the reader can find in [27]. These constructions are defined in terms of theory of Lie Geometries and their generatisations [28–30].

Proposition 3.2. [22] *Let us consider map introduced above map $G = L_1F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c, D_1, D_2, \ldots, D_l)L_2$ in the case of the temporal graph $DS_{s,r}(K)^t$. Assume that $\deg a(i)+\deg b(i) \le d$, $\deg c = d$.* Then degree of $G = G(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c)$ *is d.*

In the case of $d = 2, 3$ we can use this construction to obfuscate selected multivariate cryptosystem C, T. In particular we can take as C, T already mentioned quadratic cryptosystem TUOV (Triangular Unbalanced Oil and Vinegar cryptosystem). We can also introduce enveloping trapdoor accelerator for Matsumoto-Imai cryptosystem over finite fields of characteristic 2, for the Oil and Vinegar public keys over F_q. Another quadratic multivariate public keys defined over Jordan-Gauss graphs $D(n, K)$, where K is arbitrary finite commutative ring with the nontrivial multiplicative group. It gives us the option to use Proposition 3.2 in the case of arbitrary commutative ring K [24, 31]. We can obfuscate presented above constructions of multivariate maps of degree d with the trapdoor accelerator T below via deleting of some coordinates of points and lines with double indexes together with corresponding equations. It will give us examples of multivariate maps of prescribed degree with the trapdoor accelerator on arbitrary free module K^n. Instead of generalised Schubert graph $DS_{s,r}(K)$ with points of kind $(x) = (x_1, x_2, \ldots, x_s, x_{11}, x_{12}, \ldots, x_{sr})$ and lines $[y] = [y_1, y_2, \ldots, y_r, y_{11}, y_{12}, \ldots, y_{st}]$ we consider homomorphic image ${}^QDS_{s,r}(K)$ where Q is selected proper subset of Cartesian product of $\{1,2, \ldots, s\} = N$ and $[1, 2, \ldots, r\} = M$. We assume that $r = O(s)$, the projection $(i, j) \longrightarrow i$ maps Q onto N and the projection $(i, j) \longrightarrow j$ maps Q onto M. Let $Q = \{\alpha(1), \alpha(2), \ldots, \alpha(m)\}$ *where* $m = O(s^t)$, $1 \leq t \leq 2$. Then partition sets of ${}^QDS_{s,r}(K)$ are affine space K^{s+m} and K^{r+m}. We consider the map ${}^QF = {}^QF(a_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l, c)$ obtained in the case of linguistic graph ${}^QDS_{s,r}(K)$. We also consider ${}^QG$ as $L_1{}^QFL_2$ where L_1 and L_2 are bijective affine transformations of partition sets of ${}^QDS_{n,k}(K)$. We refer to graphs ${}^QDS_{s,r}(K)$ as *Truncated Schubert Graphs* and consider their temporal analogous ${}^QDS_{s,r}(K)^t$ introduced via the deletion of coordinates indexed by elements of $N \cdot M\text{-}Q$ and corresponding equations from the system (2′).

Let $D_1 = DS_{s,r}(K)^t/t = 1, D_2 = DS_{s,r}(K)^t/t = 2, \ldots$ stands for the *momentum graphs* of ${}^QDS_{s,r}(K)^t$.

Proposition 3'.2. [22] *Let us consider map introduced above map* $G = L_1F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c, D_1, D_2, \ldots, D_l)L_2$ *in the case of the temporal graph* ${}^QDS\ _{s,r}(K)^t$. *Assume that* $\deg a(i) + \deg b(i) \leq d$, $\deg c = d$. Then degree of $G = G(a(_1, a_2, \ldots, a_l, b_1, b_2, \ldots, b_l, c, D_1, D_2, \ldots, D_l)$ *is* d.

Corollary. *Formulated above proposition allows us to construct multivariate bijective map G of prescribed degree d, d≥2 with the trapdoor accelerator on arbitrary affine space* K^n.

We can use the construction of Proposition 3' iteratively.

Example 1.2. Let us select finite commutative ring K and positive numbers s, $m(1)$, $m(2)$, ... to generate the sequence of bijective maps of prescribed degree d on $K^{s+m(1)}$, $K^{s+m(1)+m(2)}$, ... with the trapdoor accelerators.

1 step. We use Proposition 3'.2 in the case of selected d, temporal Jordan-Gauss graph of type s, r, $s+m(1)$ where $s+m(1) \leq sr$, $l = l(1)$ is even, tuples $a(1) = a(1, i)$, $b(1) = b(1, i)$ satisfy the condition of the statement and $c = (c_1, c_2, \ldots, c_s)$ has degree 1 and the map C of kind $z_i \longrightarrow c_i(z_1, z_2, \ldots, z_s)$, $i = 1, 2, \ldots, l$ *is* an element of $AGL_s(K)$. Let the standard form G_1 from ${}^{s+m(1)}CG(K)$ with the corresponding trapdoor accelerator T_1 be the output of the procedure.

2 step and iteration. We use Proposition 3'.2 in the case of Jordan graph of type $s+m(1)$, $r(1)$, $s+m(1)+m(2)$ where $s+m(1)+m(2) \leq (s+m(1))r(1)$, $l = l(2)$ is even, $a(i)$ and $b(i)$ satisfy the condition of the statement and c coincides with the tuple $g(1) = (G_1(z_1), G_1(z_2), \ldots, G_1(z_{s+m(1)}))$. Let the standard form of G_2 and its trapdoor accelerator T_2 be the output of Step 2. Notice that the piece of information T_2 is an expansion of T_1.

We use the tuple $c = g(2) = (G_2(z_1), G_2(z_2), \ldots, G_2(z_{s+m(1)+m(2)}))$ and Proposition 3' to generate the transformation G_3 of affine space $K^{s+m(1)+m(2)+m(3)}$ with the trapdoor accelerator T_3 expanding T_2. If we use k as total number of steps, then the continuation of this recurrent procedure of generating tuples $g(3)$, $g(4)$, ..., $g(k\text{-}1)$ via free selection of even parameters $l(3)$, $l(4)$, ..., $l(k)$ gives the transformation $G_{k_}$of degree d on the affine space of dimension $s+m(1)+m(2)+\ldots+m(k)$ together with the trapdoor accelerator T_k.

Procedure 2.2 (reimage computation for (G_k, T_k)).

Assume that $G_j = {}^jL_1F_j\,{}^jL_2$, $j = 1,2, \ldots, k$ and $F_j = F(a(1, j), a(2, j), \ldots, a(l(j), j), b(1, j), b(2, j), \ldots, b(l(j),j), g(j\text{-}1), {}^jD_1, {}^jD_2, \ldots, {}^jD_{l(j)})$ acting on the affine space ${}^jW$ of dimension $s+m(1)+m(2)+\ldots+m(j) = n(j)$.

Alice obtained the ciphertext ${}^0c = ({}^0c_1, {}^0c_2, \ldots, {}^0c_{n(k)})$. She computes ${}^kL_2^{-1}({}^0c) = {}^kc$ and takes its projection ${}^kc'$ on the first $n(k\text{-}1)$ coordinates.

Alice computes ${}^{k-1}L_2^{-1}({}^kc') = {}^{k-1}c$ and takes its projection ${}^{k-1}c'$ on first $n(k\text{-}2)$ coordinates. She continue this procedure and gets the tuple ${}^1c = (b_1, b_2, \ldots, b_s, b_{s+1}, b_{s+2}, \ldots, b_{s+m(1)})$ and ${}^1c' = (b_1, b_2, \ldots, b_s)$.

Alice forms the intermediate tuple $(z_1, z_2, \ldots, z_s)$ and investigates the system of linear equations $\mathbf{c_1}(z_1, z_2, \ldots, z_s) = b_1$, $c_2(z_1, z_2, \ldots, z_s) = b_2$, ..., $c_s(z_1, z_2, \ldots, z_s) = b_s$. She gets the solution $z_1 = \alpha_1$, $z_2 = \alpha_2$, ..., $z_s = \alpha_s$. Alice computes tuples $a^*(i, 1) = a(1, 1)(\alpha_1, \alpha_2, \ldots, \alpha_s)$, $b^*(i, 1) = b(1, 1)(\alpha_1, \alpha_2, \ldots, \alpha_s)$, $i = 1, 2, \ldots, l(1)$ with coordinates from K.

Alice takes graph ${}^1D_{l(1)}$ and computes $d(l(1)) = \mathcal{J}_{b^*(l(1),1)}({}^1c)$. She takes the neighbour $d'(l(1) = N_{a^*(l(1)),1)}$ $(d(l(1))$ of the point $d(l(1)$ of colour $a^*(l(1), 1)$. Alice treats the tuple $d'(l(1))$ as the line of graph ${}^1D_{l(1)}$. She computes $\mathcal{J}_{b^*(l(1)-1),1)}$ $(d'(l(1)) = d(l(1)-1)$ and its neighbour $d'(l(1)-1) = N_{a^*(l(1)-1),1)}$ $(d(l(1)-1)$. Alice continue this process and gets $d'(1) = N_{a^*(1,1)}\,(d(1))$ in the graph ${}^1D_1$. So she gets $e(1) = \mathcal{J}_\gamma(d'(1))$, $\gamma = (\alpha_1, \alpha_2, \dots, \alpha_s)$.

The tuple $({}^1L_1)^{-1}(e(1)) = r(1)$ is the solution of the equation $L_1F_1(z_1, z_2, z_s, z_{s+1}, \dots, z_{s+m(1)}) = {}^1c{=}{}^1(L_2)^{-1}({}^2c')$ which is equivalent to $G_1(z_1, z_2, z_s, z_{s+1}, \dots, z_{s+m(1)}) = {}^2c'$.

Alice considers the equation ${}^2L_1F_2(z_1, z_2, \dots, z_s, z_{s+1}, \dots, z_{s+m(1)}, z_{s+m(1)+1}, \dots, Z_{s+m(1)+m(2)}) = {}^2c = {}^2L_2({}^3c')$. The first $s+m(1)$ equations of this system are equivalent to $L_1F_1(z_1, z_2, \dots, z_{s+m(1)}) = {}^1c$ with the solution $\gamma(1) = ({}^1\alpha_1, {}^1\alpha_2, \dots, {}^1\alpha_{s+m(1)})$.

Alice computes the specializations $a^*(1, 2), a^*(2, 2), \dots, a^*(l(2), 2), b^*(1, 2), b^*(2, 2), \dots, b^*(l(2), 2)$ of $a(1, 2), a(2, 2), \dots, a(l(2), 2), b(1, 2), b(2, 2), \dots, b(l(2), 2)$ under the substitution $z_1 = {}^1\alpha_1$, $z_2 = {}^1\alpha_2, \dots, z_{s+m(1)} = {}^1\alpha_{s+m(1)}$.

She computes the point $d(l(2)) = \mathcal{J}_{b^*(2,\ l(2))}({}^2c)$ and line $d'(l(2)) = N_{a^*(2,\ l(2))}(d(l(2)))$ of the graph ${}^2D_{l(2),}$ computes $d(l(2)-1) = \mathcal{J}_{b^*(2,\ l(2)-1)}(d'(l(2))$ and vertex $d'(l(2)-1) = N_{a^*(2,\ l(2)-1)}(d(l(2)-1))$ of the graph ${}^2\mathbf{D}_{\mathbf{l(2)-1}}$. Alice continue this process and gets $d'(1) = N_{a^*(2,1)}\,(d(1))$ in the graph ${}^2D_1$. So she gets $e(1) = \mathcal{J}_{\gamma(1)}(d'(1))$ in this graph.

The tuple $({}^2L_1)^{-1}(e(1)) = \gamma(2)$ is the solution of the equation ${}^2L_1F_2(z_1, z_2, z_s, z_{s+1}, \dots, z_{s+m(1)}, z_{s+m(1)+1}, \dots,$ zs+m(1)+m(2)) = 2c = 1(L2)-1(3c') which is equivalent to G2(z1, z2, zs, zs+1, ..., zs+m(1)+m(2)) = 3c'.

Alice continue this recurrent process and gets the solution $\gamma(k)$ of the equation $G_k(z_1, z_2, z_s, z_{s+1}, \dots, z_{s+m(1)+m(2)+\dots+m(k)}) = {}^0c$.

Example 2.2. Let us select finite commutative ring K and positive numbers $s, r, s{\geq}r, m(1), m(2), \dots$ to generate the sequence of bijective maps of prescribed degree d from $K^{s+m(1)}$ onto $K^{r+m(1)}$, from $K^{s+m(1)+m(2)}$ onto $K^{r+m(1)+m(2)}$, ... with the trapdoor accelerators. We will use Proposition 3' several times in the case of odd parameter l.

1 step. We use Proposition 3' in the case of selected d, temporal Jordan-Gauss graph of type $s, r, m(1)$ where $s{\leq}m(1){\leq}\ sr$, $l = l(1)$ is odd, tuples $a(i)$, $b(i)$ satisfy the condition of the statement and $c = (c_1, c_2, \dots, c_s)$ has degree 1 and the map C: $(z_1, z_2, \dots, z_s) \longrightarrow (c_1(z_1, z_2, \dots, z_s), c_2(z_1, z_2, \dots, z_s), \dots, c_r(z_1, z_2, \dots, z_s)$ is surjective. We can assume that linear expressions $c_1, c_2, \dots, c_r$ are written in a row echelon form.

Let the standard form the map G_1 from $K^{s+m(1}$onto $K^{r+m(1))}$ with the corresponding trapdoor accelerator T_1 be the output of this step.

2 step and iteration. We use Proposition 3' in the case of Jordan graph of type $s+m(1), r(1)+m(1), m(1)+m(2)$ where $s+m(1)+m(2){\leq}(s+m(1))(r(1)+m(1)$, $l = l(2)$ is odd, $a(i)$ and $b(i)$ satisfy the condition of the statement and $\boldsymbol{c}$ coincides with the tuple $g(1) = (G_1(z_1), G_1(z_2), \dots, G_1(z_{r+m(1)}))$. Let the standard form of G_2 and its trapdoor accelerator T_2 be the output of Step 2. Notice that the piece of information T_2 is an expansion of T_1.

We use the tuple $c = g(2) = (G_2(z_1), G_2(z_2), \dots, G_2(z_{r+m(1)+m(2)}))$ and Proposition 3' to generate the map G_3 of affine space $K^{s+m(1)+m(2)+m(3)}$ *onto* $K^{r+m(1)+m(2)+m(3)}$ with the trapdoor accelerator T_3 expanding T_2. If we use k as total number of steps, then the continuation of this recurrent procedure of generating tuples $g(3), g(4), \dots, g(k$-$\mathbf{1)}$ via free selection of odd parameters $l(3), l(4), \dots, l(k)$ gives the standard form of the map G_k_of degree d from the affine space of dimension $s+m(1)+m(2)+\dots+m(k)$ onto free module of dimension $r+m(1)+m(2)+\dots+m(k)$ together with the trapdoor accelerator T_k.

The procedure of reimage computation of G_k is similar to the case of Example 1.2.

Remark 4. 2. (nonlinear disturbance). In both examples instead of linear map C any nonlinear surjective map H of degree at most d with the trapdoor accelerator can be used. In particular one can use quadratic transformations of arbitrary free module K^n presented in [24] and [31]. In the case of Example 2. In case of finite field many classical broken or unbroken multivariate cryptosystem can be used (see [32] and further references).

3. On the multivariate maps of prescribed density with the trapdoor accelerator

Let Assume that commutative ring K contains nontrivial multiplicative group K^*. Let us consider the totality ${}^nES(K)$ of endomorphisms of $K[z_1, z_2,..., z_n]$ of kind

$$\begin{aligned}&z_1 \longrightarrow q_1 z_1^{a(1,1)} z_2^{a(1,2)} \dots z_n^{a(1,n)},\\ &z_2 \longrightarrow q_2 z_1^{a(2,1)} z_2^{a(2,2)} \dots z_n^{a(2,n)},\\ &\dots\\ &z_n \rightarrow q_n z_1^{a(n,1)} z_2^{a(n,2)} \dots z_n^{a(n,n)}\end{aligned} \qquad (3)$$

where q_i are regular elements of finite commutative ring K with the unity.

It is easy to see that the complexity of the composition of two elements of kind (3) is $O(n^3)$.

The semigroup ${}^nES(K)$ acts naturally on $(K^*)^n$ and contains large subgroup ${}^nEG(K)$ of bijective transformations of the variety [1].

Recall that we define density $den\ (f)$ of element f from $K[z_1, z_2, ..., z_n]$ written in its standard form as its number of monomial terms. The density of the tuple $H(z_1, z_2, ..., z_n)$ is defined as maximum of $den(h_i)$, $i = 1,2, ..., m$.

The following statements are proven in [22].

Proposition 1. 3. *Let us consider map introduced above map $F = F(a(1), a(2), ..., a(l), b(1), b(2), ..., b(l), c, D_1, D_2, ..., D_l)$ in the case of the temporal graph ${}^QDS_{s,r}(K)^t$ where K is a commutative ring with nontrivial multiplicative group K^*. Assume that the densities of a(i), b(i) and c are of size $O(s^{\alpha(i)})$, $O(s^{\beta(i)})$ and $O(s^{\gamma})$ such that $0 \le \alpha(i)+\beta(i) \le d$ and $\gamma \le d$ for some d, $d \ge 0$. Then den F has size $O(s^d)$.*

Remark 1.3. Parameter ***d*** can be selected as a rational number.

Corollary 1.3. *Let s = r or l is even, $r = O(s)$, $m = O(s^{\mu})$, $1 \le \mu \le 2$, H be an element of ${}^{s+m}ES(K)$ and $L \epsilon AGL_{s+m}(K)$* and *F satisfies conditions of Proposition 1.3. Then the density of standard form of G = HFL* is $O(s^{d+\mu}) = O((s+m)^{d/\mu+1})$.

Remark 2.3. We can select L of density $O(1)$ or density $O(m^{\lambda})$, $0 \le \lambda \le 1$. The simplest case is of kind $z_i \longrightarrow d_{ii}\dot{z}_i + d_{ii+1}\dot{z}_{i+1} + \dots + d_{is+m}\dot{z}_{i+sm}$, $i = 1, 2, ..., m+s$. Then the density of the map is $O((s+m)^{d/\mu+\lambda})$.

Corollary 2.3. *Assume that conditions of Corollary 1.3 holds and C = EN, where $E \epsilon {}^sEG(K)$, $N \epsilon {}^sCG(K)$. Then G induces an injective map of $(K^*)^{s+m}$ into $(K)^{s+m}$.*

Let $M_s(K) = GL_s(K) \cap {}^sES(K)$ be the monomial group of linear transformations.

Corollary 3.3. *Assume that conditions of Proposition 1.3 hold and*
$H \epsilon M_{s+m}(K)$ and $C \epsilon {}^sCG(K)$. Then G is a bijective map of K^{s+m} onto itself.

Formulated above statements allow us to construct element G of ${}^nCG(K)$ of unbounded degree and prescribed density d, $d \ge O(n)$ with the trapdoor accelerator.

We define *multiplicative trapdoor accelerator (F, T)* of F which is the map of density d such that its restriction F' on $(K^*)^n$ is injective map and the knowledge of T allows to compute the reimage of F' in a polynomial time.

Remark 3.3. We can construct multiplicative accelerators (F, T) where $F \epsilon {}^nCS(K)$ has unbounded degree and prescribed density $O(n^d)$, $d \ge 0$.

Algorithm 1.3. Public key with the multivariate map G with the multiplicative trapdoor accelerator.

Alice select even parameter ***l*** of size $O(1)$ and commutative ring K with nontrivial multiplicative group K^*. Natural examples are finite field F_q or modular arithmetic Z_q where $q = 2^s$, $s>1$.

She selects parameters n and $k = O(n)$ together with the subset $Q = \{\alpha(1), \alpha(2), ..., \alpha(m)\}$ of Cartesian product of $\{1, 2, ..., n\}$ and $\{1, 2, ..., k\}$ of cardinality m, $m = O(n^{\mu})$ where $1 \le \mu \le 2$. Alice will work with graph ${}^QDS_{n,k}(R)^t$, $k = O(n)$, $R = K[z_1, z_2, ..., z_n, z_{\alpha(1)}, z_{\alpha(2)}, ..., z_{\alpha(m)}]$. She selects parameter d and tuples of polynomials $a(1), a(2), ..., a(l), b(1), b(2), ..., b(l)$ with coordinates from $K[z_1, z_2, ..., z_n]$ satisfying conditions of Proposition 1.3, i.e. $den(a(i))$ has *size* $O(s^{\alpha(i)})$,

$den(b(i))$ has size $O(s^{\beta(i)})$ and $\alpha(i)+\beta(i) = d$.

Alice forms the tuples a_i, b_i, $i = 1, 2, \dots l$ of with coordinates of kind $q_1 z_1^{a(1,1)} z_2^{a(1,2)} \dots z_n^{a(1,s)} + q_2 z_1^{a(2,1)} z_2^{a(2,2)} \dots z_n^{a(2,n)} + \dots + \mathrm{q}_r z_1^{a(r,1)} z_2^{a(r,2)} \dots z_n^{a(r,n)}$ where $q_i \neq 0$.

She selects the pair of $E, E'\epsilon {}^{n}EG(K)$ such that $(EE', (K^*)^n)$ and $(E'E, (K^*)^s)$ are identity permutations.

The procedure 1 for this step is given below. She takes N of density $O(1)$ from $AGL_n(K)$ and L from $AGL_{n+m}(K)$ together with H and H' from ${}^{m+n}EG(K)$ such that HH' and $H'H$ are identity transformations of $(K^*)^{s+m(1)}$. Alice computes $C=EN$ moving $(z_1, z_2, \ldots, z_n)$ to $c = (c(1), c(2), \ldots, c(n))$.

She select parameters ${}^{i,j}\alpha(t)\epsilon K^*$, ${}^{i,j}\beta(t)$ and ${}^{i,j}\gamma(t)$ where $t = 1, 2, \ldots, l$, $(i, j)\epsilon Q$ for construction of momentum Jordan-Gauss graphs $D_1, D_2, \ldots, D_l$ of the temporary graph ${}^{Q}DS_{s,k}(K)^t$.

Alice will use $D_j(K[z_1, z_2, \ldots, z_s, z_{\alpha(1),} z_{\alpha(2),} \ldots, z_{\alpha(m(1))}])$ which are special momentum graphs of ${}^{Q}DS_{s,k}(R)^t$ defined by equations with coefficients from K but with the point set R^{n+m} and line set R^{k+m}.

She uses symbolic computation in the graph ${}^{Q}DS_{n,k}(R)^t$ to construct the transformation $F = F(a(1), a(2), \ldots, a(l), b(1), b(2), \ldots, b(l), c, D_1, D_2, \ldots, D_l)$ of K^{n+m} to itself. Alice uses Procedure 1 to form H from ${}^{n+m}\ EG(K)$. She forms L from $AGL_{n+m}(K)$ of density $O(m^{\lambda})$, $\lambda\leq 1$ and the element $G = HFL$ of affine Cremona semigroup. She computes the standard form of G and announces this multivariate rule publicly.

The standard form of G will be used as an encryption tool in the case of the space of plaintexts $(K^*)^{n+m}$.

Alice generates the map via special walks on the graph. The degree of the map G is $O(n+m)$. The density of the map is $O(n+m)^{\lambda+d/\mu}$.

Thus the complexity of encryption of computation of the image of $(p_1, p_2, \ldots, p_{n+m})\epsilon(K^*)^{m+n}$ is $O(n+m)^{\lambda+d/\mu+1}$.

Decryption procedure.

Public user Bob writes his plaintext $p = (p_1, p_2, \ldots, p_{m+n})$ and sends the ciphertext $s = G(p)$ to Alice.

Alice decrypts via the following procedure.

She computes $L^{-1}(s) = (d_1, d_2, \ldots, d_n, d_{\alpha(1)}, d_{\alpha(2)}, \ldots, d_{\alpha(m)}) = d$. Alice creates intermediate tuple of variables $(z_1, z_2, \ldots, z_n, z_{\alpha(1)}, z_{\alpha(2)}, \ldots, z_{\alpha(m)})$ consider the equations. She computes $N^{-1}(d_1, d_2, \ldots, d_n) = (e_1, e_2, \ldots, e_n)$ and considers the equations

$E(z_1, z_2,\ldots, z_n) = e_1$,

$E(z_1, z_2,\ldots, z_n) = e_2$,

…,

$E(z_1, z_2,\ldots, z_n) = e_n$,

Alice uses E' and gets the solution $z_1 = t_1, z_2 = t_2, \ldots, z_n = t_n$.

She computes $a(i)(t_1, t_2, \ldots, t_n) = a^*_i$, $i = 1, 2, \ldots, l$, $b(i)(t_1, t_2, \ldots, t_n) = b^*_i$, $i = 1, 2, \ldots, l$ and writes the system of linear equations $F = F(a^*(1), a^*(2), \ldots, a^*(l), b^*(1), b^*(2), \ldots, b^*(l), d')(t_1, t_2, \ldots, t_n, z_{\alpha(1)}, z_{\alpha(2)}, \ldots, z_{\alpha(m)}) = d$ where $d' = (d_1, d_2, \ldots, d_n)$.

This system is already written in row-echelon form.

So Alice gets the solution $z_{\alpha(1)} = t_{\alpha(1)}, z_{\alpha(2)} = t_{\alpha(2)},\ldots,z_{\alpha(m)} = t_{\alpha(m)}$.

She forms $t = (t_1, t_2, \ldots, t_n, t_{\alpha(1)}, t_{\alpha(2)}, \ldots, t_{\alpha(m)})$ and p as $H'(t)$.

Procedure 1.3. Let K be a finite commutative ring with unity and nontrivial multiplicative group K^* of order $d>1$. Assume that parameter n is selected and we have the task of generating two elements E and E' of ${}^{n}EG(K)$ such that EE' and $E'E$ act on $(K^*)^n$ as identity transformations.

We form the transformation $\mathcal{J}_1$ and $\mathcal{J}_2$ from ${}^{n}EG(K)$ of kind

$y_1 = \mu_1 x_1^{a(1,1)}$

$y_2 = \mu_2 x_1^{a(2,1)}\ x_2^{a(2,2)}$

…

$y_n = \mu_n x_1^{a(n,1)}\ x_2^{a(n,2)} \ldots x_n^{a(n,n)}$

where $(a(1,1), d) = 1, (a(2, 2), d) = 1, \ldots, (a(n, n), d) = 1$,

$z_1 = \mu'_1 y_1^{b(1,1)}\ y_2^{b(1,2)} \ldots y_n^{b(1,n)}$

$z_2 = \mu'_1 y_2^{b(2,2)}\ y_2^{b(2,3)} \ldots y_n^{b(2,n)}$

…

$z_n = \mu'_n y_n^{b(n,n)}$

where $(b(n,n), d) = 1, (b(n-1, 2), d) = 1, \ldots, (b(1, n), d) = 1$.

The computation of inverses J'_1 and J'_2 of the transformations J_1 and J_2 of the variety $(K^*)^n$ is straightforward. So Alice computes $E = J_1J_2$ and $E' = J'_2J'_1$.

Similarly, she constructs lower triangular and upper triangular bijective transformations JG_1 and JG_2 from $({}^{m+n}ES(K), (K^*)^{m+n})$.

So Alice computes $H = GJ_1 GJ_2$ and $H' = GJ'_2 GJ'_1$.

In case $d = 0$ and $\lambda = 0$ when the density of $a(i)$, $b(i)$, and L are $O(1)$ we obtain a pseudolinear cryptosystem. Its complexity for the encryption is $O(n+m)^2$.

In the case of $d/\mu+\lambda<1$ we get sub quadratic cryptosystem. It has complexity better than $O(n+m)^3$.

If $d/\mu+\lambda = 1$ we obtain a pseudo quadratic cryptosystem.

More general methods of generation of invertible elements of ${}^nES(K)$ can be found in [1].

The family of pseudo quadratic transformations with the trapdoor accelerators based on the modification $DS_{n,k}(K)$ in terms of generalisations of projective geometries was presented in [33].

Corollary 4.3. *Let K be a commutative ring with nontrivial multiplicative group K^*. Then for each natural n, n>2 we can construct a multivariate map of the prescribed density with the multiplicative trapdoor accelerator.*

Recall that $G = E^QQL$ induces an injective map of $(K^*)^{n+m}$ into K^{n+m}.

The standard form of G has the trapdoor accelerator Q, E, L, H, N, a_i, b_i, $i = 1, 2, \ldots, l$, T'. We assume that equations of $DS(n, K)$ are known publicly.

Remark 4.3. Note that the map with the trapdoor accelerator of polynomial density $O(n^d)$ where d, $d \geq 2$ is a natural number can be obtained as the product of J_1 and J_2 of Procedure 1 and selected multivariate map F of degree d with the trapdoor accelerator T.

In [34] the first implementation of the scheme of the previous Remark 4.3 was presented in the case of F induced by the special walk on the Jordan-Gauss graphs $D(n, K)$ and $A(n, K)$ for the case when $K = F_q$, of characteristic 2. Recall that in [24] the special walks of odd length in the Jordan-Gauss graphs $D(n, K)$ of type $1, 1, n\text{-}1$ were used for the generation of quadratic multivariate map F with the trapdoor accelerator.

In [35] the scheme of remark 4.3 was suggested for the graph-based encryption with $D(n, K)$ and $A(n, K)$ in the case of arithmetical rings $Z_{m.}$

The point $(p) = (p_1, p_2, \ldots, p_n)$ of the graph $D(n, K)$ is incident with the line $[l] = [l_1, l_2, \ldots, l_n]$, if the following relations between their coordinates hold:

$l_2\text{-}p_2 = l_1p_1$, $l_3\text{-}p_3 = l_2p_1$, $l_4\text{-}p_4 = l_1p_2$, $l_i\text{-}p_i = l_1p_{i-2}$, $l_{i+1}\text{-}p_{i+1} = l_{i-1}p_1$, $l_{i+2}\text{-}p_{i+2} = l_ip_1$, $l_{i+3}\text{-}p_{i+3}=l_1p_{i+1}$ where $i\geq5$.

So the encryption scheme is the following. Let us take graph $D(n, K[x_1, x_2, \ldots, x_n])$, sequence of colors $d(1)+x_1, d(2)+ x_1, \ldots, d(k)+x_1$ where $d(i) \in K)$, k is a length of walk.

Then we have to take sequence $x = (x_1, x_2, \ldots, x_n)$ (point from $D(n, K[x_1, x_2, \ldots, x_n])$), $v_1 = N_{d(1)+x1}(x)$, $N_{d(2)+x1}(v_1) = v_2$, $N_{d(k)+x1}(v_{k-1}) = v_1$.

Let F be the map $x_1 \longrightarrow v_1(x_1, x_2, \ldots, x_n)$, $x_2 \longrightarrow v_2(x_1, x_2, \ldots, x_n)$, ..., $x_n \longrightarrow v_n(x_1, x_2, \ldots, x_n)$. Then $\deg F = 3$.

We consider the map of kind $G = J_1J_2L_1FL_2$ where L_1 and L_2 are elements of $AGL_n(K)$.

In the case of linguistic graph $A(n, K)$ we simply change the incidence condition between points and lines:

$l_2\text{-}p_2 = l_1p_1$, $l_3\text{-}p_3 = p_1l_2$, $l_4\text{-}p_4 = l_1p_3$, $l_5\text{-}p_5 = p_1l_4$, ...

$l_n\text{-}p_n = l_1p_{n-1}$ (for even n) or $l_n\text{-}p_n = p_1l_{n-1}$ (for odd n).

Recently we implement the generating process described above map G in the case when K is arithmetic ring Z_q, $q=2^{32}$.

Let us denote G as $G(n, k, K)$ in the case when the length of the sequence of colours $d(1), d(2), \ldots, d(k)$ has length k. We present time the total number $M(G)$ of monomial terms in all g_i (global density). We refer to parameter k as the *length of the walk*. We can see the "condensed matter physics" digital effect. If k is "sufficiently large", then $M(g)$ is independent of k constant c.

We have written a program for generating elements and for encrypting a text using the generated public key. The program is written in C++. We use an average PC with processor Pentium 3.00 GHz, 2GB memory RAM, and system Windows 7.

We have implemented three cases:

1. L_1 and L_2 are identities.
2. L_1 and L_2 are maps of kind $z_1 \longrightarrow z_1+a_2z_2+\underline{a_3z_3}+ \ldots +a_tz_t$, $z_2 \longrightarrow z_2$, $z_3 \longrightarrow z_3, \ldots, z_n \longrightarrow z_n$, $a_i \neq 0$, $i = 1, 2, \ldots, n$ (linear time of computing for L_1 and L_2).
3. $L_1 = Ax+b$, $L_2 = A_1x+b_1$; matrices A, A_1 and vectors b, b_1 have mostly nonzero elements.

Tables 1, 2, and 3 present the case of graphs *D(n, K).*

Tables 4. 5, and 6 corresponds to the case 2 of graph *A(n, K).*

Table 1

Number of monomial terms of the map induced by the graph $D(n, Z_2^{32})$ case I

n	length of the walk				
	16	32	64	128	256
16	152	152	152	152	152
32	559	560	560	560	560
64	1615	2143	2144	2144	2144
128	3727	6303	7977	8384	8384

Table 2

Number of monomial terms of the cubic map induced by the graph $D(n, Z_2^{32})$, case II

n	length of the walk				
	16	32	64	128	256
16	6118	6118	6118	6118	6118
32	76447	76448	76448	76448	76448
64	813949	1066557	1066606	1066606	1066606
128	7373743	11418905	14820693	15857411	15858485

Table 3

Number of monomial terms of the map induced by the graph $D(n, Z_2^{32})$, case III

n	length of the walk				
	16	32	64	128	256
16	15504	15504	15504	15504	15504
32	209440	209440	209440	209440	209440
64	3065920	3065920	3065920	3065920	3065920
128	46866559	46866560	46866560	46866560	46866560

Table 4

Number of monomial terms of the map induced by the graph $A(n, Z_2^{32})$, case I

n	length of the walk				
	16	32	64	128	256
16	257	257	257	257	257
32	785	1025	1025	1025	1025
64	1841	3105	3849	4097	4097
128	3953	7265	9578	13681	15992

Table 5

Number of monomial terms of the map induced by the graph graph $A(n, Z_2^{32})$ case II

n	length of the walk				
	16	32	64	128	256
16	8840	8840	8840	8840	8840
32	94087	113168	113168	113168	113168
64	868811	1309887	1549061	1599519	1599520
128	7380170	11986066	17063985	19410643	22875533

Table 6

Number of monomial terms of the map induced by the graph $A(n, Z_2^{32})$, case III

n	length of the walk				
	16	32	64	128	256
16	15504	15504	15504	15504	15504
32	209440	209440	209440	209440	209440
64	3065920	3065920	3065920	3065920	3065920
128	46866560	46866560	46866560	46866560	46866560

Tables confirm that in Cases 2 and 3 we have multivariate transformations of density $O(n^3)$ *and global* $O(n^4)$. So, the encryption process for this map of unbounded degree takes $O(n^5)$.

Similarly to Example 1 we can use Proposition 4 iteratively.

Example 1.3. Alice selects finite commutative ring K and positive number s and prescribed degree d.

Step 1. Alice selects even parameter $l = l(1)$ of size $O(1)$ and the degree $d(1)$ of the initial map and parameter $\mu(1), 1\leq\mu(1)\leq2$.

She takes parameter $k = O(s)$ together with the subset $Q = \{\alpha(1), \alpha(2), ..., \alpha(m(1))\}$ of Cartesian product of $\{1, 2, ..., s\}$ and $\{1, 2, ..., k\}$ of cardinality $m(1), m(1) = O(s^{\mu(1)})$ where $1\leq\mu(1)\leq2$. Alice will work with graph ${}^{Q}DS_{s,k}(R)^{t}$, $k = O(s)$, $R = K[z_1, z_2, ..., z_s, z_{\alpha(1),} z_{\alpha(2),} ..., z_{\alpha(m(1))}]$. She selects tuples of polynomials $a(1) = a(1, 1), a(2) = a(1, 2), ..., a(l) = a(1, l), b(1, 1), b(1, 2), ..., b(1, l), l = l(1)$ with coordinates from $K[z_1, z_2, ..., z_s]$ satisfying conditions of Proposition 4, i.e.

deg $(a(i)) = \alpha(i)$, *deg* $b(i) = \beta(i)$ and $\alpha(i)+\beta(i) = d(1)$.

Alice forms the tuples $a_i, b_i, i = 1, 2, ... l$ of with coordinates of kind $q_1z_1{}^{a(1,1)} z_2{}^{a(1,2)} ... z_s{}^{a(1,s)} +q_2z_1{}^{a(2,1)} z_2{}^{a(2,2)} ... z_s{}^{a(2,s)}+...+q_rz_1{}^{a(r,1)} z_2{}^{a(r,2)} ... z_s{}^{a(r,s)}$ where $q_i \neq 0$.

She selects the pair of $E, E'\epsilon {}^{s}EG(K)$ such that $(EE', (K^{*})^{s})$ and $(E'E, (K^{*})^{s})$ are identity permutations. She takes N of density $O(1)$ from $AGL_{s}(K)$ and L of density $O(1)$ from $AGL_{s+m(1)}(K)$ together with $H = H_1$ and $H' = H'_1$ from ${}^{m(1)+s}EG(K)$ such that HH' and $H'H$ are identity transformations of $(K^{*})^{s+m(1)}$. Alice computes $C = EN$ moving $(z_1, z_2, ..., z_s)$ to $c = (c(1), c(2),..., c(s))$.
She selects parameters ${}^{i,j}\alpha_1(t)\epsilon K^{*}$, ${}^{i,j}\beta_1(t)$ and ${}^{i,j}\gamma_1(t)$ where $t = 1, 2, ..., l(1), (i, j)\epsilon Q$ for the construction of momentum Jordan-Gauss graphs ${}^{1}D_1, {}^{1}D_2, ..., {}^{1}D_l$ of the temporary graph ${}^{Q}DS_{s,k}(K)^{t}$.

Alice will use ${}^{1}D_j(K[z_1, z_2, ..., z_s, z_{\alpha(1),} z_{\alpha(2),} ..., z_{\alpha(m(1))}])$ which are special momentum graphs of ${}^{Q}DS_{s,k}(R)^{t}$, $R = K[z_1, z_2, ..., z_s, z_{\alpha(1),} z_{\alpha(2),} ..., z_{\alpha(m(1))}])$ defined by equations of ${}^{1}D_1, {}^{1}D_2, ..., {}^{1}D_l$ with coefficients from K but with the point set $R^{s+m(1)}$ and line set $R^{k+m(1)}$.

She uses symbolic computation in the graph ${}^{Q}DS_{s,k}(R)^{\ t}$ to construct the transformation $F = F_1 = F(a(1, 1), a(1, 2), ..., a(1, l(1)), b(1, 1), b(1, 2), ..., b(1, l), c, {}^{1}D_1, {}^{1}D_2, ..., {}^{1}D_{l(1)})$ of $K^{s+m(1)}$. She already formed $L = L_1$ from $AGL_{s+m(1)}(K)$ of density $O(1)$. Alice computes the element $G_1=H_1F_1L_1$ of affine Cremona semigroup. She computes the standard form of G_1. The degree of the map G_1 is $O(s+m(1))$.The density of the map is $O(s+m(1))^{d(1)/\mu(1)}$. The trapdoor accelerator T_1 consist of Q, equations of ${}^{Q}DS_{s,k}(K)$, tuples $a(1), a(2), ..., a(l), b(1), b(2), ..., b(l)$ and momentum graphs, transformations E, N and H_1, L_1.

Step 2 and the iteration.

Alice selects parameter $k(1) = O(s+m(1))$ and positive integer $s+m(1)\leq m(2)\leq s+m(k)k(1)$, even parameter $l(2)$ and constants $d(2), d(2)\geq d(1)/\mu(1)$ and $\mu(2)$ where $1\leq\mu(2)\leq 2$. She selects the subset $Q(1) = \{\alpha(1, 1), \alpha(1, 2), ..., \alpha(1, m(2))\}$ of Cartesian product of $\{1, 2, ..., s+m(1)\}$ and $\{1, 2, ..., k(1)\}$ of cardinality $m(2), m(2) = O((s+m(1))^{\mu(2)})$. Alice will work with graph ${}^{Q(1)}DS_{s+m(1),k(1)}(R)^{t}$, $R = K[z_1, z_2, ..., z_{s+m(1)}, z_{\alpha(1,1),} z_{\alpha(1, 2),} ..., z_{\alpha(1, m(2))}]$. She selects parameters to create momentum graphs ${}^{2}D_1, {}^{2}D_2, ..., {}^{2}D_{l(2)}$ of the temporary graph ${}^{Q(1)}DS_{s+m(1),k(1)}(K)^{t}$.

Alice will use ${}^{2}D_j(K[z_1, z_2, ..., z_{s+m(1)}, z_{\alpha(1, 1),} z_{\alpha(1, 2),} ..., z_{\alpha(1, m(2))}])$,

$\mathcal{J} = 1, 2, ..., l(2)$ which are special momentum graphs of ${}^{Q}DS_{s+m(1),\ k(1)}(R)^{t}$, $R = K[z_1, z_2, ..., z_s, z_{\alpha(1),} z_{\alpha(2),} ..., z_{\alpha(m(1))}])$ defined by equations of ${}^{2}D_1, {}^{2}D_2, ..., {}^{2}D_{l(2)}$ with coefficients from K but with the point set $R^{s+m(1)+m(2)}$ and line set $R^{k(1)+m(1+)m(2)}$.

She selects tuples $a(2, 1), a(2, 2), ..., a(2, l(2)), b(2, 1), b(2, 2), ..., b(2, l(2))$ with coordinates from $K[z_1, z_2, ..., z_{s+m(1)}]$ such that $den\ (a(2, i)b(2, i)) = O((s+m(1))^{d(2)})$. Alice constructs the transformation $F_2 = F(a(2, 1), a(2, 2), ..., a(2, l(2)), b(2, 1), b(2, 2), ..., b(2, l(2)), g(1), {}^{2}D_1, {}^{2}D_2, ..., {}^{2}D_{l(2)})$ of $K^{s+m(1)+m(2)}$ where g_1 is the tuple $(G_1(z_1), G_1(z_2), ..., G_1(z_{s+m(1)})$. She selects the pair of $H_2, H_2'\epsilon^{s+m(1)+m(2)}EG(K)$ such that $(H_2H_2', (K^{*})^{s+m(1)+m(2)})$ is identity permutations.

She selects $L = L_2$ from $AGL_{s+m(1)+m(2)}(K)$ of density $O(1)$ and forms

$G_2=H_2F_2L_2$.

The density of the standard form of the map G_2 will be determined as $O((s+m(1))^{d(2)})$ or $O((s+m(1)+m(2))^{d(2)/\mu(2)})$.The map G_2 has a multiplicative trapdoor accelerator T_2 which is extension of T_1 via adding $Q(1)$ of cardinality $m(2)$, parameters $k(1)$, $l(2)$ equations of ${}^{Q(1)}DS_{s+m(1),\ k(1)}\ (K)$, tuples $a(2.1), a(2, 2), ..., a(2, l(2)), b(2, 1), b(2, 2), ..., b(2, l(2))$, momentum graphs ${}^{2}D_1, {}^{2}D_2, ..., {}^{2}D_{l(2)}$ and transformations H_2, H_2', L_2.

Alice takes parameters $d(3), d(3)\geq d(2)/\mu(2), \mu(3), 1\leq\mu(3))\leq 2, k(2)$ of size $O(s+m(1)+m(2))$ and $m(3)$ of size $O(\ (s+m(1)+m(2))^{\mu(3)}$ such that $s+m(1)+m(2)\leq m(3)\leq(s+m(1)+m(2))k(2)$. She takes even parameter $l(3)$ and selects subset $Q(2) = \{\ \alpha(2, 1), \alpha(2, 2), ..., \alpha(2, m(3))\}$ of Cartesian product of $\{1, 2, ..., s+m(1)+m(2)\}$ and $\{1, 2, ..., k(2)\}$.

Alice will work with graph ${}^{Q(2)}DS_{s+m(1)+m(2),k(2)}(R)^{t}$, $R = K[z_1, z_2, ..., z_{s+m(1)+m(2)}, z_{\alpha(2,1),} z_{\alpha(2, 2),} ..., z_{\alpha(2, m(3))}]$. She selects parameters to create momentum graphs ${}^{3}D_1, {}^{3}D_2, ..., {}^{3}D_{l(3)}$ of the temporary graph ${}^{Q(2)}DS_{s+m(1)+m(2),\ k(2)}(K)^{t}$.

Alice forms tuples $a(3, 1), a(3, 2), ..., a(3, l(3)), b(3, 1), b(3, 2), ..., b(3, l(3))$ with coordinates from $K[z_1, z_2, ..., z_{s+m(1)+m(2)}\ ]$ such that $den\ (a(3, i)b(3, i)) = O((s+m(1)+m(2)+m(3))^{d(3)}$. Alice constructs the transformation $F_3 = F(a(3, 1), a(3, 2), ..., a(3, l(3)), b(3, 1), b(3, 2), ..., b(3, l(3)), g(2), {}^{3}D_1, {}^{3}D_2, ..., {}^{3}D_{l(3)})$ of $K^{s+m(1)+m(2)+m(3)}$ where $g(2)$ is the tuple $(G_2(z_1), G_2(z_2), ..., G_2(z_{s+m(1)+m(2)})$.
She selects the pair of $H_3, H_3'\epsilon^{s+m(1)+m(2)+m(3)}EG(K)$ such that $(H_3H_3', (K^{*})^{s+m(1)+m(2)+m(3)})$ is identity permutation.

She selects $L = L_3$ from $AGL_{s+m(1)+m(2)+m(3)}(K)$ of density $O(1)$ and forms $G_3 = H_3F_3L_3$.

The density of the standard form of the map G_3 will be determined as $O((s+m(1)+m(2)^{d(3)})$ or $O((s+m(1)+m(2)+m(3))^{d(3)/\mu(3)})$. The map G_3 has a multiplicative trapdoor accelerator T_3 which is extension of T_2 via adding $Q(2)$ of cardinality $m(3)$, parameters $k(2)$, $l(3)$, equations of ${}^{Q(2)}DS_{s+m(1)+m(2),k(2)}(K)$, tuples $a(3, 1), a(3, 2), \ldots, a(3, l(2)), b(3, 1), b(3, 2), \ldots, b(3, l(3))$, momentum graphs ${}^3D_1, {}^3D_2, \ldots, {}^3D_{l(3)}$ and transformations H_3, H_3', L_3.

Alice continues the iterative process. She creates $G_4, G_5, \ldots, G_r$ of the densities of kind $O(n(i))^{\beta(i)}$, $i = 4, 5, \ldots, r$ where $n(i)$ is the dimension of the space of ciphertexts and $\beta(i) = d(i)/\mu(i)$ with the multiplicative trapdoor accelerators T_i, $i =4, 5, \ldots, r$ respectively.

So the final map G_r of $K^{s+m(1)+m(2)+\ldots+m(r)}$ to itself with the multiplicative trapdoor accelerator T_r has a polynomial density.

Recall that $d(i) \geq d(i-1)/\mu(i-1)$ for $i = 2, 3, \ldots, r$. In the case when these inequalities become equalities $d(r)/\mu(r) = d(1)/(\mu(1)\mu(2)\mu(3) \ldots \mu(r))$.

Alice can select $d(1) = 0$ when G_r has density $O(1)$. Then the output will be a pseudolinear map. The choice of small parameter $d(1)$ will allow her to get sub quadratic map of the density $O(n^{\lambda})$ with arbitrarily selected λ, $\lambda<1$. Obviously, Alice can create the map G_r of prescribed density $O(n^d)$ with the multiplicative trapdoor accelerator.

Note that Alice can take $G_r L$ where L has degree 1 and density $O(n)$ and use the standard form of transformation of density $O(n^{d+1})$ with the multiplicative trapdoor accelerator.

Procedure 1.3. (reimage computation for (G_r, T_r)). Assume that $G_j = H_jF_jL_j$, $j = 1, 2, \ldots, r$ and $F_j = F(a(1, j), a(2, j), \ldots, a(l(j), j), b(1, j), b(2, j), \ldots, b(l(j),j), g(j-1), {}^jD_1, {}^jD_2, \ldots, {}^jD_{l(j)})$ acting on the affine space ${}^jW$ of dimension $s+m(1)+m(2)+\ldots+m(j) = n(j)$.

Alice obtained the ciphertext ${}^0c = ({}^0c_1, {}^0c_2, \ldots, {}^0c_{n(r)})$. She computes $L_r^{-1}({}^0c) = {}^rc$ and takes its projection ${}^rc'$ on the first $n(r-1)$ coordinates.

Alice computes $L_{r-1}^{-1}({}^rc') = {}^{r-1}c$ and takes its projection ${}^{r-1}c'$ on the first n(r-2) coordinates. She continue this procedure and gets the tuples ${}^1c = (b_1, b_2, \ldots, b_s, b_{s+1}, b_{s+2}, \ldots, b_{s+m(1)})$ and ${}^1c' = (b_1, b_2, \ldots, b_s)$.

Alice forms the intermediate tuple $(z_1, z_2, \ldots, z_s)$ and investigates the system of linear equations $c_1(z_1, z_2, \ldots, z_s) = b_1, c_2(z_1, z_2, \ldots, z_s) = b_2, \ldots, c_s(z_1, z_2, \ldots, z_s) = b_s$. She gets the solution $z_1 = \alpha_1, z_2 = \alpha_2, \ldots, z_s = \alpha_s$. In fact $(\alpha_1, \alpha_2, \ldots, \alpha_s) = E'(N^{-1}(b_1, b_2, \ldots, b_s))$.

Alice computes tuples $a^*(i, 1) = a(1, 1)(\alpha_1, \alpha_2, \ldots, \alpha_s)$, $b^*(i, 1) = b(1, 1)(\alpha_1,\alpha_2, \ldots, \alpha_s)$, $i = 1, 2, \ldots, l(1)$ with coordinates from K.

Alice takes graph ${}^1D_{l(1)}$ and computes $d(l(1)) = J_{b^*(l(1),1)}({}^1c)$. She takes the neighbour $d'(l(1) = N_{a^*(l(1)),1)}(d(l(1))$ of the point $d(l(1)$ of colour $a^*(l(1), 1)$.

Alice treats the tuple $d'(l(1))$ as the line of the graph ${}^1D_{l(1)}$. She computes $J_{b^*(l(1)-1),1)}(d'(l(1)) = d(l(1)-1)$ and its neighbour $d'(l(1)-1) = N_{a^*(l(1)-1),1)}(d(l(1)-1)$. Alice continue this process and gets $d'(1) = N_{a^*(1,1)}(d(1))$ in the graph ${}^1D_1$. So she gets $e(1) = J_\gamma(d'(1))$, $\gamma = (\alpha_1,\alpha_2,\ldots, \alpha_s)$.

The tuple $(H_1)'(e(1)) = r(1)$ is the solution of the equation $H_1F_1(z_1, z_2, z_s, z_{s+1}, \ldots, z_{s+m(1)}) = {}^1c = (L_1)^{-1}({}^2c')$ which is equivalent to $G_1(z_1, z_2, z_s, z_{s+1}, \ldots, z_{s+m(1)}) = {}^2c'$.

Alice considers the equation $H_2F_2(z_1, z_2, \ldots, z_s, z_{s+1}, \ldots, z_{s+m(1)}, z_{s+m(1)+1}, \ldots, z_{s+m(1)+m(2)}) = {}^2c = L_2({}^3c')$.

The first $s+m(1)$ equations of this system are equivalent to $H_1F_1(z_1, z_2, \ldots, z_{s+m(1)}) = {}^1c$ with the solution $\gamma(1) = ({}^1\alpha_1, {}^1\alpha_2, \ldots, {}^1\alpha_{s+m(1)})$ obtained due to the knowledge of the trapdoor accelerator.

Alice computes the specializations $a^*(1, 2), a^*(2, 2), \ldots, a^*(l(2), 2), b^*(1, 2), b^*(2, 2), \ldots, b^*(l(2), 2)$ of $a(1, 2), a(2, 2), \ldots, a(l(2), 2), b(1, 2), b(2, 2), \ldots, b(l(2), 2)$ under the substitution $z_1 = {}^1\alpha_1$, $z_2 = {}^1\alpha_2, \ldots, z_{s+m(1)} = {}^1\alpha_{s+m(1)}$.

She computes the point $d(l(2)) = J_{b^*(2,\ l(2))}({}^2c)$ and line $d'(l(2)) = N_{a^*(2,\ l(2))}(d(l(2)))$ of the graph ${}^2D_{l(2),}$ computes $d(l(2)-1) = J_{b^*(2,\ l(2)-1)}(d'(l(2))$ and vertex $d'(l(2)-1) = N_{a^*(2,\ l(2)-1)}(d(l(2)-1))$ of the graph ${}^2D_{l(2)-1}$. Alice continue this process and gets $d'(1)= N_{a^*(2,1)}(d(1))$ in the graph ${}^2D_1$. So she gets $e(2) = J_{\gamma(1)}(d'(1))$ in this graph.

The tuple $(H_2)'(e(2)) = \gamma(2)$ is the solution of the equation $H_2F_2(z_1, z_2, z_s, z_{s+1}, \ldots, z_{s+m(1)}, z_{s+m(1)+1}, \ldots, z_{s+m(1)+m(2)}) = {}^2c=L_2({}^3c')$ which is equivalent to $G_2(z_1, z_2, z_s, z_{s+1}, \ldots, z_{s+m(1)+m(2)}) = {}^3c'$. Alice continue this recurrent process and gets the solution $\gamma(r)$ of the equation $G_r(z_1, z_2, z_s, z_{s+1}, \ldots, z_{s+m(1)+m(2)+\ldots+m(k)}) = {}^0c$.

Remark 5.3. *(nonlinear disturbance).* In this iterative algorithm instead of the combination EN on K^s one can take any pseudolinear map Z with the multiplicative trapdoor accelerator at most d with the trapdoor accelerator can be used.

4. On the safe delivery of multivariate maps

4.1. On protocols of Noncommutative Cryptography

The following protocol is one of the classical instruments of Noncommutative Cryptography.

Twisted Diffie-Hellman protocol.

Similarly, let S be an abstract group which has some invertible elements.

Alice and Bob pose common element $g \epsilon S$ and the pair of invertible elements h, h^{-1} from this semigroup.

Alice selects natural numbers $k(A)$ and $r(A)$, *and* she forms

$h^{-r(A)}g^{k(A)}h^{r(A)} = g_A$.

Bob chooses $k(B)$ and $r(B)$, *and* he forms $h^{-r(B)}g^{k(B)}h^{r(B)} = g_B$.

They exchange g_A, g_B and compute the collision element X as

${}^{A}g = h^{-r(A)}{g_B}^{k(A)}h^{r(A)}$

(Alice) and ${}^{B}g = h^{-r(B)}{g_A}^{k(B)}h^{r(B)}$ (Bob) respectively.

The security of this scheme is based on the complexity of the Power Conjugacy Problem, the adversary has to solve the equation $h^{-x}g^{y}h^{x} = b$, where b coincides with g_B or g_A. The complexity of this problem essentially depends on the choice of highly noncommutative platform S.

In the case of platform $S = {}^{n}ES(K)$ where $K = F_q$ or $K = Z_q$ this problem is intractable even with the use of a quantum computer.

The computational complexity of this protocol is $O(n^3)$.

If we assume that the degree of transformations h and g from ${}^{n}ES(K)$ is $O(1)$ then the complexity of the protocol is $O(n)$.

Other platforms defined in terms of multivariate cryptography and corresponding protocols readers can be found in [22, 36–41].

Foundations of Noncommutative Cryptography, description of algorithms and cryptanalytic results reader can find in [42–62].

4.2. Some definitions

Let F be the map from $(K^*)^n$ in K^n of density $O(n^d)$, $0 \leq d \leq 1$ such that its restriction on $(K^*)^n$ is injective. Assume that T is a multiplicative trapdoor accelerator of F and Alice has the pair (F, T).

Below please find examples of the deformation.

Example 1.4. (the case of maps of unbounded degree).

Alice and Bob conduct Twisted Diffie-Hellman protocol based on the platform ${}^{n}ES(K)$. Assume that the collision map C is given by formula (1). Correspondents can use subsemigroup of ${}^{n}ES(K)$ with generators from the set $M = \{g, h, C\}$. They use open channel to agree on words $w_j(C) = ({}^{j}g_{i(1)}, {}^{j}g_{i(2)}, \ldots, {}^{j}g_{i(s(j))})$ of length $s(j)$, $s(j) \geq 1$ where ${}^{j}g_{i(1)}, {}^{j}g_{i(2)}, \ldots, {}^{j}g_{i(s(j))}$, $j = 1, 2, \ldots, r$ is the sequence of elements of the alphabet M which contains at least one appearance of C.

Let ${}^{j}g(C)$ be an element of ${}^{n}ES(K)$ generated as a product of characters of the $w_j(C)$. We form ${}^{j}h(C)$ sending x_i to ${}^{j}g(C)(x_i)a(i, j)$ where $a(i, j)$ are publicly known elements of $K\text{-}\{0\}$.

Let $G(C)$ be the sum ${}^{1}h(C) + {}^{2}h(C) + \ldots {}^{r}h(C)$.

Alice can send the tuple $(F(x_1)+G(C)(x_1), F(x_2)+G(C)(x_2), \ldots$

$F(x_n)+G(C)(x_n))$ to Bob. He is able to restore the map F.

This “steganographic” way of safe delivery of the multivariate map is secure even in the case r is a linear expression from n of size $O(n)$.

Example 2.4. (the case of nonlinear transformation of constant degree d).

Alice takes the collision element C and nonempty subsets of $\{1, 2, \ldots, n\}$ of kind $\{i(1), i(2), \ldots, i(m)\}$ of cardinality m, $1 \leq m \leq d$.

She form g_i as the linear combination of monomial terms $(q_{i(1)})^{a(i,\ i(1))}(q_{i(2)})^{a(i,\ i(2))}\ldots(q_{i(m)})^{a(i,\ i(m))}\ldots x_{i(1)}x_{i(2)}\ldots x_{i(m)}$ and constant $C(x_i)(q_1, q_2,\ldots, q_m)$ with known nonzero coefficients from K. Alice sends $(F(x_1)+g_1, F(x_2)+g_2, \ldots, F(x_n)+g_n)$ to Bob.

Remark 1.4. Assuming that the map F of degree $O(n)$ is not given publicly, Alice and Bob use it in a protocol-based secure way.

An adversary may intercept a polynomial number of pairs of kind plaintext/ciphertext but even this information can be insufficient for restoration of F without the knowledge of the symbolic type of Z, i.e. lists of nontrivial monomial terms of $F(x_i)$ with coefficients 1.

Remark 2.4. It is known that a polynomial system of equations of degree $d(n)$ can be rewritten as a system of quadratic equations via the method of introducing extra variables. If the degree is unbounded then the growth of the number of variables does not allow for investigating the resulting quadratic system. In case when the system is not given publicly the method of degree reduction can not be used.

Conclusions

The technique of Jordan-Gauss graphs and their temporal analog defined over arbitrary commutative ring K can be used for the construction of bijective multivariate map F of prescribed degree d on free module K^n with the trapdoor accelerator T which allows computing the reimage of the given value in a polynomial time. In the case of $d = 2$ and 3 such maps can be used for the construction of public keys.

If d is a constant larger than 3 we can construct sparse maps of density $O(n)$ with the trapdoor accelerator T. So the value of the function can be computed in time $O(n^2)$. For each constant d, we can construct the map of degree d, density $O(n)$, and trapdoor accelerator which allows the computation of the reimage in time $O(n^2)$. Recall that we define the density of F as the maximal density of polynomials $F(x_i)$ for $i = 1, 2, \ldots, n$.

It is known that there is a special way to increase the number of variables and rewrite the nonlinear system $F(x)=b$ as equivalent to it quadratic system in many variables. One can select "sufficiently large" d such that the corresponding quadratic system is unfeasible for cryptanalytic investigation.

We define $sup(t)$ of the monomial term $t = t(x_1, x_2, \ldots, x_n)$ as the number of variables x_i in positive power in the expression of t. The support of the multivariate map F is defined as the maximal value of $sup(F)$ supports its monomial terms. We can construct multivariate maps on K^n with the prescribed density $O(n^\alpha)$, $0\leq\alpha\leq1$, and prescribed support $O(n^\beta)$ with the trapdoor accelerator. We construct multivariate map F of unbounded degree with the support n and prescribed density $O(n^\beta)$ and multiplicative trapdoor accelerator.

Mentioned above pairs of kind (F, T) can be investigated as potential public key constructions of multivariate Cryptography. Alternatively, Alice can use her pair (F, T) in a different way. She and her trusted partner Bob can use twisted Diffie-Hellman protocol with the platform ${}^nES(K)$, and deform the output E via its transformation to polynomial transformation $D(E)$ of K^n. Alice sends $F+D(E)$ to Bob. The security of this asymmetric algorithm described in Section 1 rests on the security of the selected protocol.

Note that the mentioned above pairs (F, T) can be used as stream ciphers when the knowledge of T is shared between Alice and Bob.

Acknowledgments

This research is partially supported by the British Academy Fellowship for Researchers under Risk 2022.

Declaration on Generative AI

While preparing this work, the authors used the AI program Strike Plagiarism to search for possible plagiarism. After using this tool, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

References

[1] V. Ustimenko, On Eulerian semigroups of multivariate transformations and their cryptographic applications, Eur. J. Math. 9(93) (2023).

[2] J. Ding, A. Petzoldt, Current state of multivariate cryptography. IEEE Secur. Priv. 15(4) (2017) 28–36. doi:10.1109/MSP.2017.3151328

[3] D. Smith-Tone, 2F—A new method for constructing efficient multivariate encryption schemes, in: PQCrypto 2022, 13th International Conference on Post-Quantum Cryptography, virtual, DC, US, 2022.

[4] D. Smith-Tone, New practical multivariate signatures from a nonlinear modifier, IACR e-print archive, 2021/419.

[5] D. Smith-Tone, C. Tone, A nonlinear multivariate cryptosystem based on a random linear code. URL: https://eprint.iacr.org/2019/1355

[6] D. Jayashree, R. Dutta, Progress in multivariate cryptography: systematic review, challenges, and research directions, ACM Comput. Surv. 55(12(246)) (2023) 1–34. doi:10.1145/3571071

[7] F. Cabarcas, D. Cabarcas, J. Baena, Efficient public-key operation in multivariate schemes, Adv. Math. Commun. 13(2) (2019) 343-371. doi:10.3934/amc.2019023

[8] R. Cartor, D. Smith-Tone, EFLASH: A new multivariate encryption scheme, in: International Conference on Selected Areas in Cryptography, 2018, 281–299.

[9] A. Casanova, et al., Gemss: A great multivariate short signature, Submission to NIST, 2017.

[10] J. Chen, et al., A new encryption scheme for multivariate quadratic systems, Theor. Comput. Sci. 809 (2020) 372–383.

[11] M.-S. Chen, et al., SOFIA: MQ-based signatures in the QROM, in: IACR International Workshop on Public Key Cryptography, 2018, 3–33.

[12] D. H. Duong, et al., An efficient multivariate threshold ring signature scheme, Comput. Stand. Interfaces 74 (2021). doi:10.1016/j.csi.2020.103489

[13] J.-C. Faugere, et al., A new perturbation for multivariate public key schemes such as HFE and UOV, Cryptology ePrint Archive, 2022.

[14] W. Buellens, Improved cryptanalysis of UOV and rainbow improved cryptanalysis of UOV and rainbow, in: Advances in Cryptology, EUROCRYPT 2021, Lecture Notes in Computer Science, vol. 12696, 2021.

[15] 12th International Workshop, PQCrypto 2021, Lecture Notes in Computer Science, vol. 12841, 2021.

[16] Post quantum cryptography, in: 13th International Workshop, PQCrypt 2022, Virtual Event, September 28-30, 2022, Proceeding, Lecture Notes in Computer Science (LNCS, volume 13512)

[17] Post-quantum cryptography, in: 14th International Workshop, PQCrypto 2023, Lecture Notes in Computer Science, vol. 14154, 2023.

[18] Post-quantum cryptography, in: 15th International Workshop, PQCrypto 2024, Part 2, Lecture Notes in Computer Science, vol. 14772, 2024.

[19] N. Koblitz, Algebraic aspects of cryptography, Springer, 1998.

[20] V. Ustimenko, Maximality of affine group, hidden graph cryptosystem and graph's stream ciphers, J. Algebr. Discret. Math. 1 (2005) 51–65.

[21] V. Ustimenko, Linguistic Dynamical systems, graphs of large girth and cryptography, J. Math. Sci. 140(3) (2007) 412–434.

[22] V. Ustimenko, Graphs in terms of algebraic geometry, symbolic computations and secure communications in post-quantum world, UMCS Editorial House, 2022.

[23] V. Ustimenko, O. Pustovit, Jordan-Gauss graphs and quadratic public keys of multivariate cryptography, in: ITTAP 2024, 4th International Workshop on Information Technologies: Theoretical and Applied Problems, vol. 3896, 2024.

[24] V. Ustimenko, T. Chojecki, A. Wróblewska, On the Jordan-Gauss graphs andmultivariate public keys, in: IACR Cryptol. ePrint Arch., 2024/1793.

[25] T. Chojecki, et al., On affine forestry over integral domains and families of deep Jordan-Gauss graphs, Eur. J. Math. 11(10) (2025). doi:10.1007/s40879-024-00798-2

[26] V. Ustimenko, On computations with double schubert automaton and stable maps of multivariate cryptography, Interdiscip. Stud. Complex Syst. 19 (2021) 18–32. doi:10.31392/iscs.2021.19.018

[27] V. Ustimenko, On small world non Sunada twins and cellular Voronoi diagrams, Algebr. Discret. Math. 30(1) (2020) 118–142.

[28] A. Brower, A. Cohen, A. Nuemaier, Distance regular graphs, Springer, 1989.

[29] R. W. Carter, Simple Groups of Lie Type, Wiley, 1972.

[30] F. Buekenhout (Editor), Handbook on incidence geometry, North Holland, 1995.

[31] V. Ustimenko, A. Wróblewska, On extremal algebraic graphs, quadratic multivariate public keys and temporal rules, in: FedCSIS, 2023, 1173–1178.

[32] J. Ding, A. Petzoldt, D. S. Schmidt, Multivariate public key cryptosystems, 2nd ed., Advances in Information Security, Springer, 2020. doi:10.1007/978-1-0716-0987-3

[33] V. Ustimenko, O. Pustovit, On Schubert cells of projective geometry and pseudo-quadratic public keys of multivariate cryptography, in: Cybersecurity Providing in Information and Telecommunication Systems II, vol. 3826, 2024, 198–205.

[34] V. Ustimenko. O. Pustovit, On the implementations of new multivariate public keys based on transformations of linear degree, in: Proceedings of the Conference on Mathematical Foundations of Informatics MFOI 2919.

[35] V. Ustimenko, On new multivariate cryptosystems based on hidden Eulerian equations, Dopovidi of National Academy of Science of Ukraine N5, 2017.

[36] V. Ustimenko, A. Wroblewska, On the key exchange with nonlinear polynomial maps of stable degree, Annalles UMCS Informatica AI XI, 2 (2011) 81–93.

[37] V. Ustimenko, On short digital signatures with Eulerian transformations, IACR e-print archive, 2024/001.

[38] V. Ustimenko, On the restoration of historical matsumoto-imai cryptosystem and other schemes in terms of noncommutative cryptography, in: Future Technologies Conference (FTC) 2024, vol. 2, FTC 2024, Lecture Notes in Networks and Systems, vol. 1155. doi:10.1007/978-3-031-73122-8_7

[39] V. Ustimenko, M. Klisowski, On noncommutative cryptography with cubical multivariate maps of predictable density, in: Intelligent Computing, CompCom 2019, Advances in Intelligent Systems and Computing, vol. 998, 2019, 654–674.

[40] V. Ustimenko, M. Klisowski, On new protocols of noncommutative cryptography in terms of homomorphism of stable multivariate transformation groups, Algebra Discrete Math. 35(2) (2023) 220–250. doi:10.12958/adm1523

[41] V. Ustimenko, On multivariate algorithms of digital signatures on secure El Gamal-type mode, Computational Methods and Mathematical Modeling in Cyberphysics and Engineering Applications 1, 2024.

[42] D. N. Moldovyan, N. A. Moldovyan, A new hard problem over non-commutative finite groups for cryptographic protocols, in: international conference on mathematical methods, models, and architectures for computer network security, MMM-ACNS 2010: Computer Network Security, 2010, 183–194.

[43] L. Sakalauskas, P. Tvarijonas, A. Raulynaitis, Key agreement protocol (KAP) using conjugacy and discrete logarithm problem in group representation level, Informatica 18(1) (2007) 115–124.

[44] V. Shpilrain, A. Ushakov, The conjugacy search problem in public key cryptography: unnecessary and insufficient, in: Applicable Algebra in Engineering, Communication and Computing, vol. 17(3–4), 2006, 285–289.

[45] D. Kahrobaei, B. Khan, A non-commutative generalization of ElGamal key exchange using polycyclic groups, in: IEEE GLOBECOM 2006, Global Telecommunications Conference, 2006. doi:10.1109/GLOCOM.2006

[46] A. Myasnikov, V. Shpilrain, A. Ushakov, Group-based cryptography, Berlin, Birkhäuser Verlag, 2008.

[47] Z. Cao, New Directions of modern cryptography, Boca Raton: CRC Press, Taylor & Francis Group, 2012.

[48] B. Fine, et al. Aspects of non abelian group based cryptography: A survey and open problems, arXiv. doi:10.48550/arXiv.1103.4093

[49] A. G. Myasnikov, V. Shpilrain, A. Ushakov, Non-commutative cryptography and complexity of group-theoretic problems, American Mathematical Society, 2011.

[50] I. Anshel, M. Anshel, D. Goldfeld, An algebraic method for public-key cryptography. Math. Res. Lett. 6(3–4) (1999) 287–291.

[51] S. R. Blackburn, S. D. Galbraith, Cryptanalysis of two cryptosystems based on group actions, in: Advances in Cryptology, ASIACRYPT'99, Lecture Notes in Computer Science, vol. 1716, 1999, 52–61.

[52] K. H. Ko, et al, New public-key cryptosystem using braid groups, in: Advances in Cryptology—CRYPTO 2000, Santa Barbara, CA. Lecture Notes in Computer Science, vol. 1880, 2000, 166–183.

[53] G. Maze, C. Monico, J. Rosenthal, Public key cryptography based on semigroup actions, Adv. Math. Commun. 1(4) (2007) 489–507.

[54] P. H. Kropholler, et al., Properties of certain semigroups and their potential as platforms for cryptosystems, Semigroup Forum 81 (2010) 172–186.

[55] J. A. Lopez Ramos, et al., Group key management based on semigroup actions, J. Algebra Appl. 16(08) (2017) 1750148.

[56] G. Kumar, H. Saini, Novel noncommutative cryptography scheme using extra special group, Secur. Commun. Netw. 2017(1) (2017). doi:10.1155/2017/9036382

[57] A. Myasnikov, V. Roman'kov, A linear decomposition attack, Groups Complex. Cryptol. 7 (2015) 81–94. doi:10.1515/gcc-2015-0007.

[58] V. Roman'kov, A nonlinear decomposition attack, Groups Complex. Cryptol. 8(2) (2017) 197–207.

[59] V. Roman'kov, Two general schemes of algebraic cryptography, Groups Complex. Cryptol. 10(2) (2018) 83–98.

[60] V. Roman'kov, An improved version of the AAG cryptographic protocol, Groups Complex. Cryptol. 11(1) (2019). doi:10.1515/gcc-2019-2003

[61] B. Tsaban, Polynomial time solutions of computational problems in noncommutative algebraic cryptography, J. Cryptol. 28(3) (2015) 601–622.

[62] A. Ben-Zvi, A. Kalka, B. Tsaban, Cryptanalysis via algebraic spans, in: Advances in Cryptology – CRYPTO 2018, Lecture Notes in Computer Science, vol. 10991, 2018, 1–20.