Hindawi
Security and Communication Networks
Volume 2019, Article ID 2137561, 15 pages
https://doi.org/10.1155/2019/2137561

Research Article

On the Constructions of New Symmetric Ciphers Based on Nonbijective Multivariate Maps of Prescribed Degree

Vasyl Ustimenko[1], **Urszula Romańczuk-Polubiec**[2], **Aneta Wróblewska**[1], **Monika Katarzyna Polak**[3] **and Eustrat Zhupa**[4]

[1]*Maria Curie-Skłodowska University, Poland*
[2]*Independent Researcher, Poland*
[3]*Rochester Institute of Technology, USA*
[4]*University of Rochester, USA*

Correspondence should be addressed to Urszula Romańczuk-Polubiec; urszula_romanczuk@yahoo.pl

Received 12 October 2018; Revised 31 December 2018; Accepted 21 February 2019; Published 1 April 2019

Guest Editor: Pawel Szalachowski

The main purpose of this paper is to introduce stream ciphers with the nonbijective encryption function of multivariate nature constructed in terms of algebraic graph theory. More precisely, we describe the two main symmetric algorithms for creation of multivariate encryption transformations based on three families of bipartite graphs with partition sets isomorphic to $\mathbb{K}^n$, where $\mathbb{K}$ is selected as the finite commutative ring. The plainspace of the algorithm is $\Omega = \{x \mid \sum x_i \in \mathbb{K}^*,\ x \in \mathbb{K}^n\} \subset \mathbb{K}^n, \Omega \cong \mathbb{K}^* \times \mathbb{K}^{n-1}$. The second algorithm is a generalization of the first one with using the jump operator, where generalized encryption map has an essentially higher degree in comparison with the previous version. Moreover, the degree of this generalized map is not bounded by some constant. This property guarantees resistance of the cipher to linearization attacks.

1. Introduction

This paper is an extension of article [1] reflecting our talk at the 5th International Conference on Cryptography and Security Systems (one of the events of Federated Conference on Computer Science and Information Systems, 2018). We expand our work by adding the generalization of our symmetric cipher of multivariate nature. Generalized encryption map has essentially higher degree in comparison with previous version. The degree is not bounded by some constant. This property insures resistance of the cipher to linearization attacks.

Graph theory is applicating in diverse fields such as linguistics, biochemistry, coding theory, cryptography, communication networks, etc. The history of the use of sparse algebraic graphs in symmetric cryptographical algorithms was described in [1] with the full list of references which begins with the ideas of V. Ustimenko presented in the article from 1998 (see [2]). In this paper, we present only a short version of this history related to work of V. Ustimenko and his team. The reader can find also more general survey on some applications of Graph Theory in Cryptography in [3].

The following known graphs defined over finite commutative ring $\mathbb{K}$ were used: $D(n, \mathbb{K})$ (see [2]; for $\mathbb{K} = \mathbb{F}_q$ graphs were defined in [4]), $W(n, \mathbb{K})$ (Wenger graphs defined in [5]), graphs $A(n, \mathbb{K})$ introduced in [6], and graphs $\widetilde{D(n, \mathbb{K})}$ of [7]. Popular choices of $\mathbb{K}$ are finite fields $\mathbb{F}_{127}$, $\mathbb{F}_{2^7}$, $\mathbb{F}_{2^8}$, $\mathbb{F}_{2^{16}}$, and $\mathbb{F}_2^{32}$ and rings modular arithmetic $\mathbb{Z}_{2^7}$, $\mathbb{Z}_{2^8}$, and $\mathbb{Z}_{2^{16}}$. This research history is presented in the next section. Section 2 observes graph based stream ciphers which use bijective encryption function of multivariate nature. In fact, multivariate cryptography uses nonbijective maps and a private key decryption is also given in each case of this type. However, the vast majority of stream ciphers is defined via bijective encryption. In each case of nonbijective symmetric encryption there is a deterministic decryption process that has to be described in clear way. In Section 3 we discuss the class of nonbijective multivariate maps defined in terms of Euler theorem for arithmetical rings $\mathbb{Z}_m$, where m is a composite number. Such a map F has a special subset D

(domain) of affine space $\mathbb{Z}_m^n$ isomorphic to $\mathbb{Z}_m^* \times \mathbb{Z}_m^{n-1}$, such that the restriction $F|_D$ of F onto D is injective ($|D| = |F(D)|$). It is important that $|F(D)|$ is unknown to the adversary, who knows only cipherspace $\mathbb{Z}_m^n$. Correspondents use their knowledge on the password to obtain a description of $F(D)$. They are able to compute bijective map F_{dec} from $F(D)$ onto D which is a decryption procedure; i.e., the composition of $F|_D$ with F_{dec} is identity map on D. The definition of multivariate nonlinear map $F : \mathbb{Z}_m^n \longrightarrow \mathbb{Z}_m^n$ uses Eulerian map $x \longrightarrow ax^e + b$ of $\mathbb{Z}_m$ with $a \in \mathbb{Z}_m^*$ and $gcd(e, \phi(m)) = 1$, where ϕ is Euler function.

An adversary does not have access to a, e, or b. A hidden Eulerian equation of kind $x^e = c$ gives a heuristic support to resistance of symmetric algorithm to attacks with interception of pairs plaintext/ciphertext. Notice that, in the case of classical RSA algorithm large m decomposable into two primes is known, parameter e is given. Security of RSA rests on the complexity of finding decomposition $m = pq$ or finding multiplicative inverse d of e. In practical cases of multivariate encryption with hidden Eulerian equation (like $m = 2^k$) Eulerian function $\phi(m)$ is easy to compute, but multiplicative inverse of e is hard to find because the adversary simply does not know e. In fact, in our examples the degree $\deg(F) = \alpha(e, n)$ of multivariate map F on $\mathbb{Z}_m^n$ sending D to codomain $D' = F(D)$ heavily depends on parameter e. Decryption map of D' into D is induced by nonbijective multivariate map $G : \mathbb{Z}_m^n \longrightarrow \mathbb{Z}_m^n$ and degree $d'(n, d)$ of degree $\deg(G) = \beta(d, n)$. So "Eulerian parameters" e and d are very essential for cryptanalysis. This approach is illustrated by 3 "toy examples" of graph based symbolic computations (see (G) at the end of Section 3 with items (G.1), (G.2), and (G.3)).

A linearization attack has to disclose codomain $D' = F(D)$, determine the standard form of polynomial encryption map F with its degree, and construct a polynomial map G, such that $FG|_D$ is identity function. If one of the parameters $\alpha(e, n)$ or $\beta(d, n)$ is unbounded (or it is a large constant) then linearization tasks are infeasible. Of course, a cryptanalyst has to try alternative approaches like distinguish, Time Memory Data trade-off attacks, and guess-and-determine attack. These methods are constructed for investigation of stream ciphers with bijective encryption. Authors believe that in the case of multivariate nonbijective encryption such attacks have to be seriously modified for the practical implementation.

This approach is illustrated by three "toy examples" of graph based encryption. In Section 4 we introduce a class of bivariate graphs containing all the above-mentioned graphs. Such concept is convenient for uniform description of the encryption scheme and observation of common properties of graphs from this class (Sections 5 and 8). General Algorithm No. 1 is described in Section 5 presents symmetric cipher based on nonbijective maps. Implementation results are presented in Table 1. The Algorithm No. 2, presented in Section 8, is a generalization of the first one with using the jump operator, where generalized encryption map has an essentially higher degree in comparison with the previous version. Moreover, the degree of this generalized map is not bounded by some constant. This property guarantees resistance of the cipher to linearization attacks. We compare graphs and related algorithms corresponding to different families ($W(n, \mathbb{K})$, $D(n, \mathbb{K})$, $A(n, \mathbb{K})$, and $\widetilde{D(n, \mathbb{K})}$) in Sections 6 and 9.

Last section is the conclusion where we discuss the choice of our model. Here the reader can find remarks on multivariate cryptography and its connections with cryptographical applications of algebraic graph theory.

RSA is one of the most popular cryptosystems. It is based on a number factorization problem and on Euler's Theorem. Peter Shor discovered that the factorization problem can be effectively solved by using a theoretical quantum computer. This means RSA could not be a security tool in the future postquantum era. One of the research directions leading to a postquantum secure public key is the multivariate cryptography, which uses a polynomial maps of affine space $\mathbb{K}^n$ defined over a finite commutative ring $\mathbb{K}$ into itself as encryption tools (see [8]). This is a young promising research area because of the current lack of known cryptosystems with the proven resistance against attacks with the use of Turing machines. Another important direction of Postquantum Cryptography is the study of Hyperelliptic Curves Cryptosystems. We have to say that classical elliptic curves encryption will be not secure in the postquantum era.

Applications of algebraic graphs to cryptography started with symmetric algorithms based on explicit constructions of extremal graph theory and their directed analogues. The main idea is to convert an algebraic graph in a finite automaton and to use the pseudorandom walks on the graph as encryption tools. This approach can also be used for the key exchange protocols. Nowadays the idea of "symbolic walks" on algebraic graphs, when the walk on the graph depends on parameters given as special multivariate polynomials in variables depending from plainspace vector, appears in several public key cryptosystems.

Multivariate cryptography started from the study of potential for the special quadratic encryption multivariate bijective map of $\mathbb{K}^n$, where $\mathbb{K}$ is an extension of finite field $\mathbb{F}_q$ of characteristic 2. One of the first such cryptosystems was proposed by Imai and Matsumoto and cryptanalysis for that system was invented by J. Patarin. A survey on various modifications of this algorithm and corresponding cryptanalysis can be found in [8] or [9].

One of the first uses of nonbijective map of multivariate cryptography was in the *oil and vinegar* cryptosystem proposed in [10] and analyzed in [11]. Nowadays, this general idea is strongly supported by publication [12] devoted to security analysis of direct attacks on modified unbalanced oil and vinegar systems. It looks like such systems and rainbow signature schemes may lead to promising Public Key Schemes of Multivariate Encryption defined over finite fields. Nonbijective multivariate sparse encryption maps of degree 3 and $\geq$ 3 based on walks on algebraic graphs $D(n, \mathbb{K})$ defined over general commutative ring and their homomorphic images were proposed in [13]. Security of the corresponding cryptosystem rests on the idea of hidden discrete logarithm problem. U. Romańczuk-Polubiec and V. Ustimenko combine the idea of "oil and vinegar signature

Table 1: Encoding and decoding time.

		$A(n,\mathbb{K})$		$D(n,\mathbb{K})$		$\widetilde{D(n,\mathbb{K})}$	
Password	File size	Enc	Dec	Enc	Dec	Enc	Dec
	1K	0.0021	0.0029	0.0030	0.0026	0.0039	0.0041
	10K	0.0217	0.0253	0.0234	0.0249	0.0322	0.0366
	50K	0.1030	0.1338	0.1034	0.1423	0.1572	0.1859
3	100K	0.2158	0.2701	0.2115	0.2683	0.3309	0.3800
	500K	1.2202	1.3863	1.0432	1.3556	1.6161	1.9323
	1M	2.1955	2.8346	2.1452	2.7285	3.2809	3.9029
	10M	21.9597	27.4227	21.3803	26.6821	32.8819	38.3860
	1K	0.0416	0.0033	0.0400	0.0032	0.0401	0.0047
	10K	0.0311	0.0320	0.0302	0.0360	0.0420	0.0466
	50K	0.1393	0.1639	0.1374	0.1580	0.2125	0.2366
4	100K	0.2800	0.3314	0.2738	0.3280	0.4259	0.4816
	500K	1.4381	1.7109	1.3918	1.6541	2.1278	2.4159
	1M	2.9271	3.5035	2.8457	3.4055	4.3633	4.9664
	10M	29.5728	34.6022	28.6899	33.7773	43.7334	49.4341
	1K	0.0402	0.0045	0.0336	0.0039	0.0437	0.0058
	10K	0.0355	0.0395	0.0382	0.0440	0.0533	0.0596
	50K	0.1764	0.2038	0.1718	0.1909	0.2589	0.2876
5	100K	0.3510	0.4097	0.3391	0.3922	0.5243	0.5781
	500K	1.7778	2.0589	1.7237	2.0015	2.7088	3.0049
	1M	3.6421	4.2418	3.5507	4.1302	5.4671	6.0630
	10M	37.3170	42.0697	36.2427	40.9556	55.1103	60.4248
	1K	0.0445	0.0053	0.0412	0.0046	0.0445	0.0069
	10K	0.0426	0.0481	0.0453	0.0448	0.0705	0.0667
	50K	0.2132	0.2371	0.1987	0.2325	0.3123	0.3462
6	100K	0.4176	0.4830	0.4069	0.4678	0.6303	0.6890
	500K	2.1494	2.4572	2.0897	2.3724	3.2690	3.5826
	1M	4.3851	4.9386	4.2630	4.8109	6.7762	7.2091
	10M	47.8490	50.3557	42.6451	47.7372	65.8464	71.6511
	1K	0.0434	0.0055	0.0435	0.0059	0.0487	0.0091
	10K	0.0477	0.0540	0.0475	0.0533	0.0754	0.0848
	50K	0.2437	0.2699	0.2324	0.2671	0.3651	0.3979
7	100K	0.4903	0.5457	0.4751	0.5275	0.7315	0.7938
	500K	2.5089	2.8124	2.5655	2.7524	3.7086	4.0025
	1M	5.0959	5.7679	5.1230	5.6692	7.5859	8.2276
	10M	51.0014	56.3961	49.8712	54.9345	76.4318	87.4684

cryptosystem" with the idea of linguistic graph based map with partially invertible decomposition to introduce a new cryptosystem [13]. This algorithm can be implemented with the use of families $D(n,\mathbb{K})$ and $A(n,\mathbb{K})$ and natural homomorphism between them. Finally, in [14] "hidden RSA multivariate encryption" based on graphs $D(n,\mathbb{K})$ were proposed.

In this paper we modify the encryption map (private key) of the above-mentioned cryptosystem in terms of family of bivariate graphs defined over the commutative ring $\mathbb{K}$. These maps have multivariate nature despite the "numerical implementation" in symmetric ciphers mode with the plainspace isomorphic to $\mathbb{K}^* \times \mathbb{K}^{n-1}$.

2. Implementation of Algorithms Based on Bijective Maps

We worked on a software package that enables us to investigate strongly symmetric cases of stream ciphers based on graphs $W(n,\mathbb{K})$, $D(n,\mathbb{K})$, $\widetilde{D(n,\mathbb{K})}$, and $A(n,\mathbb{K})$, where $\mathbb{K}$ is the arithmetic ring. Some cases are already implemented by our team at the level of prototype model.

Few algorithms have been implemented in the past for very special cases under supervision of V. Ustimenko. The history of implementation of these algorithms was described in [1] with the full list of references. Below we present only a short version of this history.

The first implementation of $D(n, \mathbb{K})$ encryption was done in 2000 at the University of South Pacific (USP, Fiji Islands). The research team was composed by Professor V. Ustimenko, PhD Dharmendra Sharma (currently professor of University of Canberra), and postgraduate students V. Gounder and R. Prasad. The work was supported by the University Research Committee of the University of South Pacific (USP) grant. During this work the implementation of asymmetric mode was investigated with the chosen case for $\mathbb{K}$ was $\mathbb{F}_{127}$, with 127 being the closest prime to the size of ASCII code alphabet. It means that one has to delete just the *delete* service symbol and can encrypt arbitrary text files. The chosen string was $\alpha_i(x) = x + d_i$, where d_i are elements of the ring $\mathbb{K} = \mathbb{F}_{127}$ chosen in pseudorandom fashion. So that was a case of shifting encryption.

The affine transformations L_1 and L_2 were simply identities. The implemented cipher on ordinary PC was rather robust in performance, but with average mixing properties. It has been used at USP digital network working for campuses and USP centers located in 11 island countries of South Pacific region. The package was also used by ORACLE based system of the bursary office. Recently group of students from Okanagan College (affiliated with the University of British Columbia) implemented that stream cipher on a cluster network of PC for a large data encryption.

Another case with $\mathbb{K} = \mathbb{Z}_{256}$ and graph $D(n, \mathbb{K})$ was implemented under the Research Committee of Sultan Qaboos University (SQU, Oman) grant. The research team was composed of Professors Vasyl Ustimenko and Abderezak Tousane and students Rahma Al Habsi and Huda Al Naamani. The software uses one to one correspondence between elements of $\mathbb{Z}_{256}$ and symbols of binary alphabet. It allows encryption of various file types (with extension doc, jpg, htm, avi, pdf, ...) in a way that encrypted file is presented in the same format as the plaintext. The symmetric algorithm was used in academic networks of SQU and Kiev Mohyla Academy.

The systematic study of shifting encryption for cases of shifting encryptions of $D(n, \mathbb{K})$ was conducted at UMCS (Lublin, Poland). J. Kotorowicz used arithmetical rings $\mathbb{Z}_2^7$, $\mathbb{Z}_2^8$, $\mathbb{Z}_2^{16}$ for the implementation with various affine transformation τ_L and τ_R (see [15]). The encryption was essentially faster than in all previously known cases. The selected affine transformation leads to an encryption with very good mixing properties: the change of a single character of the plaintext or the change of a single character of the encryption string $d_1, d_2, \ldots, d_s$ causes the change of at least 98% of the ciphertext characters. In the case of $\tau_R = \tau_L^{-1}$ it can be proved that the order of $A(n, \mathbb{K})$ and $D(n, \mathbb{K})$ based encryption map grows with the growth of parameter n. The comparison of orders was completed through the study of cycle structures of $A(n, \mathbb{K})$ and $D(n, \mathbb{K})$ encryptions. The obtained results showed similarity in both cases.

M. Klisowski implemented $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$ shifting encryption on symbolic level in the cases of finite fields $\mathbb{F}_{2^7}$, $\mathbb{F}_{2^8}$, $\mathbb{F}_{2^{16}}\mathbb{F}_{2^{32}}$. In [16] A. Wróblewska proved that shifting $D(n, \mathbb{K})$ encryption is given by a cubical multivariate map (see also [17]). Computer simulation results allow estimating time of generation of these maps as functions of parameter n and densities of such multivariate cubic encryption and decryption maps. Similar results for cases of Boolean rings of sizes 2^7, 2^8, 2^{16}, and 2^{32} are obtained via computer simulations.

The PhD thesis of M. Klisowski [18] contains the first results on $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$ based multivariate maps which are not defined via shifting encryptions. He used symbolic strings of kind $\alpha_1(x) = x + c_1, \alpha_2(x) = x + c_2, \ldots, \alpha_{s-1}(x) = x + c_{s-1}$, $\alpha_s(x) = x^3 + c_s$ with constants c_i, $i = 1, 2, \ldots, s$ for special fields $\mathbb{F}_q$ in which $x^3 = b$ has unique solution. It was shown that such a choice makes direct linearization attacks impossible.

The first implementation for the case of Wenger graph based encryption was completed at the University of Sao Paolo (USP, Brasil) (see [19] and further references). Professors V. Futorny and V. Ustimenko chose field $\mathbb{F}_{253}$ of which size is the closest from below prime to the size of binary alphabet. This research was partially supported by FAPESP foundation (grant for international cooperation with USP). Computer simulation demonstrated high speed of encryption. In [19] authors evaluated the diameter of graph $W(n, \mathbb{F}_q)$ and proved that the family of these graphs $W(n, q)$, $n \leq q$ is a family of small world graphs.

Professor Routo Terada (USP, Brasil) suggested to investigate the behaviour of these algorithms under linearization attacks. Computer simulation supports the conjecture on a good resistance of the encryption scheme to such attacks.

The idea of using graphs $A(n, \mathbb{K})$ in cryptography was proposed by U. Romańczuk-Polubiec and V. Ustimenko in [6].

Some stream ciphers defined via graphs $\widetilde{D(n, \mathbb{K})}$ were proposed by M. Polak and V. Ustymenko in [7]. Furthermore, M. Polak compared LDPC codes corresponding to $A(n, \mathbb{K})$, $D(n, \mathbb{K})$, and $\widetilde{D(n, \mathbb{K})}$ in [20].

3. On the Idea of Nonbijective Maps Based on Eulerian Equations and Some Toy Examples

Let us consider the case of commutative ring $\mathbb{Z}_m$, where m is composite number.

Classical scheme of multivariate bijection map on affine space $\mathbb{Z}_m^n$ is of the form $T_1 F T_2$, where T_1 and T_2 are bijective maps of kind

$$(x_1, x_2, \ldots, x_n) \longrightarrow A(x_1, x_2, \ldots, x_n) + (d_1, d_2, \ldots, d_n), \tag{1}$$

where $A = (a_{i,j})$, $i = 1, 2, \ldots, n$ is nonsingular matrix with entries from $\mathbb{Z}_m$ and F is a nonlinear bijective polynomial map of kind $x_i \longrightarrow f_i(x_1, x_2, \ldots, x_n)$, $i = 1, 2, \ldots, n$, $f_i \in \mathbb{Z}_m[x_1, x_2, \ldots, x_n]$. Leonard Euler investigated maps $\eta : x \longrightarrow x^e$, $x \in \mathbb{Z}_m$. He discovered that if x is a regular ($gcd(x, m) = 1$) and e is mutually prime with the order $\phi(m)$ of multiplicative group of $\mathbb{Z}_m$, then the preimage of $y = \eta(x)$ is unique. Eulerian theorem states that the solution of $\eta(x) = y$ is $x = y^r$, where d is multiplicative inverse of e in the group $\mathbb{Z}_m^*$, i.e., $ed \equiv 1 (\mathrm{mod}\, m)$. Map ϕ defined on the set of positive

integers is known as *Euler function*. Everybody knows that the security of RSA encryption rests on the Eulerian Theorem.

The general idea of hidden Eulerian equation [21] suggests to use $G = T_1FT_2$, where T_1 is monomial map $x_i \longrightarrow a_i x_{\pi(i)}$, where π is some permutation. Nonlinear map F is Eulerian map given by rule $x_i \longrightarrow a_i {x_i}^{e_i} + b_i$, $i = 1, 2, \ldots, n$, where $gcd(a_i, m) = 1$, $gcd(e_i, \phi(m)) = 1$ and $b_i \in \mathbb{Z}_m$.

It is easy to see that Eulerian map from one variable $x \longrightarrow x^e = E(x)$, $x \in \mathbb{Z}_m$ is far from being a bijection in the case of composite number. For example if $m = 8$ and $e = 3$, we have $E(0) = 0$, $E(1) = 1$, $E(2) = 0$, $E(3) = 3$, $E(4) = 0$, $E(5) = 5$, $E(6) = 0$, and $E(7) = 7$. However, we can use G as encryption function on the plainspace Ω. Notice that T_1 and F preserve the subset Ω of the affine space $\mathbb{Z}_m^n$. The restriction of composition T_1F onto Ω is a bijection. So T_1F sends plaintext $\mathrm{x} = (x_1, x_2, \ldots, x_n)$ to intermediate vector $\mathrm{v} = (v_1, v_2, \ldots, v_n)$. The image of $G(\mathrm{x})$ coincides with $T_2(\mathrm{v}) = \mathrm{y}$. For decryption, the correspondent applies ${T_2}^{-1}$ to the ciphertext to obtain v and then the plaintext as ${T_1}^{-1}(F^{-1}(\mathrm{v}))$. Notice that, for the computation of $F^{-1}(\mathrm{v})$, the correspondent has to use Euler theorem. As you see the plainspace of this scheme is smaller than cipherspace like in the well-known El Gamal method. The encryption map sends the Domain Ω onto $G(\Omega)$ of the same size, but the location of these codomains is hidden from adversaries, because the map T_2 is hidden from him/her. The encryption scheme described above can be generalized in various ways. For instance, one can define Eulerian map as $x_i \longrightarrow a_i x_{i_i}^{e} + b_i$. Some modifications are suggested in [21] where even more general definition of Eulerian map is given and nonlinear multivariate bijection is used instead of affine map T_1.

Another idea is based on deformations of bijective nonlinear multivariate map H of kind $x_1 \longrightarrow h(x_1)$, $x_i \longrightarrow h_i(x_1, x_2, \ldots, x_n)$, $i = 2, 3, \ldots, n$. We assume that correspondents are able to solve $H(\mathrm{x}) = \mathrm{c}$ for x and refer to such multivariate transformation as the map with one-dimensional invariant subspace.

We consider a map $\widetilde{H}$ of kind $x_1 \longrightarrow ax_1^e + b$, $x_2 \longrightarrow h_i(x_1, x_2, \ldots, x_n)$, $i = 2, 3, \ldots, n$, where $gcd(e, \phi(m)) = 1$ and $a \in \mathbb{Z}_m^*$.

Obviously, $\widetilde{H}$ sends $\Delta = \{(x_1, x_2, \ldots, x_n) \mid x_1 \in \mathbb{Z}_m^*, x_i \in \mathbb{Z}_m, i = 2, \ldots, n\}$ to $\widetilde{H}(\Delta)$ of the same size. We consider the encryption map G of kind $G = T_1\widetilde{H}T_2$, where $T_2 \in AGL_n(\mathbb{Z}_m)$ and T_1 is an affine map of kind $x_1 \longrightarrow x_1 + x_2 + \cdots + x_n$, $x_j \longrightarrow l_j(x_1, x_2, \ldots, x_n)$, $j = 2, 3, \ldots, n$, $l_j \in \mathbb{Z}_m[x_1, x_2, \ldots, x_n]$, $\deg(l_j) = 1$.

So correspondents can use the plainspace $\Omega = \{\mathrm{x} \in \mathbb{Z}_m^n \mid x_1 + x_2 + \cdots + x_n \in \mathbb{Z}_m^*\}$. The map T_1 sends plaintext x to the tuple $\mathrm{v} = (v_1, v_2, \ldots, v_n) \in \Delta$.

The map $\widetilde{H}$ transforms v into $\mathrm{u} = (u_1, u_2, \ldots, u_n)$ with $u_1 = a{v_1}^e + b$. Finally affine map T_2 sends u to the ciphertext y. Decryption process: correspondent computes u as ${T_2}^{-1}(\mathrm{y})$. He/she solves equation ${v_1}^e = (u_1 - b)/a$ and gets v_1. He investigates the map $v_1 \longrightarrow h(v_1)$, $v_2 \longrightarrow u_2, \ldots, v_n \longrightarrow u_n$. Under the assumption that correspondents can compute the preimages of H they can get $\mathrm{v} = (v_1, v_2, \ldots, v_n)$ and the plaintext $\mathrm{x} = {T_1}^{-1}(\mathrm{v})$.

Formally, our encryption map can be expressed as T_1HCT_2, where C is the map $x_1 \longrightarrow a_1(f^{-1}(x))^e + b$, $x_j \longrightarrow x_j$, $j = 2, 3, \ldots, n$. Clearly, representation of f^{-1} in polynomial map can be a hard task.

In order to be able to use the above encryption, Alice has to compute the standard form $E = T_1\widetilde{H}T_2$ of kind $x_i \longrightarrow e_i(x_1, x_2, \ldots, e_n)$, $i = 1, 2, \ldots, n$. She has to prove that E is computable in polynomial time. To hide parameter e Alice can select

$$H = \left(h_1(x_1, x_2, \ldots, x_n), h_2(x_1, x_2, \ldots, x_n), \ldots, h_i(x_1, x_2, \ldots, x_n)\right) \tag{2}$$

with $\deg(h_i) > e$. Notice that in RSA algorithm the ring $\mathbb{Z}_m$ and parameter e of the map $x \longrightarrow x^e$, $x \in \mathbb{Z}_m^*$ are known. So one can use term *hidden RSA* for the presented scheme of multivariate public key algorithm. It means that correspondents may use m decomposed not only into two large prime numbers, but also into other composite modules.

Let us assume that correspondents are going to use E for private key symmetric encryption. Then they do not need to compute and share the polynomial form of E. So in this case requirement $d < \deg(E)$ is immaterial. Alice and Bob keep parameters a, e, and b together with matrices T_1 and T_2 as part of their private key. They have to present for everybody the algorithm of computation for the value of H in a given point $(p_1, p_2, \ldots, p_n)$.

Of course the polynomial form of E is important to study the adversary attacks in the case of interception of many pairs of kind plaintext/ciphertxt (linearization, distinguish, TMD, and guess-and-determine attacks). Known methods of cryptanalysis dealing with bijective encryption have to be modified for attacks on nonbijective multivariate maps.

Notice that

(i) $\widetilde{H}$ defined on $\mathbb{Z}_m^n$ has one-dimensional invariant subspace of tuples of kind $(\alpha, 0, \ldots, 0)$.

(ii) The restriction of $\widetilde{H}$ onto $\Delta = \{\mathrm{x} = (x_1, x_2, \ldots, x_n) \mid x_1 \in \mathbb{Z}_m^*, x_i \in \mathbb{Z}_m, i = 2, \ldots, n\}$ is an injective map.

So the map of kind $G = T_1FT_2$, where nonlinear $F : \mathbb{Z}_m^n \longrightarrow \mathbb{Z}_m^n$ satisfies (i) and (ii) can be used as encryption tool for the plainspace Ω. The cryptosystem of this kind with F defined in terms of algebraic graphs is proposed in [14].

In this paper we used described above scheme with different core function F to define family of graph based nonbijective stream ciphers. Finally we propose the following generalization of nonbijective encryption bases on the map F with the properties (i) and (ii). Instead of (ii) we consider the following property:

(iii) The restriction of F onto $\Delta = \{\mathrm{x} = (x_1, x_2, \ldots, x_n) \mid x_1 \in \mathbb{Z}_m^*, x_i \in \mathbb{Z}_m, i = 2, \ldots, n\}$ is a bijective map.

In fact, the totality $S = S(\Delta, \mathbb{Z}_m)$ of all maps satisfying (i) and (iii) is a semigroup and the restrictions of elements from S onto Δ form a group.

For example, the above defined map $\widetilde{H}$ is an element of S in the case of $b = 0$.

It is easy to see that a map of kind $G = F_1F_2 \dots F_sT_2$, $s \geq 1$ with nonlinear $F_i : \mathbb{Z}_m^n \longrightarrow \mathbb{Z}_m^n$ where $F_i \in S(\mathbb{Z}_m)$, $i = 1, 2, \dots, s-1$ and F_s satisfies properties (i) and (ii). It means that T_1GT_2 can be used for nonbijective multivariate encryption.

Let us consider a toy example of generation of bijective maps with one-dimensional invariant subspace and transformations with properties (i) and (ii) and elements from S via symbolic walks on algebraic graphs.

Let us consider a commutative ring $\mathbb{K}$ and bipartite graph $D(3, \mathbb{K})$ with the partition sets $P = \{(x_1, x_2, x_3) \mid x_i \in \mathbb{K}, i = 1, 2, 3\}$ and $L = \{(y_1, y_2, y_3) \mid y_i \in \mathbb{K}, i = 1, 2, 3\}$ and incidence relation $(x_1, x_2, x_3)I[y_1, y_2, y_3]$. We define colour $\rho(\mathrm{v}) \in \mathbb{K}$ of the vertex v via rules $\rho((x_1, x_2, x_3)) = x_1$ and $\rho([y_1, y_2, y_3]) = y_1$. It is easy to see that for each vertex v there is exactly one neighbour $\mathrm{u} = {}^{K}N_\alpha(v) = N_\alpha(v)$ ($\mathrm{v}I\mathrm{u}$) of chosen colour α. So let us consider the walk in the graph with starting point $\mathrm{p} = (p_1, p_2, p_3)$ of length t (number of edges). The information on the walk can be given by sequence of elements $\alpha_0 = p_1, \alpha_1, \alpha_2, \dots, \alpha_t$ which are colours of vertices of the walk. The walk consists of vertices $\mathrm{v}_0 = \mathrm{p}$, $\mathrm{v}_1 = N_{\alpha_1}(\mathrm{v}_0), \mathrm{v}_2 = N_{\alpha_2}(\mathrm{v}_1), \dots, \mathrm{v}_t = N_{\alpha_t}(\mathrm{v}_{t-1})$. If $\alpha_i \neq \alpha_{i+2}$ for $i = 0, 1, \dots, t-2$, then the walk does not contain consecutive edges. The walks with starting line $[l_1, l_2, l_3]$ can be described in similar way. Notice that the description of walks is written uniformly for any commutative ring $\mathbb{K}$.

To introduce "symbolic walks" in the graph $D(3, \mathbb{K})$ we need the infinite graph $D(3, \mathbb{K}[x_1, x_2, x_3])$ where $\mathbb{K}' = \mathbb{K}[x_1, x_2, x_3]$ is the ring of polynomials in variables x_1, x_2, x_3 with coefficients from $\mathbb{K}$. Points and lines of new graph are triples (f^1, f^2, f^3), $f^i \in \mathbb{K}[x_1, x_2, x_3]$, $i = 1, 2, 3$ and $[g^1, g^2, g^3]$, $g^i \in \mathbb{K}[x_1, x_2, x_3]$, $i = 1, 2, 3$.

We consider walks started from special point (x_1, x_2, x_3), where x_i are generic elements of $\mathbb{K}[x_1, x_2, x_3]$. Let a^1, a^2, …, a^t be special colours of vertices from the walk, taken from $\mathbb{K}[x_1]$. The walk contains $v_0 = (x_1, x_2, x_3)$, $v_1 = {}^{\mathbb{K}'}N_{a_1(x_1)}(v_0)$, $v_2 = {}^{\mathbb{K}'}N_{a_2(x_1)}(v_1)$, …, $v_t = {}^{\mathbb{K}'}N_{a_t(x_1)}(v_{t-1})$. The final vertex of the walk is a triple of kind $< a_t(x_1), h_1(x_1, x_2, x_3), h_2(x_1, x_2, x_3) >$ where pair $<, >$ stands for brackets $(,)$ in the case of even t and parentheses $[,]$ in the case of odd t.

We join corresponding coordinates of initial and last vertex of the above symbolic walk by arrows $x_1 \longrightarrow a_t(x_1)$, $x_2 \longrightarrow h_1(x_1, x_2, x_3)$, and $x_3 \longrightarrow h_2(x_1, x_2, x_3)$ and obtain the standard form of the transformation of K^3 into itself. We use symbol $H = H_{a_1, a_2, \dots, a_t}$ for the map corresponding to the sequence of colours $a_1(x_1), a_2(x_1), \dots, a_t(x_1)$. Notice that for the computation of this map we use only operations $+, -$, and $\times$ of commutative ring $\mathbb{K}'$.

Obviously, affine subspace $U = \{(a, 0, 0) \mid a \in \mathbb{K}\}$ is an invariant subset for the action of H.

We can check the following:

(A) In the case of bijective map $x_1 \longrightarrow a_1(x_1)$ the transformation $H_{a_1, a_2, \dots, a_t}$ is also bijective map.

In fact the preimage of $\mathrm{c} = H(p_1, p_2, p_3)$ can be computed fast as the final vertex of the following walk in the graph $D(3, \mathbb{K})$. Let $d = p_1$ be a solution of $a_t(x_1) = c_1$. $\mathrm{u}_0 = (c_1, c_2, c_3)$, $\mathrm{u}_1 = N_{a_{t-1}(d)}((\mathrm{u}_0))$, $\mathrm{u}_2 = N_{a_{t-2}(d)}((\mathrm{u}_1))$, …, $\mathrm{u}_{t-1} = N_{a_t(d)}((\mathrm{u}_{t-1}))$, $(p_1, p_2, p_3) = N_d(\mathrm{u}_{t-1})$.

(B) If $\mathbb{K} = \mathbb{Z}_m$ and $a_t(x_1)$ is chosen as $a{x_1}^r + b$ where $a \in \mathbb{Z}_m^*$ and $gcd(r, \phi(m)) = 1$ then the map $H = H_{a_1, a_2, \dots, a_t}$ satisfies the property (ii).

Let $H(p_1, p_2, p_3) = (c_1, c_2, c_3)$ and $p_1 \in \mathbb{Z}_m^*$. Then $d = p_1$ can be computed as solution of $a{x_1}^r + b = c_1$, i. e. $d = (C_1 - b)^s$ where $rs = 1 \bmod \phi(m)$. The point p can be computed as final vertex of the walk described above.

(C) If $\mathbb{K} = \mathbb{Z}_m$ and $a_t(x_1)$ is chosen as $a{x_1}^r + b$ where $a \in \mathbb{Z}_m^*$ and $gcd(r, \phi(m)) = 1$ then the map $H = H_{a_1, a_2, \dots, a_t}$ satisfies the property (iii).

(D) If $a_i(x_1) = x_1 + b_i$ for $i = 1, 2, \dots, t$ then $H = H_{a_1, a_2, \dots, a_t}$ is a bijective cubical map.

You can play with "Sage" and check that property (D) holds for instance in the cases $k = 3, 4$ and 5.

(E) If $\mathbb{K} = \mathbb{Z}_p$ where p is prime integer then the map $E_{b_1, b_2, b_3} = H_{x_1 + b_1, x_1 + b_2, x_1 + b_3}$ does not have fixed points if $b_2 \neq 0$, $b_1 \neq b_3$. Let us treat E_{b_1, b_2, b_3} as "encryption map" on $\mathbb{K}^3$ with the password (b_1, b_2, b_3). Then different passwords produce distinct ciphertext. These properties follow from the fact that the girth (length of minimal cycle) of the graph $D(3, \mathbb{Z}_p)$ is eight.

(F) Let us consider the encryption map $T_1E_{b_1, b_2, b_3}$, where T_1 is linear map that sends x_1 to $x_1 + x_2 + x + 3$ without change of x_2 and x_3. Then, changing a single character in the plaintext causes changing of vast majority of ciphertext characters. Similarly, changing a single character of the password causes the change of most characters of ciphertext.

(G) Let us consider some nonbijective Eulerian deformations for encryption scheme (F):

(G.1) Let us take $\mathbb{K} = \mathbb{Z}_m$ with composite m instead of $\mathbb{Z}_p$ and $H = H_{a(x_1) + b_1, x_1 + b_2, a(x_1) + b_3}$ where $a(x_1) = a{x_1}^r$, $a \in \mathbb{Z}_m^*$, $gcd(r, \phi(m)) = 1$ instead of $E = E_{b_1, b_2, b_3}$ and denote this map as ${}^rE_{b_1, b_2, b_3}$. We consider encryption map $E^1 = E^1(b_1, b_2, b_3) = T_1{}^rE_{b_1, b_2, b_3}$ on the plainspace $\Delta = \{(a, 0, 0) \mid a \in \mathbb{K}^*\}$.

(G.2) Let us consider Eulerian map $d(s) : (x_1, x_2, x_3) \longrightarrow (b{x_1}^s, x_2, x_3)$, $b \in \mathbb{Z}_m^*$, $gcd(s, \phi(m)) = 1$ and change E_1 for $E_2 = T_1d(s)E_1$.

(G.3) Let us consider Eulerian map $D(l, b) : (x_1, x_2, x_3) \longrightarrow (c{x_1}^l + b, x_2, x_3)$, $c \in \mathbb{Z}_m^*$, $gcd(l, \phi(m)) = 1$, $b \in \mathbb{Z}_m$ and consider encryption $E_3 = T_1d(s)E_1(b_1, b_2, 0)D(l, b_3)$.

Assume that ring elements a, b, c and integers r, s, l are internal parameters of encryption algorithms E_1, E_2, E_3 working with the same plainspace and the same keyspace of

tuples (b_1, b_2, b_3). Symbolic computation with "Sage" allows to compare degrees of maps E_1, E_2, E_3. Computer simulations demonstrates the similarity of mixing properties of E_i, $i = 1, 2, 3$, with mixing properties of E.

4. On the Class of Bivariate Graphs

Let $\mathbb{K}$ be a commutative ring. We define $T(n, \mathbb{K})$ as a bipartite graph with the set of vertices $V(T) = P \cup L$, $P \cap L = \emptyset$. We call $P = \mathbb{K}^n$ a set of points and $L = \mathbb{K}^n$ a set of lines (two copies of a Cartesian power of $\mathbb{K}$ are used). We will use two types of brackets to distinguish points $(p) \in P$ and lines $[l] \in L$:

$$\begin{aligned} (p) &= (p_1, p_2, \ldots, p_n) \in P, \\ [l] &= [l_1, l_2, \ldots, l_n] \in L. \end{aligned} \tag{3}$$

p_i, l_i $(1 \le i \le n)$ are elements of $\mathbb{K}$. We say that vertex (p) (point (p)) is incident with the vertex $[l]$ (line $[l]$) and we write $(p)I_T[l]$, if the following relations between their coordinates hold:

$$\begin{aligned} p_2 - l_2 &= e_2^1 p_1 l_1 \\ p_3 - l_3 &= e_3^1 p_1 l_2 + e_3^2 l_1 p_2 \\ &\vdots \\ p_s - l_s &= e_s^1 p_1 l_{i_s} + e_s^2 l_1 p_{j_s} \\ &\vdots \\ p_n - l_n &= e_n^1 p_1 l_{i_n} + e_n^2 l_1 p_{j_n} \end{aligned} \tag{4}$$

where $e_2^1, e_s^1, e_s^2 \in \{0, 1, -1\}$, $1 \le i_s < s$, $1 \le j_s < s$. So the incidence relations for graph $T = T(n, \mathbb{K})$ are given by condition $(p)I_T[l]$. The set of edges consists of all pairs $\{(p), [l]\}$ for which $(p)I_T[l]$. Let us consider the case of finite commutative ring $\mathbb{K}$, with $|\mathbb{K}| = k$. As it instantly follows from the definition, the order of our bipartite graph is $|V(T)| = 2k^n$ and the number of edges is $|E(T)| = k^n \cdot k = k^{n+1}$. Graphs $T = T(n, \mathbb{K})$ are k-regular. In fact, the neighbour of a given point (p) is given by the above equations, where parameters $p_1, p_2, \ldots, p_n$ are fixed elements of the ring and symbols $l_1, l_2, \ldots, l_n$ are variables. It is easy to see that if we set l_1 then the choice uniformly establishes values $l_2, l_3 \ldots, l_n$. So each point has precisely k neighbours. In a similar way we observe that the neighbourhood of any line also contains k neighbours. Notice that the order and degree of our graph defined via strings i_s, j_s, e_2^1, e_s^1, e_s^2, where $s = 2, 3, \ldots, n$, does not depend on the strings.

Let us consider some examples.

4.1. Wenger Graphs $W(n, \mathbb{K})$. In 1991 Wenger defined the family of bipartite, p-regular graphs $H_n(p)$, where p is prime number [5]. In [4] Lazebnik and Ustimenko introduced straight forward generalization $W(n, q)$ of these graphs via change of $\mathbb{F}_p$ to $\mathbb{F}_q$, where q is a prime power. They used special Lie algebra and proved that the family of bipartite, q-regular graphs $W(n, q)$, $n \ge 2$. Graphs $W(n, q)$ are defined for all prime powers and $H_n(p) = W(n, p)$ are defined only for primes.

The set of vertices of infinite incidence structure (P, L, I) is $V = P \cup L$ and the set of edges E consists of all pairs $\{(p), [l]\}$ for which $(p)I[l]$. Bipartite graphs $W(n, q)$ have partition sets P_n (collection of points) and L_n (collection of lines) isomorphic to vector space $\mathbb{F}_q^n$, where $n \in \mathbb{N}_+$. Let us use the following notations for points and lines in graph $W(n, q)$:

$$\begin{aligned} (p) &= (p_1, p_2, p_3, \ldots, p_n) \in P, \\ [l] &= [l_1, l_2, l_3, \ldots, l_n] \in L. \end{aligned} \tag{5}$$

The point (p) is incident with the line $[l]$, and we write $(p)I_W[l]$, if the following relations between their coordinates hold:

$$l_i - p_i = p_1 l_{i-1}, \tag{6}$$

for $2 \le i \le n$. The graphs $W(n, \mathbb{F}_q)$ have cycles of length 8.

One can change finite field $\mathbb{K}$ for general commutative ring $\mathbb{K}$ and work with graph $W(n, \mathbb{K})$.

4.2. Graphs $A(n, \mathbb{K})$. Graphs $A(n, \mathbb{K})$ are formally appearing as tools for the study of $D(n, \mathbb{K})$ properties by V. Ustimenko. Later on the graphs $E(n, \mathbb{K})$ were presented with another name as an independent family $A(n, q)$ for the first time in [6] for cryptographic applications.

Let us use the following notations for points and lines in the graph $A(n, \mathbb{K})$:

$$\begin{aligned} (p) &= (p_1, p_2, p_3, \ldots, p_n) \in P, \\ [l] &= [l_1, l_2, l_3, \ldots, l_n] \in L. \end{aligned} \tag{7}$$

The point (p) is incident with the line $[l]$, and we write $(p)I_A[l]$, if the following relations between their coordinates hold:

$$\begin{aligned} l_2 - p_2 &= l_1 p_1 \\ l_3 - p_3 &= p_1 l_2 \\ l_4 - p_4 &= l_1 p_3 \\ l_i - p_i &= p_1 l_{i-1} \quad \text{for odd } i \\ l_i - p_i &= l_1 p_{i-1} \quad \text{for even } i \end{aligned} \tag{8}$$

for $3 \le i \le n$.

4.3. Graphs $D(n, \mathbb{K})$. The following interpretation of a family of graphs $D(n, \mathbb{K})$ in case $\mathbb{K} = \mathbb{F}_q$ can be found in [4]. By I_D we denote the incidence relation for this graph. Let us use the following notations for points and lines:

$$\begin{aligned} (p) &= (p_1, p_2, p_3, \ldots, p_n) \in P, \\ [l] &= [l_1, l_2, l_3, \ldots, l_n] \in L. \end{aligned} \tag{9}$$

Two types of brackets allow us to distinguish points from lines. Points and lines are elements of two copies of the vector space over $\mathbb{K}$. Point (p) is incident with the line $[l]$, and we write $(p)I_D[l]$, if the following relations between their coordinates hold:

$$\begin{aligned} l_2 - p_2 &= l_1 p_1 \\ l_3 - p_3 &= p_1 l_2 \\ l_4 - p_4 &= l_1 p_2 \\ l_i - p_i &= p_1 l_{i-2} \quad \text{for } i \bmod 4 \equiv 2 \text{ or } i \bmod 4 \equiv 3 \\ l_i - p_i &= l_1 p_{i-2} \quad \text{for } i \bmod 4 \equiv 0 \text{ or } i \bmod 4 \equiv 1 \end{aligned} \tag{10}$$

where $3 \le i \le n$.

The set of vertices is $V = P \cup L$ and the set of edges E consists of all pairs $\{(p), [l]\}$ for which $(p)I_D[l]$. Bipartite graphs $D(n, \mathbb{K})$ have partition sets P (collection of points) and L (collection of lines) isomorphic to vector space $\mathbb{K}^n$, where $n \in \mathbb{N}_+$.

4.4. Graphs $\widetilde{D(n, \mathbb{K})}$. Formal definitions for the family of graphs $\widetilde{D(n, \mathbb{K})}$ were presented in [7].

Construction of projective limits graphs of $\widetilde{D(n, \mathbb{K})}$ appears in papers motivated by results on embeddings of Chevalley group geometries in the corresponding Lie algebras and construction of blow-up for an incidence system of Weyl groups. Moreover, this structure is the base for construction of family of graphs $D(n, \mathbb{K})$ (see [7]).

Let us use the analogical notations for points and lines in graph $\widetilde{D(\mathbb{K})}$:

$$\begin{aligned} (p) &= (p_1, p_2, p_3, \ldots, p_n) \in P, \\ [l] &= [l_1, l_2, l_3, \ldots, l_n] \in L. \end{aligned} \tag{11}$$

In the incidence structure $\widetilde{(P, L, I)}$ the point (p) is incident with the line $[l]$, and we write $(p)I_{\widetilde{D}}[l]$, if the following relations between their coordinates hold:

$$\begin{aligned} l_2 - p_2 &= l_1 p_1 \\ l_3 - p_3 &= p_1 l_2 \\ l_4 - p_4 &= l_1 p_2 \\ l_5 - p_5 &= l_1 p_3 - p_1 l_4 \\ l_i - p_i &= p_1 l_{i-1} \quad \text{for } i \bmod 3 \equiv 0 \\ l_i - p_i &= l_1 p_{i-2} \quad \text{for } i \bmod 3 \equiv 1 \\ l_i - p_i &= l_1 p_{i-2} - p_1 l_{i-1} \quad \text{for } i \bmod 3 \equiv 2 \end{aligned} \tag{12}$$

for $3 \le i \le n$.

Graphs from families $D(n, \mathbb{K})$ and $\widetilde{D(n, \mathbb{K})}$ are bipartite, k-regular, where $|\mathbb{K}| = k$. The girth of graphs from the described families increases with the growth of n. In fact $D(n, q)$ is a family of graphs of large girth and there is a conjecture that $\widetilde{D(n, q)}$ is another family of graphs of a large girth.

All graphs from the considered families are k-regular and bipartite and the set of vertices is $V = P \cup L$, $P \cap L = \emptyset$. They are sparse graphs.

It is clear that there is a natural homomorphism of $T(n + 1, \mathbb{K})$ onto $T(n, \mathbb{K})$ of "deleting the last coordinate" that sends $(x_1, x_2, \ldots, x_n, x_{n+1})$ to $(x_1, x_2, \ldots, x_n)$ and $[y_1, y_2, \ldots, y_n, y_{n+1}]$ to $[y_1, y_2, \ldots, y_n]$. It means that there is a well-defined projective limit $T(\mathbb{K})$ of graphs $T(n, \mathbb{K})$, $n \longrightarrow \infty$. Bivariate graphs form a special subclass of so called *linguistic graphs* for which natural projective limits are defined in a similar way.

Recall that the girth $g = g(\Gamma)$ of the graph Γ is the length of its minimal cycle.

Let us assume that the girth $g(n)$ of graphs $T(n, \mathbb{K})$ is unbounded. The obvious inequality $g(n + 1) \ge g(n)$ holds. It means that projective limit $T(\mathbb{K})$ has to be a $|\mathbb{K}|$-regular forest. We have such situation in cases of graphs $A(n, \mathbb{F}_q)$ and $D(n, \mathbb{F}_q)$. If $q \ge 2$ then $A(\mathbb{F}_q)$ is a single tree presented by the above equations. Graph $D(\mathbb{F}_q)$ is an infinite forest containing infinitely many trees.

Projective limit $W(\mathbb{F}_q)$ of Wenger graphs is an infinite connected graph containing cycles of length 8.

5. General Encryption Algorithm No. 1

We can convert graph $T(n, \mathbb{K})$ to finite automaton in the following way. Let $\mathrm{v} = (v_1, v_2, v_3, v_4, \ldots, v_n) \in V(T(n, \mathbb{K}))$ (or $\mathrm{v} = [v_1, v_2, v_3, v_4, \ldots, v_n] \in V(T(n, \mathbb{K}))$) and $N_\alpha(\mathrm{v})$ be the operator of taking neighbour of vertex v where the first coordinate is α:

$$\begin{aligned} N_\alpha(v_1, v_2, v_3, v_4, \ldots, v_n) &\longrightarrow [\alpha, *, *, *, \ldots, *], \\ N_\alpha[v_1, v_2, v_3, v_4, \ldots, v_n] &\longrightarrow (\alpha, *, *, *, \ldots, *), \end{aligned} \tag{13}$$

where $\alpha \in \mathbb{K}$. The remaining coordinates can be determined uniquely using relations describing the chosen graph $T(n, \mathbb{K})$.

We convert $T(n, \mathbb{K})$ to finite automaton via joining v an $N_\alpha(\mathrm{v})$ by directed arrow with weight α. We assume that all vertices of the graph are accepting states.

A bit more interesting object is a symbolic bivariate automaton. Let $\mathrm{a}(x) = (\alpha_1(x), \alpha_2(x), \ldots, \alpha_s(x))$ be a string of elements from $\mathbb{K}[x]$ (totality of polynomials in variable x with coefficients from $\mathbb{K}$).

We introduce operator $N^s_{\mathrm{a}(\mathrm{x})}(\mathrm{v})$, where v is a point or a line with coordinates $v_1, v_2, \ldots, v_n$, of taking the last vertex u of the path v, $\mathrm{v}_1 = N_{\alpha_1(v_1)}(\mathrm{v})$, $\mathrm{v}_2 = N_{\alpha_2(v_1)}(\mathrm{v}_1)$, …, $\mathrm{v}_s = N_{\alpha_s(v_1)}(\mathrm{v}_{s-1}) = \mathrm{u}$.

We refer to $N^s_{\mathrm{a}(\mathrm{x})}$ as a computation of the symbolic automaton with the string

$$\mathrm{a}(\mathrm{x}) = (\alpha_1(\mathrm{x}), \alpha_2(\mathrm{x}), \ldots, \alpha_s(\mathrm{x})) \tag{14}$$

$\alpha_i \in \mathbb{K}[x]$, $i = 1, \ldots, s$, and initial state $\mathrm{v} = (v_1, v_2, v_3, v_4, \ldots, v_n) \in T(n, \mathbb{K})$ (or $\mathrm{v} = [v_1, v_2, v_3, v_4, \ldots, v_n] \in T(n, \mathbb{K})$). We can consider $F_s(\mathrm{v}) = N^s_{\mathrm{a}(v_1)}(\mathrm{v})$ as a map on $P \cup L$.

It is easy to see that the restriction of this map on P is a polynomial transformation of $P = \mathbb{K}^n$ into P (parameter s is even) or L (parameter s is odd) of kind

$$\begin{aligned} x_1 &\longrightarrow f_1(x_1, x_2, \ldots, x_n), \\ x_2 &\longrightarrow f_2(x_1, x_2, \ldots, x_n), \\ &\vdots \\ x_n &\longrightarrow f_n(x_1, x_2, \ldots, x_n). \end{aligned} \tag{15}$$

Notice that generally F_s is not a bijection. Let us consider an invertibility condition for F_s.

Proposition 1. *Let the equations of kind $\alpha_s(x) = b, b \in \mathbb{K}$ have exactly one solution. Then map F_s is invertible.*

Proof. It is easy to check that if $F_s(\overline{x}) = \overline{y}$ then $F_s^{-1}(\overline{y}) = \overline{x}$. It is easy to see that $f_1(x_1, x_2, \ldots, x_n) = \alpha_s(x_1)$. Let p be some point from P_n and $F_n(\mathrm{p}) = (c_1, c_2, \ldots, c_n)$ (point or line). Then the equation $\alpha_s(x_1) = c_1$ has a unique solution η. So we can compute $\eta_1 = \alpha_1(\eta), \eta_2 = \alpha_2(\eta), \ldots, \eta_{s-1} = \alpha_{s-1}(\eta)$.

We can compute the chain $\mathrm{c} = (c_1, c_2, \ldots, c_n)$, $N_{\eta_{s-1}}(\mathrm{c}) = \mathrm{c}_1$, $N_{\eta_{s-2}}(\mathrm{c}_1) = \mathrm{c}_2, \ldots, N_{\eta_1}(\mathrm{c}_{s-2}) = \mathrm{c}_{s-1}$, $N_{\eta}((\mathrm{c}_{s-1})) = \mathrm{c}_s = (p_1, p_2, \ldots, p_n)$ with $\eta = p_1$. So F_n is a bijection. □

Notice that $N^s_{\mathrm{a}(x)}$ for $\mathrm{a}(x)$ of kind $\alpha_1(x) = \beta_1(x), \alpha_2(x) = \beta_2(\alpha_1(x)), \alpha_3 = \beta_3(\alpha_2(x)), \ldots, \alpha_s(x) = \beta_s(\alpha_{s-1}(x))$ is a composition of $N^1{}_{\beta_1(x)}, N^1{}_{\beta_2(x)}, \ldots, N^1{}_{\beta_s(x)}$. In this case invertibility of each $\beta_i(x)$, $i = 1, 2, \ldots, s$, guarantees the bijectivity of $N^s_{\mathrm{a}(x)}$. We refer to such case as *recurrently defined string*.

Let L_1 and L_2 be sparse affine bijective transformation of the affine space (free module in other terminology) $\mathbb{K}^n$

$$\begin{aligned} L_1 &= T_{A,b} : \overline{x} \longrightarrow \overline{x}A + b, \\ L_2 &= T_{C,d} : \overline{x} \longrightarrow \overline{x}C + d, \end{aligned} \tag{16}$$

where $A = [a_{i,j}]$ and $C = [c_{i,j}]$ are $n \times n$ matrices with $a_{i,j}, c_{i,j} \in \mathbb{K}$. It is clear that

$$\begin{aligned} L_1^{-1} &= T_{A,b}^{-1} = T_{A^{-1}, -bA^{-1}}, \\ L_2^{-1} &= T_{C,d}^{-1} = T_{C^{-1}, -dC^{-1}}. \end{aligned} \tag{17}$$

Let F_n be a polynomial map of $\mathbb{K}^n$ to itself. We refer to $G_n = L_1 F_n L_2$ as *affine deformation* of F_n.

5.1. Symmetric Cipher No. 1. We can use the data on the graph $T(n, \mathbb{K})$, the symbolic computation given by the string $\mathrm{a} = \mathrm{a}(x)$ of polynomials $\alpha_1(x), \alpha_2(x), \ldots, \alpha_s(x)$, where $\alpha_s(x)$ is a bijective map of $\mathbb{K}$ to itself and affine transformations L_1 and L_2 in the following encryption scheme.

Correspondents Alice and Bob agree on a private encryption key

$$K_p = (L_1, L_2, \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_s)), \tag{18}$$

and keep the key in secret. Messages are written using characters from the alphabet $\mathbb{K}$. So the plainspace is $\mathbb{K}^n$ and its elements must be treated as points (or lines) of the graph. To encrypt they use the composition

$$L_1 \circ N^s_{\mathrm{a}} \circ L_2. \tag{19}$$

Notice that the computation has to be executed in numerical level:

(1) Correspondent Alice writes plaintext $\mathrm{p} = (p_1, p_2, \ldots, p_n)$ and computes

$$L_1(\mathrm{p}) = (\nu_1, \nu_2, \ldots, \nu_n) = \nu \tag{20}$$

and treats ν as point of the bivariate graph $T(n, \mathbb{K})$.

(2) She computes parameters $\mu_i = \alpha_i(\nu_1)$ for $i = 1, 2, \ldots, s$.

(3) She computes p_0 as $L_1(\mathrm{p})$, p_1 as $N_{\mu_1}(\mathrm{p}_0)$, p_2 as $N_{\mu_2}(\mathrm{p}_1)$, $\ldots$, p_s as $N_{\mu_s}(\mathrm{p}_{s-1})$.

(4) She computes the ciphertext c as $L_2(\mathrm{p}_s)$.

Alice and Bob can use their knowledge about triple (L_1, L_2, a) for the decryption. Let us assume that Bob receives the ciphertext c from Alice. To decrypt the ciphertext Bob proceeds as follows:

(1) He has to compute c_0 as ${L_2}^{-1}(\mathrm{c})$.

(2) He treats the string of coordinates of this tuple as a vertex of the graph, which is a point in case of even s or the line in case of odd s with coordinates $c_1^0, c_2^0, \ldots, c_n^0$.

(3) Bob must find a solution η of $\alpha_s(x) = c_1^0$ and form a string $\eta_0 = \eta, \eta_1 = \alpha_1(\eta), \eta_2 = \alpha_2(\eta), \ldots, \eta_{s-1} = \alpha_{s-1}(\eta)$.

(4) He computes c_1 as $N_{\eta_{s-1}}(\mathrm{c}_0)$, c_2 as $N_{\eta_{s-2}}(\mathrm{c}_1), \ldots, \mathrm{c}_s$ as $N_{\eta_0}(\mathrm{c}_{s-1})$.

(5) He computes the plaintext p as ${L_1}^{-1}(\mathrm{c}_s)$.

Remark 2. In the case of identity maps L_1 and L_2 one can try Dijkstra's algorithm for finding the shortest path between plaintext and ciphertext. Notice that its complexity is $O(\nu \log \nu)$, but here ν is exponential q^n. Therefore we get worse complexity even than brute force search via the key space.

In the case of recurrently defined symbolic computation as above the encryption bijective map is $F_s = L_1 N^1{}_{\beta_1(x)} N^1{}_{\beta_1(x)} \ldots N^1{}_{\beta_s(x)} L_2$. As we already see, this encryption transformation is equivalent to $L_1 N^s{}_{\mathrm{a}(x)} L_2$, where $\mathrm{a}(x) = (\beta_1(x), \beta_2(\beta_1(x)), \ldots, \beta_s(\beta_{s-1}(\ldots(\beta_1(x)))))$. Recurrently defined symbolic computation is an example of the polynomial map with an invertible decomposition. It has various applications in the development of multivariate key exchange protocols and asymmetric multivariate algorithm. The most popular case of implementation is related to graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$ (see [15, 17, 18]), where $\mathbb{K}$ is a finite field of arithmetical rings $\mathbb{Z}_m$ and strings of kind $\beta_1 = x + d_1$, $\beta_2 = x + d_2, \ldots, \beta_s = x + d_s$, where $d_i + d_{i+1}, i = 1, 2, \ldots, s-2$ are regular elements of the ring $\mathbb{K}$. We refer to such case as shifting encryption.

Let us consider the case of strong symmetric encryption, when the function is $\alpha_s(x) = ax+b$, with a regular (invertible) element of $\mathbb{K}$. In this case it is easy to show that degrees of encryption map F_n and decryption map F_n^{-1} are the same. The advantage of this case is its universality. One can implement it in case of arbitrary chosen finite ring $\mathbb{K}$.

6. On Properties of Bivariate Graph Based Bijective Encryption Maps

The girth G of simple graph G is the length of its shortest cycle. It is a known fact that the girth of the graph $D(n, \mathbb{F}_q)$ is $\geq n+5$. So in the case of shifting encryption the map with the password $x+d_1, x+d_2, \ldots, x+d_s, s < n+5$ the encryption map F_n has no fixed points. So ciphertext is always different from the plaintext. Let us consider deformed shifting encryption of kind $\tau_L F_n \tau_R$. We assume that affine maps τ_L and τ_R are fixed. Correspondents are able to change string $d_1, d_2, \ldots, d_s$ for another one.

We assume that $d_i + d_{i+1} \neq 0$ for $i = 1, 2, \ldots, s-2$. Such choice means that encryption map corresponds to the path of length s. The inequality $g(D(n,q)) \geq n+5$ implies that different strings of length $s < (n+5)/2$ produce different ciphertexts. So even in the case when τ_L and τ_R are known to adversary the complexity of attacks without an access to unencrypted information is bounded from below by $q^{(n+5)/2}$.

Let $\mathbb{M}$ be a multiplicative subset of general commutative ring $\mathbb{K}$, i.e., $\mathbb{M}$ is closed under the ring multiplication and it does not contain 0. We say that a string $d_1, d_2, \ldots, d_s$ is $|\mathbb{M}|$-regular if $d_i + d_{i+1} \in \mathbb{M}$ for $i = 1, 2, \ldots, s-2$. It was proven that different M-regular strings of length $s < (n+5)/2$ produce distinct ciphertexts from the same plaintext. So in the case of $|\mathbb{K}| = k$, $|\mathbb{M}| = m$ the resistance to attacks without access to unencrypted data is bounded from below by $mk^{(n+5)/2-1}$.

It was proven that graphs $A(n, \mathbb{F}_q)$ form a family of graphs of increasing girth $h(n)$ that tends to infinity as n grows. The speed of growth of $h(n)$ needs further evaluation. It was proven also that different $|\mathbb{M}|$-regular strings of length $s < n$ produce different encryption maps.

Results on $|\mathbb{M}|$-regular strings of length restricted maps are obtained in terms of dynamical systems corresponding to graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$.

Let us assume that maps τ_L and τ_R are identities and consider the groups of transformations $GD(n, \mathbb{K})$ and $GA(n, \mathbb{K})$ generated by shifting encryption maps corresponding to strings of even length. In [16] it was proven that all elements of $GD(n, \mathbb{K})$ are cubical transformations of affine spaces P_n and L_n. Similar result for $GA(n, \mathbb{K})$ is stated in [6]. As it follows instantly from this result transformation $F_n' = \tau_L F_n \tau_R$ and its inverse are cubical transformations.

The cryptanalytic corollary of this statement is justification of linearization attacks on stream ciphers corresponding to stream ciphers based on graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$.

Let correspondents use the transformation F_n'. The adversary has knowledge on the general scheme of open algorithm but not on the data for τ_L and τ_R or on the shifting string. So he knows about cubic nature of encryption. We assume that he has access to the unencrypted information and is able to intercept quite many pairs of kind (p, c), where p is plaintext and c corresponding ciphertext.

Then adversary writes G_n, which is a formal cubical map in standard form with the unknown coefficients in front of monomial terms. He or she is able to solve system of $O(n^3)$ equations of kind $G_n(\mathrm{c}) = \mathrm{p}$ and restore the map G_n. So adversary could control the communication channel. The complexity of such direct linearization attack is $O(n^{10})$.

7. On the Implementation of Graph Based Stream Cipher Based on Nonbijective Maps

Let us describe an implemented algorithm, which can run in the case of arbitrary commutative ring $\mathbb{K}$ and arbitrary bivariate graph $T(n, \mathbb{K})$. We slightly modify the above described symmetric algorithm based on bivariate graphs $T(n, \mathbb{K})$ which is not a case of shifting encryption. Firstly, we take a symbolic computation for string $\mathrm{a} = \mathrm{a}(x) = (\alpha_1(x), \alpha_2(x), \ldots, \alpha_s(x))$, with $\alpha_i(x) = x^d + d_s$, $i = 1, 2, \ldots, s$ where d is mutually prime with the order of $\mathbb{K}^*$. So equation $x^d + d_s = c$, $x \in \mathbb{K}^*$ has at most one solution. We take L_1 as an affine bijective transformation of kind $x_1 \longrightarrow x_1 + x_2 + \cdots + x_n, x_2 \longrightarrow l_2(x_1, x_2, \ldots, x_n), x_3 \longrightarrow l_3(x_1, x_2, \ldots, x_n), \ldots, x_n \longrightarrow l_n(x_1, x_2, \ldots, x_n)$, where l_i are linear functions from $K[x_1, x_2, \ldots, x_n]$. Correspondents will use the plainspace

$$\Omega = \{(x_1, x_2, \ldots, x_n) \mid x_1 + x_2 + \cdots + x_n \in \mathbb{K}^*,\ x_i \in \mathbb{K},\ i = 1, 2, \ldots, n\}. \tag{21}$$

They will use $L_1 N_{\mathrm{a}(x)}^s L_2$ as encryption map. To execute computation in time $O(n)$ they take finite parameter s and use loaded tables for $\alpha_i(x)$, $i = 1, 2, \ldots, s$ (one-dimensional arrays $a_i(x)$, $x \in \mathbb{K}^*$). So they will compute $L_1(\mathrm{p}) = \mathrm{v} = (v_1, v_2, \ldots, v_n)$, form sequence $\mu_i = \alpha_i(v_1)$, $i = 1, 2, \ldots, s$ and compute recurrently $\mathrm{v}_i = N_{\mu_i}(\mathrm{v}_{i-1})$, $i = 1, 2, \ldots, s$. They form the ciphertext c as $L_2(\mathrm{v}_s)$.

To decrypt they will take $\mathrm{c}_0 = (c_1^0, c_2^0, \ldots, c_n^0)$ as ${L_2}^{-1}(\mathrm{c})$ and find a solution η for the equation $x^d + d_s = c_1^0$. Loaded table of values for ${\alpha_s}^{-1}$ will allow to find η fast. Next they form a string $\eta_0 = \eta, \eta_1 = \alpha_1(\eta), \eta_2 = \alpha_2(\eta), \ldots, \eta_{s-1} = \alpha_{s-1}(\eta)$. So users take string $\mathrm{c}_1 = N_{\eta_{s-1}}(\mathrm{c}_0)$, $\mathrm{c}_2 = N_{\eta_{s-2}}(\mathrm{c}_1), \ldots, \mathrm{c}_s = N_{\eta_0}(\mathrm{c}_{s-1})$. Finally they get plaintext as $L^{-1}(\mathrm{c}_s)$.

The case of this symmetric algorithm appears as a private key for a cryptosystem introduced in [14] with the plainspace $\mathbb{Z}_m^n$.

We selected string of polynomials as $\alpha_i = x^d + d_i$, $d_i \in \mathbb{K}$, $i = 1, 2, \ldots, s$ and special linear transformations L_1 and L_2, given by the lists of linear forms.

We can theoretically evaluate degrees of encryption d_{enc} and decryption d_{dec}. In cases of graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$, these parameters are bounded below by some constants depending from parameters α_i, $i = 1, 2, \ldots, s$. We can select string of parameters and get d_{dec} large enough to make cryptanalysis a difficult task. In case $\widetilde{D(n, \mathbb{K})}$ the degrees are even larger; they have size $O(n)$. Notice that direct linearization attacks are formally impossible because the encryption map is not a bijective one.

The implementation of the algorithms in the present work was done using the Python programming language, in particular version 2.7. The code does not use any out-of-the-box libraries for facilitating operations with matrices. The tests for measuring the processing time have been executed on a machine with Intel Core2 Duo CPU 9600 1.60GHz x 2, RAM memory 4.8 GB, operating with Ubuntu 16.04 LTS. The complexity of the algorithms is of order $O(sn)$, where s is the length of the password. In particular, we implemented this stream cipher for the case $\mathbb{K} = \mathbb{Z}_{256}$ and $\alpha_i(x) = x^3 + d_i$ ($d = 3$ and $d_{\text{dec}} = 43$), $i = 1, 2, \ldots, s$ without using loaded tables for functions. Table 1 represents encoding and decoding time for three types of graphs and for different files size and length of passwords. A description of the implementation of "nonlinear part" of encryption process; i.e., computation of N_{a}^{s} is given in [1]. We recommend a password for which d_2 and $d_i - d_{i+2}$, $i = 1, 2, \ldots, s-2$ are regular elements of the ring.

8. General Encryption Algorithm No. 2 with the Usage of Nonlinear Colour Jump Function

We generalize Encryption Algorithm No. 1 for creation of multivariate transformation of $\mathbb{K}^n$ based on the bivariate graph $T(n, \mathbb{K})$ and its extension over $\mathbb{K}[x_1, x_2, \ldots, x_n]$.

The extension of graph $T(n, \mathbb{K})$ is $T(n, \mathbb{K}[x_1, x_2, \ldots, x_n])$, which has two partition sets P (points) and L (lines) isomorphic to $(\mathbb{K}[x_1, x_2, \ldots, x_n])^n$. In this infinite graph the incidence relation I are given by the same equation as in the graph $T(n, \mathbb{K})$ but over the ring $\mathbb{K}[x_1, x_2, \ldots, x_n]$.

Let us consider a special vertex of the graph $T(n, \mathbb{K}[x_1, x_2, \ldots, x_n])$ kind $(x_1, x_2, \ldots, x_n)$, where x_i, $i = 1, \ldots, n$, are generations of $\mathbb{K}[x_1, x_2, \ldots, x_n]$ over $\mathbb{K}$. We use *the colour jump operator*, which transforms point $(x_1, x_2, \ldots, x_n)$ to point $(g(x_1), x_2, \ldots, x_n)$ from $T(n, \mathbb{K}[x_1, x_2, \ldots, x_n])$.

We create *the jump fusion* $F_s^{\Xi} : \mathbb{K}^n \longrightarrow \mathbb{K}^n$ of nonlinear colour jump operator Ξ_g with modification previously defined F_s in the following way:

(1) First step, we use Ξ_g to point x:

$$\mathrm{x} = (x_1, x_2, x_3 \ldots, x_n) \longrightarrow$$
$$\Xi_g(x_1, x_2 \ldots, x_n) = (g(x_1), x_2, x_3, \ldots, x_n) = \mathrm{x}' \quad (22)$$
$$(\text{we make a jump})$$

and we treat x' as the point of the graph $T(n, \mathbb{K}[x_1, x_2, \ldots, x_n])$.

(2) In the next steps of jump fusion, we modified F_s differently in the cases of even and odd parameter s

(2.1) If $s = 2r + 1$, we use symbolic key defined by polynomials $\alpha_1(x_1)$, $\alpha_2(x_1) = \beta_1(g(x_1))$, $\alpha_3(x_1)$, $\alpha_4(x_1) = \beta_2(g(x_1))$, …, $\alpha_{2r}(x_1) = \beta_r(g(x_1))$, $\alpha_{2r+1}(x_1)$, where $\alpha_i, \beta_i \in \mathbb{K}[x]$ and α_{2r+1} is a bijection.

(2.2) If $s = 2r$, we use symbolic key defined by polynomials $\alpha_1(x_1)$, $\alpha_2(x_1) = \beta_1(g(x_1))$, $\alpha_3(x_1)$, $\alpha_4(x_1) = \beta_2(g(x_1))$, …, $\alpha_{2r}(x_1) = \beta_r(g(x_1))$, where $\alpha_i, \beta_i \in \mathbb{K}[x]$ and α_{2r} is a bijection.

(3) We form a path in the graph $T(n, \mathbb{K}[x_1, x_2, \ldots, x_n])$ of

$$\mathrm{x}' I \mathrm{v}_1 I \mathrm{v}_2 I \ldots I \mathrm{v}_s, \quad (23)$$

where $\rho(\mathrm{v}_i) = \alpha_i(x_1)$, i=1,2,…,s.

(4) Final vertex of the path is

$$\mathrm{v}_s = \alpha_s(\mathrm{x}_1), \mathrm{h}_2, \mathrm{h}_2, \ldots, \mathrm{h}_n, \quad (24)$$

where $h_i = \mathbb{K}[x_1, x_2, \ldots, x_n]$.

The jump fusion F_s^{Ξ} is the map

$$\begin{aligned} x_1 &\longrightarrow \alpha_s(x_1), \\ x_2 &\longrightarrow h_2(x_1, x_2, \ldots, x_n), \\ &\vdots \\ x_n &\longrightarrow h_n(x_1, x_2, \ldots, x_n). \end{aligned} \quad (25)$$

If the jump is trivial, i.e., $\Xi_g(x_1, x_2 \ldots, x_n) = (x_1, x_2, x_3, \ldots, x_n) = \mathrm{x}$ then $F_s^{\Xi} = F_s$.

In the case of affine deformation of F_n^{Ξ} by affine bijective transformation L_1 and L_2 of $\mathbb{K}^n$ our map

$$\begin{aligned} x_1 &\longrightarrow w_1(x_1, x_2, \ldots, x_n), \\ x_2 &\longrightarrow w_2(x_1, x_2, \ldots, x_n), \\ &\vdots \\ x_n &\longrightarrow w_n(x_1, x_2, \ldots, x_n), \end{aligned} \quad (26)$$

where $w_i \in \mathbb{K}[x_1, x_2, x_3, \ldots, x_n]$.

8.1. Symmetric Cipher No. 2 with the Jump Operator. Let $T(n, \mathbb{K})$ be bivariate graph with partition sets P (points) and L (lines) isomorphic to $\mathbb{K}^n$. We consider *the colour jump operator* Ξ_g, $g \in \mathbb{K}[x]$ in the set if points P, which transforms $(x_1, x_2, \ldots, x_n)$ to point $(g(x_1), x_2, \ldots, x_n)$.

Let us modify Symmetric Cipher No. 1 in the following way. The correspondents Alice and Bob use the data on the graph $T(n, \mathbb{K})$, with the colour jump operator Ξ_g, the symbolic computation given by the string $\mathrm{a} = \mathrm{a}(\mathrm{x})$ of polynomials $\alpha_1(x), \alpha_2(x), \ldots, \alpha_s(x)$, where $\alpha_s(x)$ is a bijective map of $\mathbb{K}$ to itself and affine transformations L_1 and L_2 in the following encryption scheme.

Thus, correspondents Alice and Bob agree on the private encryption key

$$K_p = \left(L_1, L_2, \Xi_g, a_g = (\alpha_1, \alpha_2, \ldots, \alpha_s)\right), \quad (27)$$

where polynomials $\alpha_1(x), \alpha_2(x), \ldots, \alpha_s(x)$ are defined differently depending on the length of the password s, i.e.,

(i) if $s = 2r + 1$ then $\mu_1 = \alpha_1(x)$, $\mu_2 = \alpha_2(x) = \beta_1(g(x))$, $\mu_3 = \alpha_3(x)$, $\mu_4 = \alpha_4(x) = \beta_2(g(x))$, …, $\mu_{2r} = \alpha_{2r}(x) = \beta_r(g(x))$, $\mu_{2r+1} = \alpha_{2r+1}(x)$ and $\alpha_i, \beta_i \in \mathbb{K}[x]$,

(ii) if $s = 2r$ then $\mu_1 = \alpha_1(x)$, $\mu_2 = \alpha_2(x) = \beta_1(g(x))$, $\mu_3 = \alpha_3(x)$, $\mu_4 = \alpha_4(x) = \beta_2(g(x))$, …, $\mu_{2r-1} = \alpha_{2r-1}(x)\mu_{2r} = \alpha_{2r}(x) = \beta_r(g(x))$, and $\alpha_i, \beta_i \in \mathbb{K}[x]$

Notice that the computation has to be executed in numerical level also in this scheme:

(1) Correspondent Alice writes plaintext $\mathrm{p} = (p_1, p_2, \ldots, p_n) \in \Omega$ and computes

$$L_1(p) = (v_1, v_2, \ldots, v_n) = v. \tag{28}$$

(2) In the next step she uses the colour jump operator and computes

$$\Xi_g(v) = (g(v_1), v_2, \ldots, v_n) = v' \tag{29}$$

and treats v' as point of the bivariate graph $T(n, \mathbb{K})$.

(3) She computes parameters $\mu_i = \alpha_i(v_1)$ for $i = 1, 2, \ldots, s$ with properly defined α_i.

(4) She computes p_1 as $N_{\mu_1}(v')$, p_2 as $N_{\mu_2}(\mathrm{p}_1)$, …, p_s as $N_{\mu_s}(\mathrm{p}_{s-1})$.

(5) Finally, she computes the ciphertext c as $L_2(\mathrm{p}_s)$.

Alice and Bob can use knowledge about their private key $(L_1, L_2, \Xi_g, \mathrm{a_g})$ for the decryption.

Let us assume that Bob receives the ciphertext c from Alice. To decrypt the ciphertext Bob proceeds as follows:

(1) He has to compute c_0 as ${L_2}^{-1}(\mathrm{c})$.

(2) He treats the string of coordinates of this tuple as a vertex of the graph, which is a point in case of even s or the line in case of odd s with coordinates $c_1^0, c_2^0, \ldots, c_n^0$.

(3) Bob must find a solution η of $\alpha_s(x) = c_1^0$ and form a string $\eta_0 = g(\eta)$, $\eta_1 = \alpha_1(\eta)$, $\eta_2 = \alpha_2(\eta)$, …, $\eta_{s-1} = \alpha_{s-1}(\eta)$.

(4) He computes c_1 as $N_{\eta_{s-1}}(\mathrm{c}_0)$, c_2 as $N_{\eta_{s-2}}(\mathrm{c}_1)$, …, c_s as $N_{\eta_0}(\mathrm{c}_{s-1}) = (c_0^s, c_1^s, c_2^s, \ldots, c_n^s)$.

(5) Next, he computes

$$\Xi^{-1}(\mathrm{c}_s) = (\eta, c_1^s, c_2^s, \ldots, c_n^s) = \mathrm{c}_s'. \tag{30}$$

(6) Finally, he computes the plaintext p as ${L_1}^{-1}(\mathrm{c}_s')$.

9. On the Implementation of Graph Based Stream Cipher Based on Nonbijective Maps with the Use of Nonlinear Colour Jump Operator

In this section we describe an implementation of general Encryption Algorithm No. 2 with the use of nonlinear colour jump function Ξ_g in the case of rings $\mathbb{K} = \mathbb{Z}_m$ and $\mathbb{K} = \mathbb{F}_q$ and graph $T(n, \mathbb{K})$.

We take a bijective transformation $x_1 \longrightarrow x_1 + x_2 + x_3 + \cdots + x_s$, $x_i \longrightarrow l_i(x_1, x_2, \ldots, x_n)$ $(i = 1, 2, \ldots, n)$, which is computable in time $O(n)$. For example linear forms $l_2 = x_2 + x_3 + \cdots + x_n$, $l_3 = x_3 + x_4 + \cdots + x_n$, …, $l_n = x_n$ can be chosen. We select L_2 as bijective affine transformation computable in time $O(n)$.

9.1. Case I: Let s Be an Odd Number $s = 2r + 1$. To define jump of the colour correspondents can take an arbitrary $g \in \mathbb{K}[x]$. The value of g on given element of commutative ring is computable for $O(n)$ elementary steps. According to the point (2.1) in Section 8, they select $\alpha_1(x)$ as $ax^d + d_1$, $\alpha_3(x)$ as $ax^d + d_3$, …, $\alpha_s(x) = ax^d + d_s$, $\beta_1(x) = x + b_1$, $\beta_2(x) = x + b_2$, …, $\beta_r(x) = x + b_r$, where $gcd(d, \phi(m)) = 1$ for $\mathbb{K} = \mathbb{Z}_m$ and $gcd(d, q - 1) = 1$ for $\mathbb{K} = \mathbb{F}_q$. Correspondents are working with the plainspace $\Omega = \{\mathrm{x} \mid x_1 + x_2 + \cdots + x_n \in \mathbb{K}^*\}$ in the case $\mathbb{K} = \mathbb{Z}_m$ and ${\mathbb{F}_q}^n$ in which the encryption map is a bijection. We assume that parameters a, $d_i - d_{i-2}$, $b_i - b_{i-1}$ are regular ring elements.

Decryption. Correspondent Bob takes obtained ciphertext c and computes ${L_2}^{-1}(\mathrm{c}) = [c_1^0, c_2^0, \ldots, c_n] = \mathrm{c}_0$. He solves the equation $ay^d + d_{2r+1} = c_1^0$ in unknown y. Let η be the solution. Bob computes $\eta_1 = \alpha_1(\eta), \eta_2 = \beta_1(g(\eta)), \ldots, \eta_{2r-1} = \alpha_{2r-1}(\eta)$, $\eta_{s-1} = \eta_{2r} = \beta_r(g(\eta))$.

He forms the path c_0, $N_{\eta_{2r}}(\mathrm{c}_0) = \mathrm{c}_1$, $N_{\eta_{2r-1}(\mathrm{c}_1)} = \mathrm{c}_2$, …, $N_{\eta_1}(\mathrm{c}_{2r-1}) = \mathrm{c}_{2r}$, $N_{g(\eta)}(\mathrm{c}_{2r}) = \mathrm{c}_{2r+1} = (c_1^s, c_2^s, \ldots, c_n^s)$ with $c_1^s = g(\eta)$ and next he uses jump operator $\Xi_g(\mathrm{c}_s) = (\eta, c_2^s, \ldots, c_n^s) = \mathrm{c}_s'$. Finally Bob computes the plaintext as ${L_1}^{-1}(\mathrm{c}_s')$.

9.2. Case II: Let s Be an Even Number $s = 2r$. Correspondents take $g(x) = tx^e + o$ to conduct jump operator Ξ_g, where $gcd(e, \phi(m)) = 1$ for $\mathbb{K} = \mathbb{Z}_m$ and $gcd(e, q-1) = 1$ for $\mathbb{K} = \mathbb{F}_q$ and t is regular element of $\mathbb{K}$. They select $\alpha_1(x) = f(x)$, where $f(x)$ is an arbitrary element of $\mathbb{K}[x]$ of degree d, e.g., $f(x) = ax^d$. They work also with $\alpha_{2i-1}(x) = f(x) + d_i$, $i = 1, 2, \ldots, r$ and $\alpha_{2i}(x) = x + b_i$, $i = 1, 2, \ldots, r$ (see the point (2.2) in Section 8). Similarly to previous case, parameters t, $d_i - d_{i-1}$, $b_i - b_{i-1}$ are regular ring elements.

Decryption. Correspondent Bob takes obtained ciphertext c and computes ${L_2}^{-1} = [c_1^0, c_2^0, \ldots, c_n^0] = \mathrm{c}_0$. He solves the equation $ty^e + o + d_r = c_1^0$. Let η be the solution. He computes $\eta_1 = f(\eta)$, $\eta_2 = \alpha_2(g(\eta))$, $a_3 = f(w) + b_1$, $a_4 = \alpha_4(g(\eta))$, …, $\eta_{2r-1} = f(\eta) + b_r$.

Bob computes the path c_0, $N_{\eta_{2r-1}}(\mathrm{c}_0) = \mathrm{c}_1$, $N_{\eta_{2r-2}}(\mathrm{c}_1) = \mathrm{c}_2$, …, $N_{\eta_1}(\mathrm{c}_{2r-1}) = \mathrm{c}_{2r-2}$, $N_{g(\eta)}(\mathrm{c}_{2r-2}) = \mathrm{c}_{2r-1}$, and next he uses the jump operator $\Xi_g(\mathrm{c}_s) = (\eta, c_2^s, \ldots, c_n^s) = \mathrm{c}_s'$. Finally Bob computes the plaintext as ${L_1}^{-1}(\mathrm{c}_s')$.

Example 3. Let s=3 (case I). Correspondents will use the plainspace

$$\Omega = \{(x_1, x_2, x_3, x_4) \mid x_1 + x_2 + x_3 + x_4 \in \mathbb{F}_{11}^*, x_i \in \mathbb{F}_{11},\ i = 1, 2, 3, 4\} \tag{31}$$

and private encryption key

$$K_p = \left(L_1, L_2, \Xi_g(x) = (g(x_1), x_2, x_3, x_4), a_g = (\alpha_1, \alpha_2, \alpha_3)\right), \tag{32}$$

where $g(x) = x^3 + x + 1$, $\alpha_1(x) = 3x^3 + 2$, $\alpha_2 = \beta_1(g(x))$ with $\beta_1(x) = x + 1$ and $\alpha_3(x) = 3x^3 + 5$. Notice that $\phi(11) = 10$, $gcd(3, 10) = 1$, and $d_3 - d_1 = 5 - 2 = 3$ is regular element. Let

$$L_1(x_1, x_2, x_3, x_4) = (x_1 + x_2 + x_3 + x_4, x_2 + x_3 + x_4, x_3 + x_4, x_4) \quad (33)$$

and L_2 be an identity map. Let us use graph $D(4, \mathbb{F}_{11})$ and users should do all calculations in modular arithmetic defined for $\mathbb{F}_{11}$.

(1) Alice writes plaintext $p = (1, 2, 5, 0)$ ($p \in \Omega$ since $gcd(1 + 2 + 5 + 0, 11) = 1$) and computes $L_1(p) = L_1(1, 2, 5, 0) = (8, 7, 5, 0) = (v_1, v_2, v_3, v_4) = v$.

(2) She computes $g(v_1) = g(8) = 8^3 + 8 + 1 = 4$ and

$$\Xi_g(v) = (g(v_1), v_2, v_3, v_4) = (4, 7, 5, 0) = v'. \quad (34)$$

(3) $\mu_1 = \alpha_1(v_1) = \alpha_1(8) = 3 \cdot 8^3 + 2 = 9$, $\mu_2 = \alpha_2(v_1) = \beta_1(g(v_1)) = \beta_1(g(8)) = \beta_1(4) = 4 + 1 = 5$ and $\mu_3 = \alpha_3(8) = 3 \cdot 8^3 + 5 = 1$.

(4) She gets $v' = (4, 7, 5, 0) \longrightarrow N_{\mu_1}(v') = [9, 10, 1, 8] = \mathrm{p}_1 \longrightarrow N_{\mu_2}(\mathrm{p}_1) = (5, 9, 6, 4) = \mathrm{p}_2 \longrightarrow N_{\mu_3}(\mathrm{p}_2) = [1, 3, 10, 2] = \mathrm{p}_3$.

(5) Finally, $L_2(\mathrm{p}_3) = L_2([1, 3, 10, 2]) = [1, 3, 10, 2] = \mathrm{c}$ is the ciphertext.

Bob knows the key K_p and the encryption scheme. Bob can decrypt the ciphertext $\mathrm{c} = [1, 3, 10, 2]$ in the following steps:

(1) He computes $L_2^{-1}(\mathrm{c}) = L_2^{-1}([1, 3, 10, 2]) = [1, 3, 10, 2] = [c_1^0, c_2^0, c_3^0, c_4^0] = \mathrm{c}^0$.

(2) s is odd and Bob treats $[1, 3, 10, 2]$ as a vertex (line) in the graph $D(4, \mathbb{F}_{11})$.

(3) He writes the equation $3x^3 + 5 = c_1^0$ and then rewrites it as $3x^3 = c_1^0 - 5$ (3, x are regular; left side is from $\mathbb{F}_{11}^*$ so $c_1 - 5$ has to be regular). He writes $x^3 = (1 - 5) \cdot 3^{-1}$ (regular) and uses the Euler theorem to find d (a multiplicative inverse of 3 mod $\phi(11)$, i.e., 7).

Then $\eta = [(1-5)3^{-1}]^7 = [7 \cdot 4]^7 = 6^7 = 8$ is the unique solution. He forms a string of elements $\eta_0 = g(\eta) = g(8) = 4$, $\eta_1 = \alpha_1(\eta) = 9$ and $\eta_2 = \alpha_2(\eta) = 5$.

(4) He gets $\mathrm{c}_0 = [1, 3, 10, 2] \longrightarrow N_{\eta_2}(\mathrm{c}_0)(5, 9, 6, 4) \longrightarrow N_{\eta_1}(\mathrm{c}_0)[9, 10, 1, 8] = \mathrm{c}_2 \longrightarrow, N_{\eta_0}(\mathrm{c}_2) = (4, 7, 5, 0) = (c_1^3, c_2^3, c_3^3) = \mathrm{c}_3$.

(5) He computes

$$\Xi_g^{-1}(\mathrm{c}_3) = (\eta, c_2^3, c_3^3) = (8, 7, 5, 0) = \mathrm{c}_3' \quad (35)$$

(6) Finally, he computes the plaintext $\mathrm{p} = L_1^{-1}(c_3') = (1, 2, 5, 0)$.

9.3. Degree Estimates. In the cases $T(n, \mathbb{K}) = W(n, \mathbb{K})$ and $T(n, \mathbb{K}) = \widetilde{D}(n, \mathbb{K})$ the implementations of Algorithm No. 1 and Algorithm No. 2 use encryption map of degree $O(n)$.

The encryption and decryption map have multivariate nature. Let us assume that $\deg(g)$ (odd s) and $\deg(f)$ (even s) are given by linear function of kind $cn + b$, $c > 0$. Then in the case of graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$ the degrees of encryption and decryption maps are bounded below by $cn + b + 2$ and bounded above by $3(cn + b)$ because the shifting encryption is a cubical map.

10. Summary

The paper presents a class of stream ciphers defined in terms of graphs given by equations over the finite commutative ring $\mathbb{K}$. The algorithm has multivariate nature: plaintext is a tuple from the free module $\mathbb{K}^n$, key string is also an element of $\mathbb{K}^m$, and the encryption map is polynomial transformation of $\mathbb{K}^n$ into itself. Users have options to vary parameters n and m and ring $\mathbb{K}$. If the parameter m is bounded by a constant, then the speed of numerical recurrent of encryption is $O(n)$. The key can be given as a sequence of polynomials in a single variable x. We observe results on simplest case of key strings $x + d_1$, $x + d_2$, …, $x + d_s$ obtained by theoretical studies and via computer simulation in case of finite fields or arithmetical rings of kind $\mathbb{Z}_{2^m}$. In case of graphs $D(n, \mathbb{K})$ and $A(n, \mathbb{K})$ simple conditions on d_i ensure that different keys produce distinct ciphertexts and allow estimating the complexity of adversary attacks without access to plaintext. In the above-mentioned case encryption and decryption maps are cubical and adversary after the interception of $O(n^3)$ pairs of kind plaintext-ciphertext can conduct a linearization attack in time $O(n^{10})$. In case of $\widetilde{D(n, \mathbb{K})}$ the degree of both maps grows linearly with the growth of parameter n, which makes the search for the inverse map via linearization attacks a difficult task. Additionally, authors started investigation of bijective and nonbijective encryption maps with keys of kind $x^d + d_1$, $x^d + d_2$, …, $x^d + d_s$, where $d > 1$.

In the nonbijective case the plainspace is large subset of $\mathbb{K}^n$ and the adversary has to restore the multivariate encryption transformation E and search for polynomial map E' such that EE' fixes each plaintext. Known methods do not allow to solve this task in polynomial time. Special case with high degree E' is implemented. Loaded tables for x^d allow a fast encryption of text even in case of large parameter d.

Our general Algorithm No. 2 uses encryption and decryption procedures of polynomial nature. Both procedures are approximated by maps of unbounded degree. This fact leads to resistance of the cipher to linearization attacks by an adversary.

11. Conclusion

The main purpose of this paper is to introduce stream ciphers with nonbijective encryption function of multivariate nature constructed in terms of algebraic graph theory. This class of encryption transformation has already been used as a tool of multivariate cryptography for the construction of public key

candidates to be used in postquantum era. The idea of "hidden Eulerian equation" melted in large variety of coefficients of multivariate public rules was used as heuristic argument of security. We apply this idea in Symmetrical Cryptography which is not endangered by appearance of Quantum Cryptography and introduces robust stream ciphers resistant to linearization attacks. Authors agree that for practical uses further research is needed. Statistical evaluation of mixing properties via measuring of avalanche effect has to be done. We have to compare such properties in the case of Algorithm No. 1 and Algorithm No. 2 and bijective multivariate graph based encryption. Besides linearization attacks other tools have to be used like distinguish, Time Memory Data trade-off attacks, guess-and-determine attack, and various attacks of algebraic nature. Authors believe that the abovementioned attacks which were developed to investigate bijective encryption have to be seriously modified to work with nonbijective ciphers. We hope that our algorithms will attract attention of cryptanalysts and they will test cryptanalytical instruments on such new families of ciphers. We have to compare such properties in the case of Algorithms No. 1 and No. 2 and bijective multivariate graph based encryption. Besides linearization attacks, other tools have to be used like Distinguishing attack, Time/memory/data trade-off attack, Guess-and-Determine attack, Resynchronization attack, and various attacks of algebraic nature (see [22–30]).

Data Availability

Some parts of data (Table 1; General Algorithm No. 1) used to support the findings of this study are included within the article. This paper is an extension of our article no. 1 which reflects our talk at the 5th International Conference on Cryptography and Security Systems, 2018 (one of the events of Federated Conference on Computer Science and Information Systems, FedCSIS 2018).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] V. Ustimenko, U. Romańczuk-Polubiec, A. Wróblewska, M. Polak, and E. Zhupa, "On the implementation of new symmetric ciphers based on non-bijective multivariate maps," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems*, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., vol. 15, pp. 397–405, ACSIS, 2018.

[2] V. Ustimenko, "Coordinatisation of trees and their quotients," in *The Voronoj's Impact on Modern Science*, vol. 2, pp. 125–152, Institute of Mathematics, Kiev, Ukraine, 1998.

[3] P. L. Priyadarsini, "A survey on some applications of graph theory in cryptography," *Journal of Discrete Mathematical Sciences & Cryptography*, vol. 18, no. 3, pp. 209–217, 2015.

[4] F. Lazebnik and V. A. Ustimenko, "Some algebraic constructions of dense graphs of large girth and of large size," in *Expanding Graphs*, vol. 10 of *DIMACS series in Discrete Mathematics and Theoretical Computer Science*, pp. 75–93, 1993.

[5] R. Wenger, "Extremal graphs with no C4, C6 and C10s," *Journal of Combinatorial Theory, Series B*, vol. 52, no. 1, pp. 113–116, 1991.

[6] U. Romańczuk and V. Ustimenko, "On the key exchange with matrices of large order and graph based nonlinear maps," *Albanian Journal of Mathematics*, vol. 4, no. 4, pp. 203–211, 2010.

[7] M. Polak and V. Ustimenko, "On stream cipher based on a family of graphs *D(n,q)* of increasing girth," *Albanian Journal of Mathematics*, vol. 8, no. 2, pp. 37–45, 2014.

[8] J. Ding, J. E. Gower, and D. S. Schmidt, *Multivariate Public Key Cryptosystems*, vol. 25 of *Advances in Information Security*, Springer, 2006.

[9] L. Goubin, J. Patarin, and B.-Y. Yang, "Multivariate cryptography," in *Encyclopedia of Cryptography and Security*, pp. 824–828, 2nd edition, 2011.

[10] J. Patarin, "The oil i vinegar digital signatures," in *Dagstuhl Workshop on Cryptography*, 1997.

[11] A. Kipnis and A. Shamir, "Cryptanalysis of the oil and vinegar signature scheme," in *Advances in Cryptology - Crypto 96*, vol. 1462 of *Lecture Notes in Computer Science*, pp. 257–266, Springer, 1998.

[12] S. Bulygin, A. Petzoldt, and J. Buchmann, "Towards provable security of the Unbalanced Oil and Vinegar signature scheme under direct attacks," in *Progress in cryptology-INDOCRYPT*, G. Gong and K. C. Gupta, Eds., vol. 6498 of *Lecture Notes in Computer Science*, pp. 17–32, Springer, 2010.

[13] U. Romańczuk-Polubiec and V. Ustimenko, "On two windows multivariate cryptosystem depending on random parameters," *Algebra and Discrete Mathematics*, vol. 19, no. 1, pp. 101–129, 2015.

[14] V. Ustimenko, "On algebraic graph theory and non-bijective multivariate maps in cryptography," *Algebra and Discrete Mathematics*, vol. 20, no. 1, pp. 152–170, 2015.

[15] J. S. Kotorowicz, *Kryptograficzne algorytmy strumieniowe oparte na specjalnych grafach algebraicznych [Ph.D. thesis]*, Institute of Fundamental Technological Research, Polish Academy of Sciences, 2014.

[16] A. Wróblewska, "On some properties of graph based public keys," *Albanian Journal of Mathematics*, vol. 2, no. 3, pp. 229–234, 2008, NATO Advanced Studies Institute: "New challenges in digital communications".

[17] A. Wróblewska, *Lingwistyczne układy dynamiczne oparte na grafach algebraicznych i ich zastosowanie w kryptografii [Ph.D. thesis]*, Institute of Fundamental Technological Research, Polish Academy of Sciences, 2016.

[18] M. Klisowski, *Zwiekszenie bezpieczeństwa kryptograficznych algorytmów wielu zmiennych bazujacych na algebraicznej teorii grafów [Ph.D. thesis]*, Czestochowa University of Technology, 2014.

[19] V. Futorny and V. Ustimenko, "On small world semiplanes with generalised Schubert cells," *Acta Applicandae Mathematicae*, vol. 98, no. 1, pp. 47–61, 2007.

[20] M. Polak, *On the applications of algebraic graph theory to coding [Ph.D. thesis]*, Maria Curie-Sklodowska University, 2016.

[21] V. A. Ustimenko, "On new multivariate cryptosystems based on hidden Eulerian equations," *Reports of the National Academy of Sciences of Ukraine*, no. 5, pp. 17–24, 2017.

[22] G. G. Rose and P. Hawkes, "On the applicability of distinguishing attacksagainst stream ciphers," in *Proceedings of the 3rd NESSIE Workshop*, p. 6, 2002.

[23] M. E. Hellman, "A cryptanalytic time-memory trade-off," *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, vol. 26, no. 4, pp. 401–406, 1980.

[24] P. Hawkes and G. G. Rose, "Guess-and-determine attacks on snow," in *Selected Areas in Cryptography, SAC 2002*, I. Nyberg and H. Heys, Eds., vol. 2595 of *Lecture Notes in Computer Science*, pp. 37–46, Springer, 2003.

[25] F. Armknecht, J. Lano, and B. Preneel, "Extending the resynchronization attack," in *Selected Areas in Cryptography*, H. Handschuh and A. Hasan, Eds., vol. 3357 of *Lecture Notes in Computer Science*, pp. 19–38, Springer, 2004.

[26] J. D. Golić, "Linear cryptanalysis of stream ciphers," in *Fast Software Encryption, FSE 1994*, B. Preneel, Ed., vol. 1008 of *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, Germany, 1995.

[27] N. Courtois, A. Klimov, J. Patarin et al., "Efficient algorithms for solving overdefined systems of multivariate polynomial equations," in *Proceedings of the Advances in Cryptology - EUROCRYPT 2000*, B. Preneel, Ed., vol. 1807 of *Lecture Notes in Computer Science*, pp. 392–407, Springer-Verlag, Bruges, Belgium, 2000.

[28] N. T. Courtois and J. Pieprzyk, "Cryptanalysis of block ciphers with overdefined systems of equations," in *Advances in Cryptology - ASI-ACRYPT 2002*, Y. Zheng, Ed., vol. 2501 of *Lecture Notes in Computer Science*, pp. 267–287, Springer, Berlin, Germany, 2002.

[29] N. T. Courtois, "Fast algebraic attacks on stream ciphers with linear feedback," in *Advances in cryptology—CRYPTO 2003*, D. Boneh, Ed., vol. 2729 of *Lecture Notes in Comput. Sci.*, pp. 176–194, Springer, Berlin, Germany, 2003.

[30] F. Armknecht, "Improving fast algebraic attacks," in *Fast Software Encryption*, vol. 3017 of *Lecture Notes in Computer Science*, pp. 65–82, Springer, Berlin, Heidelberg, Germany, 2004.