On semigroups of multiplicative Cremona transformations and new solutions of Post Quantum Cryptography.

Vasyl Ustimenko

University of Maria Curie Sklodowska in Lublin, Poland
E-mail: vasyl@hektor.umcs.lublin.pl

Abstract. Noncommutative cryptography is based on the applications of algebraic structures like noncommutative groups, semigroups and noncommutative rings. Its intersection with Multivariate cryptography contains studies of cryptographic applications of subsemigroups and subgroups of affine Cremona semigroups defined over finite commutative ring K. We consider special semigroups of transformations of the variety $(K^*)^n$, $K=F_q$ or $K=Z_m$ defined via multiplications of variables.

Efficiently computed homomorphisms between such subsemigroups can be used in Post Quantum protocols schemes and their inverse versions when correspondents elaborate mutually inverse transformations of $(K^*)^n$.

The security of these schemes is based on a complexity of decomposition problem for element of the semigroup into product of given generators. So the proposed algorithms are strong candidates for their usage in postquantum technologies.

Key words: Postquantum Cryptography, Noncommutative and Multivariate Cryptography, key exchange protocols, inverse protocols, semigroups of transformations, decomposition problem.

1 On Post Quantum, Multivariate Cryptography and Noncommutative Cryptography.

Post Quantum Cryptography serves for the research on asymmetrical cryptographic algorithms which can be potentially resistant against attacks based on the use of quantum computer. The security of currently popular algorithms are based on the complexity of the following the three known hard problems: integer factorisation, discrete logarithm problem, discrete logarithm for elliptic curves. Each of these problems can be solved in polynomial time by Peter Shor's algorithm for theoretical quantum computer. Cryptographers have already started research on postquantum security. They have also counted on the new results of general complexity theory.

Modern PQC is divided into several directions such as Multivariate Cryptography, Nonlinear Cryptography , Lattice based Cryptography, Hash based Cryptography, Code based Cryptography, studies of isogenies for superelliptic curves.

The oldest direction is Multivariate Cryptography (see [1], [2], [3]) which uses polynomial maps of affine space K^n defined over a finite commutative ring into itself as encryption tools. It exploits the complexity of finding a solution of a system of nonlinear equations from many variables. Multivariate cryptography uses as encryption tools nonlinear polynomial transformations of kind

$$x_1 \to f_1(x_1, x_2, ..., x_n), x_2 \to f_2(x_1, x_2, ..., x_n), ...x_n \to f_n(x_1, x_2, ..., x_n)$$

transforming affine space K^n, where $f_i : K[x_1, x_2, ..., x_n], i = 1,2, ..., n$ are multivariate polynomials usually given in standard form, i. e. via a list of monomials in a chosen order.

We are going to present new cryptoalgorithms in the area of intersection of Multivariate Cryptography and Non-commutative cryptography which appeared with attempts to apply Combinatorial group theory to Information Security.

If G is noncommutative group then correspondents can use conjugations of elements involved in protocol, some algorithms of this kind were suggested in [4], [5], [6], [7], where group G is given with the usage of generators and relations. Security of such algorithms is connected to Conjugacy Search Problem (CSP) and Power Conjugacy Search Problem (PCSP), which combine CSP and Discrete Logarithm Problem and their generalizations.

The extension of group based cryptography is essentially wider direction of

Non-commutative cryptography which is an active area of cryptology, where the cryptographic primitives and systems are based on algebraic structures like groups, semigroups and noncommutative rings (see [8], [9], [10], [11], [12], [13], [14], [15], [16]). This direction of security research has very rapid development (see [17], [18]} and further references in these publications).

One of the earliest applications of a non-commutative algebraic structure for cryptographic purposes was the use of braid groups to develop cryptographic protocols. Later several other non-commutative structures like Tompson groups and Grigorchuk groups have been been identified as potential candidates for cryptographic post quantum applications. The standard way of presentations of groups and semigroups is the usage of generators and relations (Combinatorial Group Theory), Semigroup based cryptography consists of general cryptographical schemes defined in terms of wide classes of semigroups and their implementations for chosen semigroup families (so called platform semigroups).

In papers [19], [20] the author considers some modifications of Diffie-Hellman protocol when G is given as subgroup of affine Cremona semigroup $S(K^n)$ over finite commutative ring K of all polynomial transformations. The author assumes that each element is given in its standard form of Multivariate Cryptography. To use semigroup operation one has to compute the composition of transformations. This was an attempt to combine methods of Non Commutative Cryptography and Multivariate Cryptography.

Paper [21] suggests some usage of homomorphisms of subsemigroups of affine Cremona groups for protocols and cryptosystems which are not generalisations of Diffie-Hellman algorithm and its El Gamal type modifications. Some examples are given there, the implementations of these schemes with evaluation of densities of involved polynomial transformations are described in [22].

The aim of the current paper is to apply formal schemes of [21] to the case of transformations of variety($K^*)^n$, where K^* is multiplicative group of commutative ring $K \epsilon \{Z_m, F_q \mid m>2, q>2\}$

We present the new post quantum key exchange protocols and cryptosystems of El Gamal type of Non-commutative Cryptography which uses homomorphisms of two semigroups acting on $(K^*)^n$ (3.1-3.6) and two straightforward algorithms without the usage of homomorphisms. Hope that some of presented algorithms will be used in Post Quantum future.

2. **On Eulerian semigroup and hard computational problem.**

Let K be a finite commutative ring with the multiplicative group K^* of regular elements of the ring. We take Cartesian power ${}^nE(K) = (K^*)^n$ and consider an Eulerian semigroup ${}^nES(K)$ of transformations of kind

$x_1 \rightarrow \mu_1 x_1^{a(1,1)} x_2^{a(1,2)} \dots x_m^{a(1,n)}$,

$x_2 \rightarrow \mu_2 x_1^{a(2,1)} x_2^{a(2,2)} \dots x_m^{a(2,n)}$,

…

$x_m \rightarrow \mu_n x_1^{a(n,1)} x_2^{a(n,2)} \dots x_m^{a(n,n)}$,

where $a(i,j)$ are elements of arithmetic ring Z_d, $d=|K^*|$, $\mu_i \epsilon K^*$.

Let ${}^nEG(K)$ stand for Eulerian group of invertible transformations from ${}^nES(K)$. Simple example of element from ${}^nEG(K)$ is a written above transformation where $a(i,j)=1$ for $i \neq j$ or $i=j=1$, and $a(j,j)=2$ for $j \geq 2$. It is easy to see that the group of monomial linear transformations M_n is a

subgroup of ${}^nEG(K)$. So semigroup ${}^nES(K)$ is a highly noncommutative algebraic system. Each element from ${}^nES(K)$ can be considered as transformation of a free module K^n.

Let π and δ be two permutations on the set $\{1,2,..., n\}$. Let us consider a transformation of $(K^*)^n$, $K=Z_m$ or K= F_q and $d = |K^*|$. We define transformation ${}^AJG(\pi, \delta)$, where A is triangular matrix with positive integer entries $0 \le a(i,j) \le d$, $i \ge d$ defined by the following closed formula.

$y_{\pi(1)} = \mu_1 x_{\delta(1)}^{a(1,1)}$

$y_{\pi(2)} = \mu_2 x_{\delta(1)}^{a(2,1)} x_{\delta(2)}^{a(2,2)}$

...

$y_{\pi(n)} = \mu_n x_{\delta(1)}^{a(n,1)} x_{\delta(2)}^{a(n,2)} \dots x_{\delta(n)}^{a(n,n)}$

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

We refer to ${}^AJG(\pi, \delta)$ as Jordan transformations Gauss multiplicative transformation or simply *JG* element. It is an invertible element of ${}^nES(K)$ with the inverse of kind ${}^BJG(\delta, \pi)$ such that $a(i,i)b(i,i)=1 \pmod d$. Notice that in the case $K= Z_m$ straightforward process of computation the inverse of *JG* element is connected with the factorization problem of integer m. If n=1 and m is a product of two large p[rimes p and q the complexity of the problem is used in RSA public key algorithm. The idea to use composition of *JG elements or their generalisations with injective maps of* K^n into K^n was used in [23] ($K=Z_m$) *and* [24]($K= F_q$.).

We say that τ is *tame Eulerian element* over Z_m or F_q if it is a composition of seve${}^nE$ral Jordan Gauss multiplicative maps over commutative ring or field respectively. It is clear that τ sends variable x_i to a certain monomial term. The decomposition of τ into product of Jordan Gauss transformation allows us to find the solution of equations $\tau(x) = b$ for x from $(Z_m^*)^n$ or $(F^*_q)^m$. So tame Eulerian transformations over Z_m or F_q are special elements of ${}^nEG(Z_m)$ or ${}^nEG(F_q)$ respectively.

We refer to elements of ${}^nES(K)$ as multiplicative Cremona element. Assume that the order of K is constant. As it follows from definition the computation of the value of element from ${}^nES(K)$ on the given element of K^n is estimated by $O(n^2)$. The product of two multiplicative Cremona elements can be computed in time $O(n^4)$.

We are not discussing here the complexity **o**f computing the inverse for general element $g \in {}^nEG(K)$ on Turing machine or Quantum computer and problem finding the inverse for tame Eulerian elements.

Remark. Let G be a subgroup of ${}^nEG(K)$, $K \in \{Z_m, F_q\}$ generated by Jordan-Gauss elements $g_1, g_2, \dots, g_t$. The *word problem* of finding the decomposition of $g \in G$ into product of generator g_i is difficult, i. e. polynomial algorithms to solve it with Turing machine or Quantum Computer are unknown. If word problem is solved and the inverses of g_i is computable then the inverse of g is determined. Notice that if $n=1$, $K=Z_m$, , $m=pq$ where p and q are large primes and G is generated by $g_1 = \mu g_1^a$ the problem is unsolvable with Turing machine but it can be solved with Quantum Computer.

Each element of the semigroup ${}^nES(K)$ is written in the chosen basis $e_1, e_2, \dots, e_n$.

Let $J=\{i(1), i(2), \dots, i(k)\}$ be a subset of $\{1,2,..,n\}$ and $W_J = \langle e_{i(1)}, e_{i(2)}, \dots, e_{i(k)} \rangle$ be a corresponding symplectic subspace . We refer to totality ${}^nP_J(K)$ of maps $F \in {}^nES(K)$ preserving W_J as parabolic semigroup of ${}^nES(K)$. The map F from ${}^nP_J(K)$ transforms tuple $(x_{i(1)}, x_{i(2)}, \dots, x_{i(n)})$ according to the rule $x_{i(1)} \to \mu_{i(1)} x_{i(1)}^{a(1,1)} x_{i(2)}^{a(1,2)} \dots x_{i(k)}^{a(1,k)}$,

$x_{i(2)} \to \mu_{i(2)} x_{i(1)}^{a(2,1)} x_{i(2)}^{a(2,2)} \dots x_{i(k)}^{a(2,k)}, \dots, x_{i(k)} \to \mu_{i(k)} x_{i(1)}^{a(k,1)} x_{i(2)}^{a(k,2)} \dots x_{i(k)}^{a(2,k)}$.

Let π_J be the restriction of element F from ${}^nP_J(K)$ onto W_J. The map π_J defines canonical homomorphism of ${}^nP_J(K)$ onto ${}^kES(K)$. If Q is extension of K we can consider semigroup ${}^nP_{J,K}(Q)$ of maps from ${}^nES(\mathrm{Q})$ transforming $(x_{i(1)}, x_{i(2)}, \ldots, x_{i(n)})$ according to written above rule. The restriction of map $F \epsilon {}^nP_{J,K}(Q)$ on W_J defines homomorphism $\pi_{J,K}$ from ${}^nP_{J,K}(Q)$ onto ${}^kES(K)$.

3. Protocols and cryptosystems in terms of semigroup ${}^nES(K)$.

Let us consider some protocols and cryptosystems based on the idea of a hidden canonical homomorphism. Notice that if commutative ring K' is an extension of K then embedding of K into K' defines canonical embedding of ${}^nES(K)$ into ${}^nES(K')$. Let ${}^nJG(K)$ stand for the totality of all Jordan-Gauss transformations from ${}^nES(K)$.

3.1 **Tahoma protocol.**

Alice takes finite extensions Q and R of $K \epsilon \{Z_{m.}, F_q\}$ and J of cardinality k and consider a *zigzag diagram*

$$\begin{array}{ccc} {}^nP_{J,K}(Q) & \rightarrow & {}^nES(\mathrm{Q}) \\ & & \downarrow \\ {}^kES(R) & \leftarrow & {}^kES(K) \end{array}$$

The horizontal arrows correspond to embeddings of semigroups, vertical arrow corresponds to $\pi_{J,K.}$ We assume that $K=Q=R$ in the case of $K=Z_{m.}$ and R and Q are finite fields in the case of $K=F_q$. Alice takes elements $h_1, h_2, \ldots, h_s$ from ${}^kES(K)$ and creates elements $ext(h_i)$ from their $\pi_{J,K.}$reimages via adding the rules $x_j \rightarrow \mu_j x_1^{a(j,1)} x_2^{a(j,2)} \ldots x_n^{a(j,n)}$ where $\mu_j \epsilon Q^*$ and j is not an element of J. She selects set $S=\{g_1, g_2, \ldots, g_t\}$ of Jordan-Gauss elements g_i, $i=1,2, \ldots,t$ in ${}^nES(\mathrm{Q})$ and word in alphabet S to form tame element w of subgroup $G=<S>$ of ${}^nES(Q)$ together with w^{-1}. Similarly Alice takes Jordan Gauss generators $S'=\{u_1, u_2, \ldots, u_r\}$ in ${}^kES(R)$, selects word in alphabet S' and forms tame element $u \epsilon <S'>$ and its inverse u^{-1}. She forms pairs $(a_i = w^{-1} ext(h_i) w,\ b_i = u^{-1}(h_i)u),\ i=1, 2, \ldots, s$ and sends them to Bob. He takes formal alphabet $Z=\{z_1, z_2, \ldots, z_s\}$ and writes word $w_B=v(z_1, z_2, \ldots, z_s)$ in Z of length d, $d>s$ and computes specialization $z_i=a_i$ and $z_i=b_i$ and takes resulting elements $a=v(a_1, a_2, \ldots, a_s) \epsilon {}^nES(Q)$ and $b=v(b_1, b_2, \ldots, b_s) \epsilon {}^kES(R)$ respectively. Bob keeps b for himself and sends a to Alice.

Alice computes ${}^1a = wa\, w^{-1}$. She takes ${}^2a = \pi_{J,K.}({}^1a)$ and obtains collision element b as $u^{-1}({}^2a)u$.

3.2 Inverse Tahoma protocol.

As in previous protocol Alice works with presented above zigzag diagram**.** She selects sets of Jordan Gauss generators S in ${}^nES(Q)$ and S' in ${}^kES(R)$ to construct pairs of tame elements w, w^{-1} and u, u^{-1}. Now she takes set ${}^1S$ of Jordan Gauss elements over R from ${}^kES(K) \cap {}^kJG(R)$ and forms elements $h_1, h_2, \ldots, h_s$ from $<{}^1S>$ and their inverses $h_1^{-1} h_2^{-1}, \ldots, h_s^{-1}$ in ${}^kEG(R)$. Notice that elements h_i^{-1}, $i=1,2,\ldots,s$ are elements of ${}^kES(\mathrm{K})$ and larger semigroups ${}^kES(\mathrm{R})$ and ${}^kES(\mathrm{Q})$.

Alice forms $ext(h_i)$ in ${}^nES(\mathrm{Q})$. In the new algorithm she computes pairs $(a_i = w^{-1} ext(h_i) w,\ b_i = u^{-1}(h_i^{-1})u),\ i=1, 2, \ldots, s$ and sends them to Bob.

He takes formal alphabet $Z=\{z_1, z_2, \ldots, z_s\}$ and writes word $w_B=v(z_1, z_2, \ldots, z_s)=(u_1, u_2, \ldots, u_d)$ in Z of length d, $d>s$ together with the *reverse* word $Rev(w_B)=(u_d, u_{d-1}, \ldots, u_1)$. Bob computes the specialization $z_i=a_i$ of word w_B and $z_i=b_i$ of word $Rev(w_B)$ and takes resulting elements $a=v(a_1, a_2, \ldots, a_s) \epsilon {}^nES(Q)$ and $b=v(b_1, b_2, \ldots, b_s) \epsilon {}^kES(R)$ respectively

Notice that bϵ $^{k}EG(R)$. He sends a to Alice and keeps b for himself. Alice computes $^{1}a=wa\,w^{-1}$. She takes $^{2}a=\pi_{J,K.}(\,^{1}a\,)$ and obtains element b^{-1} as $u^{-\,1}(\,^{2}a\,)$u.

Remark. Alice and Bob can securely communicate in the following way. Alice writes message as a string of characters $(p_1, p_2\,, \ldots, p_k\,)$ in alphabet R^* encrypts it by application of b^{-1}. Bob decrypts it with his transformation b.

Similarly Bob uses b for the encryption of his message from the plainspace $(R^*)^k$ and Alice decrypts it with b^{-1}.

3. 4. Group enveloped Diffie- Hellman key exchange protocol.

As in the inverse protocol of the previous unit Alice works with presented above zigzag diagram**.** She selects sets. For simplicity assume that $Q=K=R$. Alice selects sets of Jordan Gauss generators S in $^{n}ES(K)$ and S' in $^{k}ES(K)$ to construct pairs of tame elements w, w^{-1} and u, u^{-1}. Now she takes set ^{1}S of Jordan Gauss elements over K from $^{k}ES(K)$ and forms elements $h_1, h_2\,, \ldots, h_s$ from $<^{1}S>$ and their inverses $h_1^{-1}\, h_2^{-1}, \ldots, h_s^{-1}$ in $^{k}EG(K)$. Alice takes $g\epsilon\,^{k}ES(K)$ and positive integer parameter k_A. Alice creates elements $ext(h_i)$, $ext(\,h_i^{-1})$ and $ext(g)$ from their $\pi_{J\,}$.reimages via adding the rules $x_j \rightarrow \mu_j x_1^{\,a(j,1)}\, x_2^{\,a(j,2)}\, \ldots\, x_n^{\,a(j,n)}$ where $\mu_j \epsilon\, K^*$ and j is not an element of J. She forms pairs $(a_i = w^{-1}ext(h_i)w,\ b_i = u^{-1}(h_i)\,u),\ i=1, 2, \ldots, s$ and sends them to Bob together with pairs (a_i^{-1}, b_i^{-1}), $g_A = u^{-1}g^{l}\,u$, $l= k_A$ and $g'=w^{-1}ext(g)w$.

Bob takes formal alphabet $Z=\{z_1, z_2\,, \ldots, z_s\}$ and writes word $w_B=v(z_1, z_2\,, \ldots, z_s)= (u_1, u_2\,, \ldots, u_d\,)$ in Z of length d, $d>s$ together with the *reverse* word $Rev(w_B)=(u_d, u_{d-1}\,, \ldots, u_1\,)$. Bob computes the specialization $z_i=a_i$ of word w_B and $z_i = a_i^{-1}$ of $Rev(w_B)$ and writes resulting elements a and a^{-1} from $^{n}ES(K)$. Similarly he creates b and b^{-1} via specialization $z_i=b_i$ of w_B and specialization and $z_i=b_i^{\,-1}$ of word $Rev(w_B)$ in the group $^{k}EG(K)$ respectively. Bob takes his natural integer k_B. He computes $^{B}g=a^{-1}\,g^{d}a,\ d=k_B$ and sends it to Alice and keeps the collision map $c=b^{-1}g_A\,^{d}b,\ d=k_B$. Alice computes collision map as $u^{-1}(\pi_J\,(w\,^{B}g\,w^{-1}))^{\,l}\,u,\ l= k_A$.
Remark. Adversary has to decompose ^{B}g into a_i and g'. After that he/she has to substitute g_A instead of g' and b_i instead of a_i.

3. 5. The inverse version of group enveloped Diffie-Hellman key exchange protocol.

Assume that $K=R=Q$ and Alice works with the simplified zigzag diagram $^{k}ES(R)= {}^{k}ES(K)$. She forms the same data as in the case of 3.4 but $g\epsilon^{k}ES(K)$ has to be invertible. So Alice takes additional set ^{2}S *of* Jordan-Gauss elements from $^{k}EG(K)$ and forms pair of kind (g, g^{-1}), $g\epsilon<^{2}S>$. She sends to Bob pairs (a_i^{-1}, b_i^{-1}), $g_A = u^{-1}g^{l}\,u$, $l= k_A$ and $g^*=w^{-1}ext(g^{-1})w$ instead of g' of 3.4.

Bob uses word in the alphabet of formal variables and generates elements a and a^{-1} from $^{n}EG(K)$ and b, $b^{-1}\,\epsilon\,^{k}EG(K)$ in the same way with the case of 3.4 and takes his natural integer k_B. Now he computes $^{B}g=a^{-1}\,g^{*d}a,\ d=k_B$ and sends it to Alice and keeps the map $f=b^{-1}g_A\,^{d}b,\ d=k_B$. Alice computes the inverse map for f as $u^{-1}(\pi_J\,(w\,^{B}g\,w^{-1}))^{\,l}\,u,\ l= k_A$.

Remark. Alice and Bob have bijective transformations f and f^{-1} of the variety K^*. So they can exchange messages written in alphabet.

3.6. Inverse two wheeled Diffie-Hellman protocol.

Alice takes semigroup ${}^nES(K)$ and some partition J_1 , J_2 of $N=\{1.2,...,n\}$, $|J_1|=t$ and $|J_2|=d$, $t+d=n$. So the intersection of J_1 and J_2 is an empty set and the union of these sets equals N. She takes ${}^nP_{J1}(K)\cap{}^nP_{J2}(K)$ which is direct product of semigroups $E_1={}^tES(K)$ and $E_2={}^dES(K)$.

Alice selects set S of Jordan Gauss elements $g_1, g_2,\ldots, g_r$ of ${}^nEG(K)$, computes their inverses and generates tame element $g \in G$, $G=<g_1, g_2,\ldots, g_r>$.
She takes parameter k_A, computes g^l , $l=k_A$ together with g^{-1}. Alice selects Jordan-Gauss elements $e_1, e_2,\ldots, e_s$ of E_1 and forms elements u and u^{-1} of E_1 . She sends $g_A = u^{-1}g^l u$ and $u^{-1}g^{-1}u = g'$ to Bob together with partition J_1 , J_2 .

Bob selects Jordan Gauss elements $f_1, f_2,\ldots, f_p$. He generates pair v, v^{-1} of tame elements from E_2. Bob choses his parameter k_B. He creates $g_B = v^{-1}g'^{1}v$, $l=k_B$ and sends it to Alice, but keeps for himself ${}^Bg=v^{-1}g_A{}^1v$. Alice computes the inverse ${}^Ag$ for ${}^Bg$ as $u^{-1}\, g_B{}^1 u$, $l=k_A$.

3. 7. Two wheeled Diffie-Hellman protocol.

Alice and Bob share the subgroup ${}^nES(K)$ together with the partition J_1, J_2. She has a free choice to take any g from ${}^nES(K)$. Alice and Bob form invertible elements u and v from E_1 and E_2 similarly to previous algorithm and their positive integers k_A and k_B.

The scheme is simple . Alice and Bob exchange $g_A = u^{-1}g^l u$, $l=k_A$ and $g_B == v^{-1}g^t v$, $t=k_B$. After that she computes collision element as ${}^Ag = u^{-1}\, g_B{}^{tl}\, u$ and he compute this element as ${}^Bg = v^{-1}\, g_A{}^{lt}\, v$.

Remark. Adversary has to find a decomposition of g_B into the triple of kind $x^{-1}g^y x$, $x \in E1$ **(**or decompose g_A into the triple of kind $x^{-1}g^y x$, $x \in E2$**)**
We can take platform ${}^nES(K)$ and consider the following known key exchange scheme.

3. 8. Twisted Diffie-Hellman protocol.

Alice and Bob share element $g \in {}^nES(K)$ and pair of tame elements h, h^{-1} from ${}^nEG(K)$.
Alice takes positive integer $t=k_A$ and $d=r_A$ and forms $h^{-d}g^t h^d = g_A$**.** Bob takes $s=k_B$ and $p=r_B$. and forms $h^{-p}g^s h^p = g_B$**.** They exchange g_A and g_B and compute collision element as ${}^Ag = h^{-d}g_B{}^t h^d$ and ${}^Bg = h^{-p}g_B{}^t h^s\, h^p$ respectively.

3. 8. Inverse twisted Diffie-Hellman protocol.

Correspondents follow the scheme 3.8 with the tame element $g \in {}^nEG(K)$ and Alice sends $h^{-d}g^{-t}h^d = g_A$ to Bob and she gets $h^{-p}g^s h^p = g_B$ from him. They use the same formulae for ${}^Ag$ and ${}^Bg$. But in the new version these elements are mutual inverses.

REFERENCES

[1] J. Ding., J.E. Gower and D.S. Schmidt., Multivariate Public Key Cryptosystems, 260. Springer, Advances in Information Security, v. 25, (2006).

[2] N. Koblitz, Algebraic aspects of cryptography, Springer (1998)., 206 P.

[3] L. Goubin, J.Patarin and Bo-Yin Yang, Multivariate Cryptography. Encyclopedia of Cryptography and Security, (2nd Ed.) 2011, 824-828.

[4] D. N. Moldovyan and N.A. Moldovyan, A New Hard Problem over Non-commutative Finite Groups for Cryptographic Protocols, International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security, MMM-ACNS 2010: Computer Network Security pp 183-194.

[5] L. Sakalauskas., P. Tvarijonas and A. Raulynaitis, Key Agreement Protocol (KAP) Using Conjugacy and Discrete Logarithm Problema in Group Representation Level}, INFORMATICA, 2007, vol. !8, No 1, 115-124.
[6] V. Shpilrain, A. Ushakov,The conjugacy search problem in public key cryptography: unnecessary and insufficient,Applicable Algebra in Engineering, Communication and Computing, August 2006, Volume 17, Issue 3–4, pp 285–289.
[7] Delaram Kahrobaei and Bilal Khan, A non-commutative generalization of ElGamal key exchange using polycyclic groups, In IEEE GLOBECOM 2006 - 2006 Global Telecommunications Conference [4150920] DOI: 10.1109/GLOCOM.2006.
[8] Alexei Myasnikov; Vladimir Shpilrain and Alexander Ushakov (2008). Group-based Cryptography. Berlin: Birkhäuser Verlag.
[9] Zhenfu Cao (2012). New Directions of Modern Cryptography. Boca Raton: CRC Press, Taylor & Francis Group. ISBN 978-1-4665-0140-9.
[10] Benjamin Fine, et. al. "Aspects of Non abelian Group Based Cryptography: A Survey and Open Problems". arXiv:1103.4093.
[11] Alexei G. Myasnikov; Vladimir Shpilrain and Alexander Ushakov (2011). Non-commutative Cryptography and Complexity of Group-theoretic Problems. American Mathematical Society.
[12] I. Anshel, M. Anshel and D. Goldfeld: An algebraic method for public-key cryptography. Math. Res.Lett. 6(3–4), 287–291 (1999).
[13] S.R. Blackburn and S.D. Galbraith: Cryptanalysis of two cryptosystems based on group actions. In: Advances in Cryptology—ASIACRYPT ’99. Lecture Notes in Computer Science, vol. 1716, pp. 52–61. Springer, Berlin (1999).
[14] K.H. Ko, S.J. Lee, J.H. Cheon, J.W. Han, J.S. Kang and C. Park: New public-key cryptosystem using braid groups. In: Advances in Cryptology—CRYPTO 2000, Santa Barbara, CA. Lecture Notes in Computer Science, vol. 1880, pp. 166–183. Springer, Berlin (2000)
[15] G. Maze, C. Monico and Rosenthal, J.: Public key cryptography based on semigroup actions. Adv.Math. Commun. **1**(4), 489–507 (2007)
[16] P.H. Kropholler and S.J. Pride , W.A.M. Othman K.B. Wong, P.C. Wong, Properties of certain semigroups and their potential as platforms for cryptosystems, Semigroup Forum (2010) 81: 172–186
[17] J.A. Lopez Ramos, J. Rosenthal, D. Schipani and R. Schnyder, Group key management based on semigroup actions, Journal of Algebra and its applications, vol.16 (to appear in 2019).
[18] Gautam Kumar and Hemraj Saini, Novel Noncommutative Cryptography Scheme Using Extra Special Group, Security and Communication Networks ,Volume 2017, Article ID 9036382, 21 pages, https://doi.org/10.1155/2017/9036382
[19] V. Ustimenko, On the families of stable transformations of large order and their cryptographical applications, Tatra Mt. Math. Publ., 70 (2017), 107-117.
[20] V. Ustimenko, On desynchronised multivariate El Gamal algorithm, Cryptology ePrint Archive, 712, 2017.
[21] V. Ustimenko, On new symbolic key exchange protocols and cryptosystems based on hidden tame homomorphism,. *Dopov. Nac. akad. nauk Ukraine,2018, n 10, pp.26-36.*
[22] V.Ustimenko, M. Klisowski , On Noncommutative Cryptography with cubical multivariate maps of predictable density, Proceedings of ‘’Computing 2019’’ conference, London, 16-17, July (to appear).
[23]. V. Ustimenko, On new multivariate cryptosystems based on hidden Eulerian equations, Dopov. Nath Acad of Sci, Ukraine, 2017. № 5, pp 17-24.
[24] V. Ustimenko, On new multivariate cryptosystems based on hidden Eulerian equations over finite fields, Cryptology ePrint Archive, 093, 2017.