National Research University Higher School of Economics

As a manuscript

Fomin Denis Bonislavovich

Building Nonlinear Bijective Transformations for Protecting Data Security in Untrusted Devices

DISSERTATION SUMMARY

for the purpose of obtaining academic degree
Doctor of Philosophy in Applied Mathematics

Academic supervisor:
Candidate of Sciences in Physics and Mathematics
Nesterenko Alexei Yuryevich

Moscow — 2023

Introduction

This dissertation research is devoted to solving the problem of increasing the security level of algorithmic methods of information protection, in the synthesis of which, among other things, it is necessary to use nonlinear bijective transformations (permutations). It is necessary to guarantee their cryptographic characteristics. Thus functioning of algorithms of data privacy protection takes place in conditions of untrusted environment, namely at possibility of an intruder to get information about time of operations execution, that imposes additional requirements to designs of nonlinear bijective transformations. Without losing generality, this dissertation research examines only data privacy issues.

Cloud technologies, which is usually owned by a service provider, are increasingly being developed. The degree of trust in cloud infrastructure depends on the protections used, which are shaped by an adversary model that requires extensive research to investigate the adequacy of that model. Due to the presence of hardware computing facilities on the service provider side, an adversary has additional opportunities related to the possibility of using information from side channels. One of the most common such channels is the execution time of operations on a computing device.

In the dissertation, a study of the adversary's capabilities, leading to the impossibility of ensuring the property of confidentiality of data when implementing the algorithms for their protection in an untrusted environment on graphical computers is carried out. In particular, it is shown that the software tools used at the time of the study, implementing them, are potentially not secure, which leads to the need to use other ways of their implementation.

For effective implementation of data encryption algorithms, the possibility to represent its nonlinear transformations in the form of “small” number of logical operations is necessary. At the same time, the security of the algorithm directly depends on the nonlinearities of the permutations used in its synthesis. In the dissertation research the problems of effectively realized nonlinear bijective transformations having “high” cryptographic properties, which allows to use them in the synthesis of algorithms for data encryption are considered. The thesis also proposes a new method for analyzing block ciphers, based on the peculiarities of the used nonlinear transformations. An approach is proposed and justified, which allows to estimate the security of the used encryption algorithm.

Relevance. The increase in the number of computing devices leads to the need to process large amounts of data. At the same time there is a constant transfer, processing, storage of confidential information, for the protection of which various software, hardware and organizational measures are used. Ensuring the confidentiality of information for a large amount of time is a complex and urgent task, the solution of which is associated with the synthesis of information protection algorithms.

An additional difficulty in ensuring the properties of data security is the possibility of an adversary to obtain additional information from the side channels. In 1995, an article [46] was published, which showed the fundamental possibility of using the execution time of operations to recover information unknown to the adversary. At present it is necessary to take into account such capabilities of the adversary, which is reflected in the regulatory documents [4; 5].

Both the security of the information system as a whole and the complexity of ensuring its correct functioning depend on the correct choice of the adversary model. Insufficient requirements can lead to a violation of the security properties of the information system, while overstated requirements — to the impossibility or high complexity of its use.

The possibility of implementing attacks that use information about the execution time of operations on the computer's CPU is well known,[46]. Moreover, the use of modern standardized algorithms of data encryption is not safe with respect to such attacks (see [8; 13]). The architecture of graphics coprocessors is very different from the architecture of the modern computer's CPU: they are massively parallel computing devices simultaneously implementing a large number of simple operations.

Thus, the task of investigating the possibility of applying attacks using information from side channels in general and, in particular, by time of operations during the evaluation of data protection algorithms on GPU is relevant. Moreover, in work [48] it is spoken about potential possibility of such attacks, however it is supposed that their application will not be effective or can be impossible at the minimum change of a way of realization of algorithm of protection of confidentiality of data.

One of the ways to protect against analysis timing attacks is to implement transformations without storing special substitution tables, which allow efficient implementation of data protection algorithms [6; 29; 50; 72]. This requires representing all transformations using logical operations, which leads to high running times of the data protection algorithms without additional optimizations.

In [9] a new approach to such optimizations was proposed, which consists in the fact that instead of performing operations on a single argument by using the registers of the computing device, parallel calculation of the values of the encryption algorithm is possible. In this case, the greater the length of the register of the computing device, the higher the performance of such an implementation. In the literature, this method is called the bitslice implementation and is currently actively used for data encryption, [7; 69], which ensures that the timing attacks are impossible in the case of both the CPU and the GPU, [62].

The efficiency of bitslice implementations directly depends on the number of logical operations needed to compute the nonlinear transformation. Thus, to be able to use bitslice implementations of an encryption algorithms, nonlinear transformations must be «easy to implement», which suggests that they are expected to meet similar requirements to those of low-resource nonlinear transformations, [57]. At the same time, the synthesis of permutations for low-resource devices uses either low-dimensional permutations (over $\mathbb{F}_2^4$, $\mathbb{F}_2^5$) a priori easier to implement than high-dimensional permutations, or nonlinear bijective transformations which are represented by a composition of functions with arguments of lower dimensionality, [14; 18; 63]. Currently, there are three main universal approaches to constructing nonlinear bijective transformations in the form of logic elements: full search using graph traversal in depth and using the meet-in-the-middle method ([21; 40; 78]), heuristic search ([80; 37; 40; 71]) and the use of algebraically defined permutations (in particular, inverse permutations), [63].

At the same time, the synthesis of encryption algorithms must use nonlinear transformations that allow to confront the known methods of analysis. The effectiveness of linear attacks [36; 45; 59], differential attacks [10; 36] and some types of algebraic attacks [23; 39; 68; 76] directly depends on the cryptographic properties of the permutations [19]:

- nonlinearity;
- differential δ-uniformity;
- algebraic degree and algebraic degree of inverse permutation;
- graph algebraic immunity.

The cryptographic properties of low-dimensional permutations are well studied, but they are far from similar values even for large-dimensional random permutations.

Thus, their use in the synthesis of promising encryption algorithms, requires the implementation of a larger number of permutations while maintaining the same security level.

In view of the above, there are many reasons for constructing higher-dimensional permutations using functions defined over lower-dimensional spaces:

- is possible to implement in software with substitution tables (T-tables),
- is possible to implement a transformation with a small number of logical operations,
- is possible to use these permutations to implement ligtweight algorithms,
- is possible to use effective hardware masking [14; 51].

There are a large number of ways to build such nonlinear transformations: based on the Feistel network [18; 33; 54], using a Misty network [18; 34; 58], SPN networks [53; 70; 75] or other designs [74]. At the same time, for the nonlinear bijective transformations listed above, the cryptographic characteristics are usually not higher than those obtained by random search.

Thus, the relevant task is to design non-linear bijective transformations, which can be represented using functions of arguments of lower dimensionality, whose non-linearity indices will be better than those of similar ones obtained by random search. One such way is based on the use of a “butterfly” type construction, which was proposed in [67] during the study of the possibility of decomposition of the well-known 6-bit differentialy 2-uniform permutation [15] and the method of decomposition construction for the nonlinear transformation of Russian cryptographic standards [11].

Constructive features of permutations defined by the “butterfly” type construction may affect the security of the data confidentiality algorithm. According to paragraph 27 of the Doctrine of Information Security of the Russian Federation [3] in the synthesis of information security algorithms must be at the stage of development to guarantee the impossibility of various attacks, which indicates the relevance of research of the influence of the properties of the proposed nonlinear bijective transformations on the security of the algorithm as a whole.

The aim of this work is to improve algorithmic methods of information security, when they are functioning in an untrusted environment, through the development of new principles of nonlinear bijective transformations.

In order to achieve this goal, it was necessary to solve the following **tasks**:

1. Evaluate the security level of encryption algorithms when implemented on heterogeneous platforms, allowing an intruder to obtain information about the execution time.
2. Development of new ways to build nonlinear bijective transformations, the use of which reduces the effectiveness of the known and attacks represented in this work. Evaluation of the main cryptographic characteristics of the proposed permutations.
3. Development of new attacks based on the properties of the proposed nonlinear bijective transformations.

Academic novelty:

1. A new attack on data privacy algorithms implemented on GPU, using information about the execution time, is proposed.
2. New classes of nonlinear bijective transformations are proposed, and algorithms for their construction are developed.
3. A new invariant attack on XSL-networks is proposed.

The theoretical value of this dissertation research lies in the development of mathematical methods and models used in the synthesis and analysis of symmetric encryption algorithms, including their implementation in untrusted environments.

An attack on encryption algorithms implemented on GPU, using information about the execution time, is proposed. It is shown that in spite of the high complexity and lack of published data about the architectural features of the modern GPU platforms, it is possible to propose an attack, which allows to restore sequentially the unknown to the advertiser parameters.

Using the discrete function theory, as well as finite field theory, new families of nonlinear bijective transformations and algorithms of their construction are proposed and estimates of their cryptographic characteristics are obtained. The developed mathematical apparatus makes it possible to guarantee properties for a sufficiently wide family of nonlinear bijective transformations.

A new invariant attack on XSL-networks, using the apparatus of graph theory and linear algebra, is proposed. With its use it is possible to obtain estimates of resistance for modern and promising encryption algorithms or to guarantee the impossibility of such attacks.

The practical significance of this dissertation research is as follows:

1. The new attack showed low security of the AES algorithm when implemented in an untrusted environment on GPU using pre-calculated tables, as confirmed by independent studies [20; 62]. It is shown that for the Russian standardized encryption algorithm [2], the application of the specified method is not effective.
2. New families of nonlinear bijective transformations, allow their use in the synthesis of promising algorithms for data privacy protection.
3. The security level of the standardized encryption algorithm [2] is not reduced with respect to the attack developed in this research.
4. The results obtained in the dissertation study can be used in the educational process in the preparation lecture courses.

The thesis research applies the mathematical apparatus and approaches of various sections of mathematics, such as discrete mathematics, finite field theory, graph theory, algorithm theory, as well as experimental studies of the proposed primitives and algorithms.

The main results:

1. An attack on special form encryption algorithms implemented on GPU, based on the information about the execution time.
2. New parametric families of permutations and evaluation of their cryptographic characteristics: nonlinearity, algebraic degree, differential δ-uniformity.
3. A new invariant attack on XSL-network based encryption algoritms.

The reliability of the obtained results is confirmed by the correctness of the problem statement and the applied research methods, provided by rigorous mathematical proofs of the statements and confirmed by their consistency with the results of experimental studies. In addition, the results obtained in this dissertation research are in accordance with the results obtained by other authors:

- In works [20; 62] points out that the timing attacks on symmetric encryption algorithms of special kind, implemented on GPU, is applicable in the practical implementation of cloud computing.
- Later works also investigated the issue of using the execution time of operations in the implementation of symmetric encryption algorithms, but using a different approach (see e.g. [31; 41—43]).

- Independent author [24; 25] obtained nonlinear bijective transformations similar to the one parametric family of permutations presented in the thesis, and having the same cryptographic characteristics.
- Experimental studies of the nonlinear bijective transformations proposed by the author were independently investigated in [30; 47].
- In [17], in particular, it is shown that the round transformations of the Kuznyechik algorithm do not have nonlinear invariants of special form, which does not contradict the results obtained by the author.

Approbation of the obtained results. The main results of the thesis research were presented at the following international and All-Russian conferences:

- 2023, The 12th Workshop on Current Trends in Cryptology CTCrypt 2023 (Volgograd), June 6-9, 2023, report: “Computational work for some TU-based permutations”.
- 2023, The 12th Workshop on Current Trends in Cryptology CTCrypt 2023 (Volgograd), June 6-9, 2023, report: “On one way of constructing unbalanced TU-based permutations”.
- 2021, The 23th “RusCrypto’2021” (Solnechnogorsk), March 23-26, 2021, report: “Resistance of the Kuznyechik algorithm to a generalized invariant attack”.
- 2021, The 10th Workshop on Current Trends in Cryptology CTCrypt 2021 (Ruza), June 1-4, 2021, report: “On Differential Uniformity of Permutations Derived Using a Generalized Construction”.
- 2021, The 10th Workshop on Current Trends in Cryptology CTCrypt 2021 (Ruza), June 1-4, 2021, report: “On the Impossibility of an Invariant Attack on Kuznyechik”.
- 2021, The 20th International Conference ”Siberian Scientific School-Seminar ”Computer Security and Cryptography SIBECRYPT’21 named after G.P. Agibalov (Novosibirsk), September 6-11, 2021, report: “On the way of constructing differentially 2δ-uniform permutations over $F_{2^{2m}}$”.
- 2021, The 20th International Conference ”Siberian Scientific School-Seminar ”Computer Security and Cryptography SIBECRYPT’21 named after G.P. Agibalov (Novosibirsk), September 6-11, 2021, report: “On a heuristic approach to constructing bijective vector Boolean functions with given cryptographic properties”.

- 2020, The 9th Workshop on Current Trends in Cryptology CTCrypt 2020 (Ruza), September 15-17, 2020, report: “A compact bit-sliced representation of Kuznechik S-box”.
- 2019, The 8th Workshop on Current Trends in Cryptology CTCrypt 2019 (Svetlogorsk), June 3-7, 2019, report: “On the Way of Constructing $2n$-Bit Permutations from n-Bit Ones”.
- 2019, The 18-th All-Russian Conference ”Siberian Scientific School-Seminar with International Participation ”Computer Security and Cryptography”SIBECRYPT’19 (Tomsk), September 9-14, report: “Hardware implementation of one class of 8-bit permutations”.
- 2018, VII Workshop on Current Trends in Cryptology CTCrypt 2018 (Suzdal), May 28-30, 2018, report: “New classes of 8-bit permutations based on a butterfly structure”.
- 2018, XIX All-Russian Symposium on Applied and Industrial Mathematics (autumn session) (Sochi), September 22-30, 2018, report: “On approaches to the construction of lightweight nonlinear transformations”.
- 2015, The 4th Workshop on Current Trends in Cryptology CTCrypt 2015 (Kazan), June 3-5, 2015, report: “A timing attack on CUDA implementations of an AES-type block cipher”.

Content of work. The results of the thesis research can be divided into the following chapters:

1. Evaluation of the possibility of using the information about the execution time of operations to violate the security properties of symmetric encryption algorithms which are implemented on GPUs.
2. Selection of primitives for the construction of non-linear bijective transformation to ensure the effectiveness of software and hardware implementation.
3. Construction of parametric families of permutations and evaluation of their cryptographic characteristics.
4. Evaluation of the impact of design features of the proposed parametric families of permutations on the security of symmetric encryption algorithms.

Let us outline the main results of the thesis research. To begin with, we introduce the necessary notations.

We will call a two-element field a set $\mathbb{F}_2 = \{0,1\}$ with naturally given operations of addition "+" and multiplication "·". Let $(\mathbb{F}_2^n, +) = \{(a_0, a_1, \ldots, a_{n-1}), a_i \in \mathbb{F}_2, i \in \overline{0,n-1}\}$ — arithmetic vector space of dimension n, $\theta = (0,0,\ldots,0)$ — zero of vector space. If we consider the additive group of vector space $(\mathbb{F}_2^n, \oplus)$ and specify the multiplication operation in a special way, we can construct a finite field which we denote by $(\mathbb{F}_{2^n}, +, \cdot)$.

First chapter of this dissertation research is devoted to the study of the principle capability of an adversary to recover unknown data using information from side channels.

Historically, the first attack using information from side channels is the recovery of key information of public key data algorithms [46] by their running time on a CPU. With the development of science, new methods of analysis using the information from a variety of information leakage sources: signals in the power supply channel, signals from electromagnetic radiation, temperature information or sounds made by the device [49; 79].

Prior to 2015, there were no known attacks on cryptoalgorithms implemented on GPU, based on information from side channels of leakage. At the CTCrypt'15 conference, the first such analysis method was presented, where information about the execution time of the encryption algorithm was used as a side channel. Later, in October 2015, at the ICCD'15 conference, authors presented a paper, [73], which proposed an attack using information obtained from the power circuit of a GPU.

The first chapter is devoted to the description of the attack presented at CTCrypt'15 and later published in [81]. The subject of the first chapter was the implementation of an encryption algorithm implemented on a GPU using the most efficient approach based on the precomputed tables, [29; 38; 44]. The object of the study is — an XSL network based block cipher. For the purpose of this chapter, we will assume that n is a block size in bits, m is a number of permutations on subvectors of length n', $n = n' \cdot m$.

The following transformations are used in the implementation of the algorithms from the proposed family:

- Addition of an unknown parameter, called a round key: $\mathrm{X}[K] \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, где $\mathrm{X}[K](a) = K \oplus a$; $a, K \in \mathbb{F}_2^n$.
- A nonlinear transformation that is a parallel application of the permutations of the space $\mathbb{F}_2^{n'}$: $\mathrm{S} \colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, $\mathrm{S}(a) = \mathrm{S}(a_1, \ldots, a_s) = (\pi(a_1), \ldots, \pi(a_m))$, $a_i \in \mathbb{F}_2^{n'}, \pi \in S\left(\mathbb{F}_2^{n'}\right), i = 1, \ldots, m$.

– Linear transformation: $\mathrm{L}\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$, $\mathrm{L}(a) = a \cdot L'$, where $L' \in \mathrm{GL}_n$.

When implementing the symmetric encryption algorithm, the operations described above are iteratively applied, with each iteration called a round.

In the case where m is the square of some natural number $m = k^2$, we can assume that transformations of algorithms from the proposed family are performed over the elements $k \times k$ of the matrix:

$$\begin{pmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,k-1} \\ x_{1,0} & x_{1,1} & \cdots & x_{1,k-1} \\ \cdots & \cdots & \cdots & x_{k-1,k-1} \\ x_{k-1,0} & x_{k-1,1} & \cdots & x_{k-1,k-1} \end{pmatrix},$$

$x_{i,j} \in \mathbb{F}_2^{n'}$, $i,j = \{0, 1, \ldots, k-1\}$. In this case, a linear transformation can be defined by a composition of two transformations: multiplication of some reversible matrix by each column of the matrix and permutation of matrix elements so that each row and each column of the resulting matrix contains an element of each of k columns of the original matrix. Such linear transformations will be called A-type transformations. The AES encryption algorithm, [26] and GOST 34.11-2018 [1] hash function use A-type linear transformations. Within this chapter, the intermediate value obtained after the i-th iteration of the encryption algorithm under consideration will be denoted by

$$x^{(i)} = \begin{pmatrix} x^{(i)}_{0,0} & x^{(i)}_{0,1} & \cdots & x^{(i)}_{0,k-1} \\ x^{(i)}_{1,0} & x^{(i)}_{1,1} & \cdots & x^{(i)}_{1,k-1} \\ \cdots & \cdots & \cdots & x^{(i)}_{k-1,k-1} \\ x^{(i)}_{k-1,0} & x^{(i)}_{k-1,1} & \cdots & x^{(i)}_{k-1,k-1} \end{pmatrix},$$

$x^{(i)}_{i,j} \in \mathbb{F}_2^{n'}$, $i,j = \{0, 1, \ldots, k-1\}$, $x^{(0)}$ is a plain text. For ease of explanation, we will use the notation $x^{(i)}[j,k]$ for the value $x^{(i)}_{j,k}$, $j,k = \{0, 1, \ldots, k-1\}$.

Proposition 1. *[81] For the encryption algorithm with the A-type linear transformation an arbitrary value $x^{(2)}_{i,j}$, $i,j \in \{1, \ldots, k-1\}$, after the first round is defined by exactly k values on the first round of transformations and in order to fix the value $x^{(2)}_{i,j}$ as a constant, there are $2^{n' \cdot k-1}$ ways to choose these k values after the X operation on the first round of the algorithm.*

Consider the AES algorithm for which $k = 4$. Denote the value obtained after the key overlay in the first round by a matrix:

$$\begin{pmatrix} x_{0,0} & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,0} & x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,0} & x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,0} & x_{3,1} & x_{3,2} & x_{3,3} \end{pmatrix},$$

$x_{i,j} \in \mathbb{F}_2^8$, $i,j = \{0, 1, \ldots, 3\}$. According to the statement 1 there are 2^{24} ways to set the value $x_{i,0}^{(2)}$ by some constant for arbitrary $i \in \{0, \ldots, 3\}$ by setting the values $x_{0,0}, x_{1,1}, x_{2,2}, x_{3,3}$ in a special way. Other than that:

can be set as a constant:	by setting in a special way:
$\left\{x_{i,0}^{(2)}, i = 0, \ldots, 3\right\}$	$\{x_{0,0}, x_{1,1}, x_{2,2}, x_{3,3}\}$
$\left\{x_{i,1}^{(2)}, i = 0, \ldots, 3\right\}$	$\{x_{0,1}, x_{1,2}, x_{2,3}, x_{3,0}\}$
$\left\{x_{i,2}^{(2)}, i = 0, \ldots, 3\right\}$	$\{x_{0,2}, x_{1,3}, x_{2,0}, x_{3,1}\}$
$\left\{x_{i,3}^{(2)}, i = 0, \ldots, 3\right\}$	$\{x_{0,3}, x_{1,0}, x_{2,1}, x_{3,2}\}$

and it is possible to do it in 2^{24} ways.

To recover the key in the first round, the following property of shared memory used to store replacement tables applies: 32 threads access memory at the same time, all memory cells are divided into 32 banks, the access speed of which is directly proportional to the maximum number of accesses to the same memory bank. In this case, the memory access speed will be minimal when all threads access the same memory bank, or when no pair of threads accesses the same memory bank, [64]. Using this property, it is possible to recover all k of key values having length $n' \cdot k$. Thus, $2^{n' \cdot k}$ by different ways of choosing the input data of encryption algorithm can determine the time when the speed of implementation was minimal, which allows to recover the key in the first round. Similarly, it is possible to recover the rest of the round keys. As an illustration, let us describe the way of recovery of the first round key

$$K = \begin{pmatrix} k_{0,0}^{(1)} & k_{0,1}^{(1)} & k_{0,2}^{(1)} & k_{0,3}^{(1)} \\ k_{1,0}^{(1)} & k_{1,1}^{(1)} & k_{1,2}^{(1)} & k_{1,3}^{(1)} \\ k_{2,0}^{(1)} & k_{2,1}^{(1)} & k_{2,2}^{(1)} & k_{2,3}^{(1)} \\ k_{3,0}^{(1)} & k_{3,1}^{(1)} & k_{3,2}^{(1)} & k_{3,3}^{(1)} \end{pmatrix}$$

of AES algorithm (see algorithm 1). Realization of the random variable x that has a uniform distribution on the set D will be denoted by: $x \overset{U}{\leftarrow} W$, the function that calculates the time of the encryption procedure of the algorithm E_K of data $D_1, D_2, \ldots, D_M$ will be denoted by time $(E_K, \{D_1, D_2, \ldots, D_M\})$. When describing the algorithm, we will also use the integer division operation "\".

Algorithm 1: Recovering keys $k^1_{0,0}$,$k^1_{1,1}$,$k^1_{2,2}$,$k^1_{3,3}$ of AES algorithm

```
Data: Black box $E_K$, implementing the AES encryption algorithm, $M$ is a
      parameter
for $i = 1$ to $M$ do
    for $j = 0$ to $3$ do
        for $k = 0$ to $3$ do
            $x_{j,k} \overset{U}{\leftarrow} \mathbb{F}_2^8$
    $D_i \leftarrow \begin{pmatrix} \theta & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,0} & x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,0} & x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,0} & x_{3,1} & x_{3,2} & x_{3,3} \end{pmatrix} \cdot L^{-1}$
$T_0 =$ time $(E_K, \{D_1, D_2, \ldots, D_M\})$
for $n = 1$ to $2^{32} - 1$ do
    for $i = 1$ to $M$ do
        for $j = 0$ to $3$ do
            $D_i[j,j] \leftarrow D_i[j,j] \oplus \left(\left((n \oplus (n-1)) \backslash 2^{8 \cdot j}\right) \pmod{2^8}\right)$
    $T_n =$ time $(E_K, \{D_1, D_2, \ldots, D_M\})$
Result: $\underset{n \in \{0,1,\ldots,2^{32}-1\}}{\arg\min} T_n$
```

The values $k^1_{0,0}$,$k^1_{1,1}$,$k^1_{2,2}$,$k^1_{3,3}$ are uniquely determined by the value given by the algorithm 1. Similarly, the value of the entire key of the first round is recovered.

Thus, the complexity of key recovery of the considered algorithm is equal to $2^{n'k}$ operations of calculating values of a special kind. The parameter M of the algorithm 1 is chosen experimentally. In the paper [81] the data and graphs showing the practical applicability of the proposed method of analysis are given.

Thus, it is shown that the use of precalculated replacement tables to implement symmetric encryption algorithms, potentially allows an adversary to recover unknown parameters. Moreover, to implement side-channel attacks, the memory access operation is often studied, as it is considered one of the most informative for the adversary, [28; 31; 35; 55; 73].

It follows from the above that to protect against the considered attacks and a number of other analysis methods, it is necessary to use an implementation of transformations for which there is no need to access memory, that is, using only logical operations (e.g., bitslice-realization). However, it can lead to low implementation speed of data privacy protection algorithms. Thus, it is necessary to choose a nonlinear bijective transformation design with guaranteed availability of an efficient implementation.

The second chapter of the thesis research is devoted to the choice of primitives for the construction of a nonlinear bijective transformation, allowing to guarantee the efficiency of software and hardware implementation.

In 2016, a group of authors investigated the possibilities of representing some permutations with "good" cryptographic properties using functions defined over spaces of lower dimensionality. As a result of the study, an approach to construct permutations based on the so-called TU-representation [12; 66], which in some sense can be considered a generalization of the two-round Feistel network, was proposed. The permutations based on this principle will be called F-constructions (Feistel-like constructions). The nonlinear bijective transformation used in domestic standardized algorithms as well as the CCZ-equivalent permutation of the only known differential 2-uniform permutation of spaces of the form $\mathbb{F}_2^{2m}$, have a TU representation, [12; 66; 67].

At present, the most promising in terms of implementation of data privacy protection algorithms is the use of transformations from the symmetric group $S\left(\mathbb{F}_2^8\right)$, which have "good" nonlinearity indexes. The nonlinear bijective transformation of Kuznyechik algorithm is represented as a composition of the following functions (see figure 1): function $\mathbb{F}_2^4 \to \mathbb{F}_2^4$, permutations from the symmetric group $S\left(\mathbb{F}_2^4\right)$, multiplication in the field $\mathbb{F}_{2^4}$, multiplexer (conditional operator), linear functions GL_8.

Let us consider a way to determine the complexity of each of these functions. In [78] the search and construction of all permutations of the space $\mathbb{F}_2^4$ realized in less than 12 logical operations, as well as copying operations `MOV`. This represents about 90% of all permutations, and efficiently implemented representatives were found for 271 out of 302 affine equivalence classes. A differentially 4-uniform permutation was constructed, which can be realized by only 9 operations, has a nonlinearity equal to 4, and an algebraic degree equal to 2. In [78], not all 4-bit permutations have been constructed, and the question of the minimal representation of a sufficiently broad class of permutations remains open at present. Among the 16 classes of affine equivalence having a differential uniformity and nonlinearity equal to 4, only for 7 classes were efficiently implemented representatives found (from 9 to 13 operations `AND`, `OR`, `XOR`, `NOT`, `MOV`).

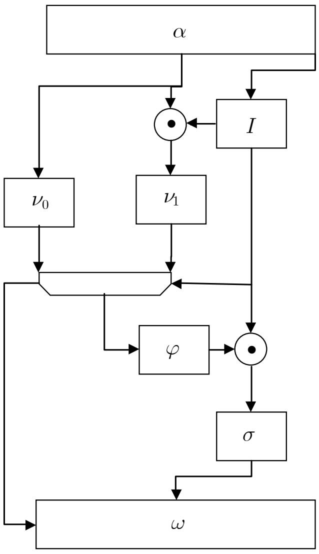

Рисунок 1 — TU-representation of Kuznyechik algorithm [12]

In [22] the author explores a different base, consisting only of operations AND and XOR. The rejection of the MOV operation was due to the fact that the authors are focused primarily on a hardware implementation, where the copy operation does not require additional resources. Limiting the basic operations to AND and XOR is due to the fact that using only these operations facilitates the creation of the so-called threshold implementation of permutations that allows to counteract the methods that use information from the side channels. Russian specialists in [21] present the results of a study devoted to the search of $\mathbb{F}_2^4$ space permutations implemented in the AND and XOR bases in more than 11 instructions. For a number of permutations for which no efficiently implemented representatives were found in [78], the authors of [21] give estimates of the number of instructions used. It is worth noting that, in general, it is not entirely correct to compare the results of [78] and [21] due to differences in the basis. For another 7 classes of 16 having a differential and nonlinearity index equal to 4, effective representatives were found (but already in the AND and XOR bases).

It is worth noting that monomial permutations of the $\mathbb{F}_2^4$ space are either linear (the labor intensity of their implementation is easy to determine), or are linearly equivalent to the inverse permutation, the implementation of which has been well studied, [61; 65; 77].

In general, however, finding an efficient implementation of a particular function or permutation $\mathbb{F}_2^4 \to \mathbb{F}_2^4$ is a rather time-consuming task, [16]. In this regard, various heuristic algorithms are often used in solving this problem. The most well-known algorithms are ESPRESSO [71] and BOOM [37], which are implemented in a large number of commercial packages. Russian authors presented their analogue, which was used in the search for a minimal permutation representation of the Kuznyechik algorithm in [80].

Multiplication in the field is obviously a quadratic form and is effectively implemented, [80]. Separately, it is necessary to consider the labor intensity of multiplexer implementation. According to [12] the following calculations take place: "$\texttt{If}\, r = \theta \,\texttt{then}\, l = \nu_0(l) \,\texttt{otherwise}\, l = \nu_1(l \cdot I(r))$", where ν_0, ν_1, I are nonlinear bijective functions of space $\mathbb{F}_2^4$. Consider an indicator function taking a value equal to 1 at the point $r = \theta$ and zero at all other points:

$$\mathrm{Ind}_\theta(r) = \bar{r}_1 \cdot \bar{r}_2 \cdot \bar{r}_3 \cdot \bar{r}_4 = \overline{r_1 + r_2 + r_3 + r_4}.$$

It is implemented in 4 logical operations, its negation — in 3. Then the calculation of l is done by the equation:

$$l = \mathrm{Ind}_\theta(r) \cdot \nu_0(l) + \overline{\mathrm{Ind}_\theta(r)} \cdot \nu_1(l \cdot I(r)).$$

The calculation of this function can be simplified as follows:

$$l = \mathrm{Ind}_\theta(r) \cdot (\nu_0(l) \oplus \nu_1(\theta)) \oplus \nu_1(l \cdot I(r))$$

since I is a monomial permutation. Obviously, the implementation is the more efficient the smaller the weight of the value $\nu_1(\theta)$.

Thus, we choose as primitives for constructing the permutations of $\mathbb{F}_2^8$ space the permutations of $\mathbb{F}_2^4$ space, as well as the multiplication operation in the field $\mathbb{F}_{2^4}$ and multiplexer. It is desirable that the permutations of the $\mathbb{F}_2^4$ space be either monomial, have a fixed point at zero, or be linear. Obviously, the smaller number of nonlinear transformations used leads to higher implementation efficiency.

The permutations considered below at different values of parameters are implemented using from 2 to 6 permutations of the space $\mathbb{F}_2^4$. In this case, the most effective permutation $F(x_1, x_2) = (y_1, y_2)$ (in terms of the number of nonlinear transformations used) is the bijective function defined by the following equations:

1. $x' = x^{-1}$
2. $y' = y^{-1}$
3. $x'' = x \cdot y'$
4. $y'' = x' \cdot y'$
5. $\texttt{if}\, x = \theta \,\texttt{then}\, y_1 = y' \,\texttt{else}\, y_1 = y''$
6. $\texttt{if}\, x = y \,\texttt{then}\, y_2 = x' \,\texttt{else}\, y_2 = x''$

In addition, the results of [92] show the efficiency of implementing the nonlinear bijective transformations defined in the thesis study on hardware platforms.

The third chapter of this dissertation research is devoted to the construction of nonlinear bijective transformations and evaluation of their cryptographic properties. We describe the main cryptographic characteristics of nonlinear bijective transformations used in the analysis of symmetric cryptographic algorithms, [19].

When designing encryption algorithms, it is necessary to use permutations to resist known methods of analysis. The effectiveness of linear [36; 45; 56; 59], difference [10; 36; 56] and some types of algebraic analysis methods [23; 39; 68; 76] depends directly on the cryptographic characteristics of the nonlinear transformations used in the algorithm. Such characteristics (for a fixed value of n) are as follows:

- non-linearity of the permutation;
- differentially δ-uniformity of the permutation;
- algebraic degrees of permutation and inverse permutation;
- algebraic immunity of the permutation.

Definition 1 ([82]). *Let F_1, $F_2\colon \mathbb{F}_2^m \times \mathbb{F}_2^m \to \mathbb{F}_2^m$ — arbitrary $(2m,m)$-functions. Let us define the transformation $F(x_1, x_2) = (y_1, y_2)$, which we will call F-construction (see figure 2), by the following system:*

$$\begin{cases} y_1 = F_1\left(x_1, x_2\right) \\ y_2 = F_2\left(x_2, y_1\right) \end{cases}. \qquad (1)$$

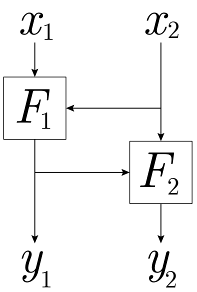

Рисунок 2 — F-construction

The F-construction is the basis for the construction of nonlinear bijective transformations in this dissertation study.

Proposition 2 ([12; 82]). *Let F_1, $F_2\colon \mathbb{F}_2^m \times \mathbb{F}_2^m \to \mathbb{F}_2^m$ be such functions for which, given a fixed arbitrary z_2 function $F_i(z_1,z_2)$, $i \in \overline{1,2}$ is bijection on the variable z_1. Then*

1) transformation F is a permutation on the set $\mathbb{F}_2^m \times \mathbb{F}_2^m$,

2) *the total number of permutations F that are defined by the equations* (1) *is* $(2^m!)^{2^{m+1}}$.

Thus, to set nonlinear bijective transformations defined by F-construction, one must choose F_1, F_2-functions satisfying the statement 2. The construction of balanced (n,m)-functions having nonlinearity not lower than some boundary $\mathbf{N}$ is a difficult problem at large values of the boundary $\mathbf{N}$ and at $n \geqslant 8$ and $m \geqslant 4$. One well-known approach is to construct balanced (n,m)-functions from unbalanced (n,m)-functions with high nonlinearity (see e.g. [27]), which is investigated in the thesis study by choosing F_i, $i \in \overline{1,2}$ functions in the equation (1).

Let $s'(x,y)$ be a function of $\mathbb{F}_2^m \times \mathbb{F}_2^m \to \mathbb{F}_2^m$ and for some $\dot{y} \in \mathbb{F}_2^m$, the function $s'(x,\dot{y})$ is not a permutation on the variable x. Then such a point $\dot{y}$ will be called a punctured point of the function s'. The set of punctured points of the function s' will be denoted by $\dot{Y} \subseteq \mathbb{F}_2^m$:

$$\dot{Y} = \{\dot{y}\colon\ |\{s'(x,\dot{y}), x \in \mathbb{F}_2^m\}| < 2^m\}\,.$$

In the case where $\dot{Y}$ is not empty, we can redefine a function at each punctured point $\dot{y} \in \dot{Y}$ and construct a new function $s(x,y)$ such that $s\colon \mathbb{F}_2^m \times \mathbb{F}_2^m \to \mathbb{F}_2^m$ is a permutation on the variable $x \in \mathbb{F}_2^m$ while fixing an arbitrary value $y \in \mathbb{F}_2^m$. Let $\widehat{\pi}_y(x)$, $y \in \dot{Y}$ be permutations of $\mathbb{F}_2^m$ space, then set $(2m,m)$-function s as follows:

$$s(x,y) = \begin{cases} s'(x,y), & y \notin \dot{Y} \\ \widehat{\pi}_y(x), & y \in \dot{Y} \end{cases}. \tag{2}$$

In order to estimate the nonlinearity of the function $s(x,y)$ it is necessary to be able to calculate the Walsh–Hadamard transform of this function. For the functions $s'(x,\dot{y})$, $\dot{y} \in \dot{Y}$, as functions of one variable x, let us introduce the notation $g_{\dot{y}}(x)$.

Proposition 3 ([82]). *Let $s'(x,y)$ — $(2m,m)$-function with a set of punctuated points $\dot{Y}$, $\widehat{\pi}_{\dot{y}}$ — set of permutations on $\mathbb{F}_2^m$, $\dot{y} \in \dot{Y}$. Let us define $(2m,m)$-function $s(x,y)$ with no punctured points by the equation* (2)*. Let α, $\beta, \gamma \in \mathbb{F}_2^m$, then the coefficients of the Walsh–Hadamard transform s be calculated by the following equation:*

$$W_s^{\alpha\|\beta,\gamma} = \begin{cases} W_{s'}^{\alpha\|\beta,\gamma} + \sum\limits_{\dot{y}\in\dot{Y}} (-1)^{\langle\beta,\dot{y}\rangle} \left(W_{\widehat{\pi}_{\dot{y}}}^{\alpha,\gamma} - W_{g_{\dot{y}}}^{\alpha,\gamma}\right), & \alpha \neq \theta \\ W_{s'}^{\alpha\|\beta,\gamma} + \sum\limits_{\dot{y}\in\dot{Y}} (-1)^{\langle\beta,\dot{y}\rangle} \left(2 \cdot wt\left(\langle\gamma, g_{\dot{y}}(x)\rangle\right) - 2^m\right), & \alpha = \theta, \gamma \neq \theta \\ W_{s'}^{\alpha\|\beta,\gamma}, & \alpha = \theta, \gamma = \theta \end{cases}. \tag{3}$$

We can obtain the following estimate for the linearity of the function s, constructed using the equation 2.

Corollary 1 ([82]). *Under the conditions of the statement 3, the following upper bound on the linearity of the L_s function s is:*

$$L_s \leqslant \max\left\{L_{s'} + \sum_{\dot{y}\in\dot{Y}} \left(L_{\widehat{\pi}_{\dot{y}}} + L_{g_{\dot{y}}(x)}\right), L_{s'} + \sum_{\dot{y}\in\dot{Y}} \left|2^m - 2\cdot \min_{\gamma\in\mathbb{F}_2^m\setminus\theta} wt\left(\langle\gamma, g_{\dot{y}}(x)\rangle\right)\right|\right\}.$$

Let us consider the problem of constructing a function s with linearity no higher than some bound of $\mathbf{L}$. Then, we can search for functions s' and $\widehat{\pi}_y$ such that the upper bound obtained by 1 will be less than $\mathbf{L}$. Since every summand in

$$\sum_{\dot{y}\in\dot{Y}} \left(L_{\widehat{\pi}_{\dot{y}}} + L_{g_{\dot{y}}(x)}\right) \text{ and } \sum_{\dot{y}\in\dot{Y}} \left|2^m - 2\cdot wt\left(g_{\dot{y}}\left(x\right)\right)\right|$$

is a non-negative integer (or even positive, since for any permutation π its linearity is greater than 0, $L_\pi > 0$), then for the same value of linearity of function s' the function s obtained by the equation (2) potentially has the greater linearity the more points the function s' has punched points.

Indeed, let $(2m,m)$-function s' be given, which has exactly one punctured point $\dot{y}$. Let $g(x) = s\left(x, \dot{y}\right)$ and let $\widehat{\pi}$ be a permutation on $\mathbb{F}_2^m$. Let us define $(2m,m)$-function s with no punctured points as follows:

$$s(x,y) = \begin{cases} s'(x,y), & y \neq \dot{y} \\ \widehat{\pi}(x), & y = \dot{y} \end{cases}. \qquad (4)$$

For such functions, we can get a potentially smaller estimate on linearity L_s.

Corollary 2 ([82]). *Let s be a $(2m,m)$-function given by the equation* (4). *Then the following upper bound on linearity of L_s of function s is true:*

$$L_s \leqslant \max\left\{L_{s'} + L_{\widehat{\pi}} + L_g, L_{s'} + \left|2^m - 2\cdot \min_{\gamma\in\mathbb{F}_2^m\setminus\theta} wt\left(\langle\gamma, g(x)\rangle\right)\right|\right\}.$$

Due to the consequence 2, the functions s' which have only one punctured point $\dot{y}$ are of most interest.

Let us show that with additional constraints on the function g it is possible to specify an upper bound on the linearity of the function s. For example, when an arbitrary nondegenerate linear combination of coordinate functions $g(x)$ is identically equal to 0 or 1, which is equivalent to the fact that the function $g(x)$ is a constant.

Proposition 4 ([82]). *Let s' be $(2m,m)$-function having exactly one punctured point $\dot{y}$, $g(x) = s'(x,\dot{y})$, $\widehat{\pi}$ is a permutation on $\mathbb{F}_2^m$. Let us define $(2m,m)$-function s by the equation* (4).

Then, if an arbitrary linear combination of function $g(x)$ is either 0 or 1, then the coefficients of Walsh–Hadamard transform of function $s(x,y)$ for arbitrary $\alpha, \beta, \gamma \in \mathbb{F}_2^m$ are evaluated by the following equation:

$$W_s^{\alpha\|\beta,\gamma} = \begin{cases} W_{s'}^{\alpha\|\beta,\gamma} + (-1)^{\langle\beta,\dot{y}\rangle} \cdot W_{\widehat{\pi}}^{\alpha,\gamma}, & \alpha \neq \theta \\ 0, & \alpha = \theta, \gamma \neq \theta \\ W_{s'}^{\theta\|\beta,\theta}, & \alpha = \theta, \gamma = \theta \end{cases} . \tag{5}$$

Let us show that there exist functions s' such that an arbitrary linear combination of coordinate functions $g(x)$ is identically equal to 0 or 1 and the function s' itself has sufficiently high nonlinearity. For example, an arbitrary $(2m,m)$ bent function with a single punctured point has this property.

Proposition 5 ([82]). *Let $b(x,y)\colon \mathbb{F}_2^m \times \mathbb{F}_2^m \to \mathbb{F}_2^m$ be a bent-function having exactly one punctured point $\dot{y}$. Then an arbitrary linear combination of coordinate functions $g(x) = b(x,\dot{y})$ is identically equal to 0 or 1.*

Corollary 3 ([82]). *Under the assertion 4, the upper and lower bounds on the linearity of the L_s function s are given by the following inequalities:*

$$L_{s'} - L_{\widehat{\pi}} \leqslant L_s \leqslant L_{s'} + L_{\widehat{\pi}}.$$

If s' a vectorial bent-function, then

$$L_{s'} < L_s \leqslant L_{s'} + L_{\widehat{\pi}}.$$

In particular, the corollary 3 guarantees a way to construct $(2m,m)$function s by the equation (4) with nonlinearity at least $L_s \leqslant L_{s'} + L_{\widehat{\pi}}$. In the case $m = 4$, the smallest possible value of $L_{\widehat{\pi}}$ is 4. Then using s' as a bent-function we can obtain $(8,4)$-function which has linearity 12.

Let s'_1 and s'_2 be two $(2m,m)$-functions with the punctuated points $\dot{y}_1$ and $\dot{y}_2$, respectively, $\widehat{\pi}_1$ and $\widehat{\pi}_2$ are two permutations over $\mathbb{F}_2^m$. Let's define the functions F_1 and F_2 according to the equation (4) and determine the permutation $F \in S\left(\mathbb{F}_2^{2m}\right)$, $F(x_1, x_2) = (y_1, y_2)$, using the F-construction by the equation (1):

$$y_1 = F_1(x_1,x_2) = \begin{cases} s'_1(x_1,x_2), & x_2 \neq \dot{y}_1 \\ \widehat{\pi}_1(x_1), & x_2 = \dot{y}_1 \end{cases}, \tag{6}$$

$$y_2 = F_2\left(x_2,y_1\right) = \begin{cases} s'_2\left(x_2,y_1\right), & y_1 \neq \dot{y}_2 \\ \widehat{\pi}_2\left(x_2\right), & y_1 = \dot{y}_2 \end{cases}. \tag{7}$$

According to the assumptions used in fixing arbitrary $x_2 \neq \dot{y}_i$, the function $s'_i\left(x_1,x_2\right)$ (for arbitrary $i \in \overline{1,2}$) is bijective on the variable x_1. Then for $x_2 \neq \dot{y}_i$ there are correctly defined bijective mappings $s'^{-1}_i\left(y, x_2\right)$ as functions on one variable y when $x_2 \neq \dot{y}_i$ is fixed. Similarly, the functions $F_i^{-1}(y,x_2)$ are correctly defined as permutations on the variable y when $x_2 \in \mathbb{F}_2^m$ is fixed.

Let's define an expression for the permutation $F^{-1}(y_1, y_2) = (x_1, x_2)$ that is inverse to F: (6)–(7):

$$x_2 = F_2^{-1}\left(y_1, y_2\right) = \begin{cases} s'^{-1}_2(y_2,y_1), & y_1 \neq \dot{y}_2 \\ \widehat{\pi}_2^{-1}(y_2), & y_1 = \dot{y}_2 \end{cases}, \tag{8}$$

$$x_1 = F_1^{-1}\left(y_1,x_2\right) = \begin{cases} s'^{-1}_1\left(y_1,x_2\right), & x_2 \neq \dot{y}_1 \\ \widehat{\pi}_1^{-1}(y_1), & x_2 = \dot{y}_1 \end{cases}. \tag{9}$$

Thus F_2^{-1}, like F_1^{-1}, are functions of the form (4) with one punctured point, and the values of y_2 and x_1 are not directly expressed through x_1, x_2 and y_1, y_2 respectively. For example, y_2 is expressed through x_1 and x_2 from (7) by a rather complicated equation:

$$y_2 = \begin{cases} s'_2\left(x_2,s'_1(x_1, x_2)\right), & x_2 \neq \dot{y}_1, s'_1(x_1,x_2) \neq \dot{y}_2 \\ s'_2\left(x_2,\widehat{\pi}_1(x_1)\right), & x_2 = \dot{y}_1, \widehat{\pi}_1(x_1) \neq \dot{y}_2 \\ \widehat{\pi}_2\left(x_2\right) & x_2 \neq \dot{y}_1, s'_1(x_1,x_2) = \dot{y}_2 \\ \widehat{\pi}_2\left(x_2\right) & x_2 = \dot{y}_1, \widehat{\pi}_1(x_1) = \dot{y}_2 \end{cases}. \tag{10}$$

By setting the constraints on the functions s'_1 and s'_2 we can simplify the expressions for y_2 as a function of x_1, x_2 and x_1 as a function of y_1, y_2 and reduce it to the form (4).

Proposition 6 ([82]). *Let the permutation $F(x_1, x_2) = (y_1, y_2)$ is given by the equations* (6)–(7). *Let also*

1. $\widehat{\pi}_1(x_1) = \dot{y}_2 \Leftrightarrow F_1\left(x_1, x_2\right) = \dot{y}_2$,
2. $x_2 = \dot{y}_1 \Leftrightarrow F_2\left(x_2, y_1\right) = \widehat{\pi}_2\left(\dot{y}_1\right)$.

Then

1. *y_2 is expressed in terms of x_1, x_2 by the following equation:*

$$y_2 = FI_2(x_1, x_2) = \begin{cases} s'_2\left(x_2, s'_1(x_1,x_2)\right), & x_1 \neq \widehat{\pi}_1^{-1}\left(\dot{y}_2\right) \\ \widehat{\pi}_2\left(x_2\right), & x_1 = \widehat{\pi}_1^{-1}\left(\dot{y}_2\right) \end{cases}, \tag{11}$$

2. x_1 is expressed in terms of y_1, y_2 by the following equation:

$$x_1 = FI_1(y_1, y_2) = \begin{cases} s_1'^{-1}\left(y_1, s_2'^{-1}(y_2{,}y_1)\right), & y_2 \neq \dot{y}_1 \\ \widehat{\pi}_1^{-1}\left(y_1\right), & y_2 = \dot{y}_1 \end{cases}. \qquad (12)$$

Thus, if the conditions of the 6 statement are true, the permutation $F(x_1, x_2) = (y_1, y_2)$ is given by the equations (6) and (11), and the reverse permutation $F^{-1}(y_1, y_2) = (x_1, x_2)$ is given by the equations (8) and (12). Each of these equations is given by a equation of the form (4) using functions with one punctured point. Let's list these functions:

- y_1 at $x_2 \neq \dot{y}_1$ is given by the equation $s_1'\left(x_1{,}x_2\right)$;
- y_2 at $x_1 \neq \widehat{\pi}^{-1}\left(\dot{y}_1\right)$ is given by the equation $s_2'\left(x_2{,}s_1'(x_1{,}x_2)\right) = s''(x_2, x_1)$;
- x_1 at $y_2 \neq \dot{y}_1$ is given by the equation $s_1'^{-1}\left(y_1{,}s_2'^{-1}(y_2, y_1)\right) = s_1''(y_1, y_2)$;
- x_2 at $y_1 \neq \dot{y}_2$ is given by the equation $s_2'^{-1}\left(y_2{,}y_1\right)$.

Since the nonlinearity of F equals the nonlinearity of F^{-1}, we can propose an algorithm for constructing a permutation with linearity no higher than **L** based on the statements 4 and 6, described in detail in [82].

If the conditions of the statement 6 are true, it is possible to guarantee the maximal possible algebraic degree of permutations F and F^{-1}. In this case, the permutations F and F^{-1} are expressed by the equations:

$$(y_1, y_2) = F\left(x_1, x_2\right) = \left(F_1\left(x_1, x_2\right), FI_2\left(x_1, x_2\right)\right), \qquad (13)$$

$$(x_1, x_2) = F^{-1}\left(y_1, y_2\right) = \left(FI_1\left(y_1, y_2\right), F_2^{-1}\left(y_1, y_2\right)\right).$$

Definition 2. *A $(2m{,}m)$-function $s'(x{,}y)$ will be called a function with one punctured point $\dot{y}$ if for all $y \neq \dot{y}$ the function $s'(x{,}y)$ is a permutation for the variable x and $s'\left(x{,}\dot{y}\right)$ is not a permutation for the variable x. If $s'\left(x{,}\dot{y}\right) = const$, then such a function will be called C-function with a punctured point $\dot{y}$.*

In the case where the meaning of the gouged point is clear from the context, we will simply talk about the C-function.

Proposition 7 ([85]). *Let the permutation F is defined by the equation* (13). *Then*

- *if* $\deg\left(s_i'\right) \neq 2m - 1$ *and* $(2m{,}m)$*-function* s_i' *is a* C*-function, then*

$$\deg\left(F_i\right) = 2m - 1 \Leftrightarrow \deg\left(\widehat{\pi}_i\right) = m - 1, i \in \overline{1{,}2};$$

– *if* $\deg\left(s_i'^{-1}\right) \neq 2m-1$ *and* $(2m,m)$*-function* $s_i'^{-1}$ *is a* C*-function, then*

$$\deg\left(F_i^{-1}\right) = 2m-1 \Leftrightarrow \deg\left(\widehat{\pi}_1\right) = m-1, i \in \overline{1{,}2};$$

– *if* $\deg\left(s_i'\right) \neq 2m-1$ *and* $(2m,m)$*-function* s_i' *is a* C*-function, then*

$$\deg\left(FI_i\right) = 2m-1 \Leftrightarrow \deg\left(\widehat{\pi}_2\right) = m-1, i \in \overline{1{,}2}.$$

Corollary 4 ([85]). *Let all the conditions of the statement 7 and*

$$\deg\left(F_i\right) = \deg\left(F_i^{-1}\right) = \deg\left(FI_i\right) = 2m-1.$$

Then F *and* F^{-1} *have a maximum possible algebraic degree equal to* $2m-1$.

Consider the connection between the differential δ uniformity of the F permutation and the parameters of the transformations used in constructing it.

Lemma 1 ([85]). *Let the permutation* F *be calculated by the equation* (13)*,* $a_1, a_2, b_1, b_2 \in \mathbb{F}_2^m$*, then* $\delta_F^{a_1\|a_2, b_1\|b_2}$ *is greater than or equal to the number of solutions to the following system of equations*

$$\begin{cases} s_1'\left(x_1, x_2\right) \oplus s_1'(x_1 \oplus a_1, x_2 \oplus a_2) = b_1 \\ s_2''\left(x_1, x_2\right) \oplus s_2''(x_1 \oplus a_1, x_2 \oplus a_2) = b_2 \end{cases} \tag{14}$$

with the following constraints on the values of the variables x_1 *and* x_2*:*

1. $x_2 \neq \dot{y}_1$, $x_2 \neq \dot{y}_1 \oplus a_2$;
2. $x_1 \neq \widehat{\pi}_1^{-1}\left(\dot{y}_2\right)$, $x_1 \neq \widehat{\pi}_1^{-1}\left(\dot{y}_2\right) \oplus a_1$.

Remark 1. *The lemma 1 allows to perform a directed search for pairs of functions* $s_1'\left(x_1, x_2\right)$ *and* $s_2''\left(x_1, x_2\right)$ *such that the differential* δ*-uniformity of the constructed permutation* F *is not higher than a predetermined bound* Δ*. A necessary condition for differential* Δ*-uniformity of the substitution* F *is that the number of solutions of the system* (14) *will be less than or equal to* Δ*.*

Let us show by the example of parametric families of permutations discussed in [83] that the lemma 1 allows us to limit the possible parameter values at which the permutation F will have a differential δ-uniformity not less than a given one.

Let the functions

$$y_1 = F_1\left(x_1, x_2\right) = \begin{cases} \pi_1\left(x_1\right) \cdot x_2, & x_2 \neq \theta \\ \widehat{\pi}_1\left(x_1\right), & x_2 = \theta \end{cases}, \tag{15}$$

$$y_2 = F_2\left(x_2, y_1\right) = \begin{cases} \pi_2\left(x_2 \cdot y_1\right), & y_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), & y_1 = \theta \end{cases}, \tag{16}$$

where the permutations π_i, $\widehat{\pi}_i$, $i \in \overline{1{,}2}$ are parameters of the family of permutations given by the formula (1).

Note that the functions

$$s_1'(x_1, x_2) = \pi_1(x_1) \cdot x_2 \text{ and } s_2'(x_2{,}y_1) = \pi_2\left(x_2 \cdot y_1\right)$$

are C-functions having one punctured point $x_2 = \theta$ and $y_1 = \theta$ respectively. For these functions, the proposition 4 and the corollary 3 are applicable. It is worth noting that s_1' is a bent function belonging to the Maiorana–McFarland class, and the function s_2' belongs to the extended Maiorana–McFarland class of bent functions if π_2 is a linear substitution (see e.g. [19]).

Let's express y_2 as a function of x_1 and x_2 using the proposition 6 of this work. To fulfill the conditions of the proposition above, it is necessary that

$$F_1\left(\widehat{\pi}_1^{-1}(\theta), x_2\right) = \theta, \tag{17}$$

$$F_2(\theta, y_1) = \widehat{\pi}_2(\theta). \tag{18}$$

From equality (17) it follows that $\pi_1(x_1) = \theta \Leftrightarrow \widehat{\pi}\left(x_1\right) = \theta$, and from equality (18) it follows that $\pi_2(\theta) = \widehat{\pi}_2(\theta)$. Then

$$y_2 = \begin{cases} \pi_2\left(\left(x_2\right)^2 \cdot \pi_1\left(x_1\right)\right), & x_1 \neq \widehat{\pi}^{-1}(\theta) \\ \widehat{\pi}_2\left(x_2\right), & x_1 = \widehat{\pi}^{-1}(\theta) \end{cases}. \tag{19}$$

Let $\widehat{\pi}_1^{-1}(\theta) = c_1$, $\widehat{\pi}_2(\theta) = c_2$, the permutation $F(x_1, x_2) = (y_1, y_2)$ is defined by equations (15), (19). Then the affine-equivalent permutation $G = F(x_1 + c_1, x_2) + (\theta, c_2)$ is obviously also defined by the equations (15), (19) (with other parameters). Then, without losing generality, we will further consider only the case where θ is fixed point for π_i, $\widehat{\pi}_i$, $i \in \overline{1{,}2}$.

Definition 3 ([85])**.** *Let $x_1, x_2 \in \mathbb{F}_2^m$, $\pi_i, \widehat{\pi}_i \in S\left(\mathbb{F}_2^m\right)$, $\pi_i(\theta) = \theta$, $\widehat{\pi}_i(\theta) = \theta$, $i \in \overline{1{,}2}$, then the permutation F_A, defined as follows*

$$y_1 = \begin{cases} \pi_1\left(x_1\right) \cdot x_2, & x_2 \neq \theta \\ \widehat{\pi}_1\left(x_1\right), & x_2 = \theta \end{cases},$$

$$y_2 = \begin{cases} \pi_2\left(\left(x_2\right)^2 \cdot \pi_1\left(x_1\right)\right), & x_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), & x_1 = \theta \end{cases}.$$

will be called a permutation from the parametric family of type "A" or just a permutation of type "A".

The following proposition allows us to significantly reduce the parametric family of type "A". It is possible not to consider permutations with linear parameter π_2 since in this case such permutations will be differentially $\delta_{F_A} \geqslant 2^m - 2$-uniform, which will not allow their use in the synthesis of secure symmetric encryption algorithms.

Proposition 8 ([85]). *Let F_A be a permutation from the parametric family of type "A". If π_2 is a linear permutation, then $\delta_{F_A} \geqslant 2^m - 2$.*

In the case when π_2 is a linear function, the function s_1' defined in this section is a bent function and has the highest possible nonlinearity, but, according to the proposition 8, the constructed permutation will have a high differential δ-uniformity.

There remains the question of choosing specific permutations π_i, $\widehat{\pi}_i$, $i \in \overline{1,2}$. In [83] we considered type "A" permutations for the case $m = 4$.

For simplicity, the parameters $\pi_i, i \in \overline{1,2}$ will be fixed by monomial permutations. Such permutations have the form x^d, where GCD $(d, 2^m - 2) = 1$. Given Fermat's little theorem, we are only interested in $d < 2^m - 2$.

In this case, the equations specifying the permutation can be rewritten in the following form:

$$y_1 = \begin{cases} x_1^{\alpha} \cdot x_2, & x_2 \neq \theta \\ \widehat{\pi}_1\left(x_1\right), & x_2 = \theta \end{cases},$$

$$y_2 = \begin{cases} \left(x_2^2 \cdot x_1^{\alpha}\right)^{\beta}, & x_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), & x_1 = \theta \end{cases} = \begin{cases} x_2^{2\beta} \cdot x_1^{\alpha\beta}, & x_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), & x_1 = \theta \end{cases}.$$

In this case, according to the proposition 8 the transformation x^{β} must be a nonlinear transformation. Such permutations have been considered in [83] for the case $m = 4$, in which permutations of type "A" with monomial parameter were investigated experimentally. For the case $m = 4$ there exist 8 values of d such that GCD $\left(d, 2^4 - 2\right) = 1$ are 1, 2, 4, 7, 8, 11, 13, 14. Moreover, if $d \in \{1,2,4,8\}$, then x^d defines a linear permutation. According to the proposition 8 π_2 cannot be linear.

In [83], by fixing α with an arbitrary value from the set $\alpha \in \{1,2,4,7,8,11,13,14\}$ and $\beta \in \{1,2,4,8\}$, by a suitable choice $\widehat{\pi}_i$ we obtain permutations with the following non-linearity indices:

- nonlinearity — 108,
- differential δ-uniformity — 6,
- algebraic degree — 7.

That is, in the most interesting case from a practical point of view $m = 4$, the proposition 8 sets a sufficient condition for constructing permutations with "good" cryptographic properties.

Let the functions

$$F_1(x_1, x_2) = \begin{cases} x_1 \cdot \pi_1(x_2), & \pi_1(x_2) \neq \theta \\ \widehat{\pi}_1(x_1), & \pi_1(x_2) = \theta \end{cases},$$

$$F_2(x_2, y_1) = \begin{cases} x_2 \cdot \pi_2(y_1), & \pi_2(y_2) \neq \theta \\ \widehat{\pi}_2(x_2), & \pi_2(y_2) = \theta \end{cases},$$

where the permutations π_i, $\widehat{\pi}_i$, $i \in \overline{1,2}$ are parameters of the family of permutations given by the expression (1).

Note that the functions $s'(x_1, x_2) = x_1 \cdot \pi_1(x_2)$ and $s'(x_2, y_1) = x_2 \cdot \pi_2(y_1)$ are C-functions as well as the Maiorana-McFarland bent functions, [19].

As before, let us express y_2 as a function of x_1 and x_2 using the proposition 6. And similarly, we will consider only the case where θ is fixed point for π_i, $\widehat{\pi}_i$, $i \in \overline{1,2}$.

Definition 4 ([85]). *Let $x_1, x_2 \in \mathbb{F}_2^m$, $\pi_i, \widehat{\pi}_i \in S(\mathbb{F}_2^m)$, $\pi_i(\theta) = \theta$, $\widehat{\pi}_i(\theta) = \theta$, $i \in \overline{1,2}$, then the permutation F_B defined by equations*

$$y_1 = \begin{cases} x_1 \cdot \pi_1(x_2), & x_2 \neq \theta \\ \widehat{\pi}_1(x_1), & x_2 = \theta \end{cases},$$

$$y_2 = \begin{cases} x_2 \cdot \pi_2(x_1 \cdot \pi_1(x_2)), & x_1 \neq \theta \\ \widehat{\pi}_2(x_2), & x_1 = \theta \end{cases},$$

will be called a permutation from the parametric family of type "B" or just a permutation of type "B".

Let's determine the inverse of the permutation type "B".

$$x_1 = \begin{cases} y_1 \cdot \pi_2(y_2)^{-1}, \pi_2(y_2) \neq \theta \\ \widehat{\pi}_2^{-1}(y_1), \pi_2(y_2) = \theta \end{cases},$$

$$x_2 = \begin{cases} y_2 \cdot \pi_1\left(x_2\right)^{-1}, \pi_1\left(x_2\right) \neq \theta \\ \widehat{\pi}_1^{-1}\left(y_2\right), \pi_1\left(x_2\right) = \theta \end{cases} .$$

Thus an inverse of a permutation of type "B" is itself a permutation of type "B".

Let us use the lemma 1 and describe the value of parameters at which the permutation is known to have high differential δ-uniformity.

Proposition 9 ([85]). *Let $H < S\left(\mathbb{F}_2^m\right)$ be the set of linear permutations. If $\pi_2 \in H$ or $\pi_1 \in x^{-1}H$, then $\delta_{S_B} \geqslant 2^m - 2$.*

Consider the case of monomial permutations: $\pi_1 = x^{\alpha}$, $\pi_2 = x^{\beta}$ where α, β satisfies the equality GCD $\left(\alpha, 2^4 - 2\right) = 1$, GCD $\left(\beta, 2^4 - 2\right) = 1$. Then

$$y_2 = \begin{cases} x_1 \cdot x_2^{\alpha}, x_2 \neq \theta \\ \widehat{\pi}_1\left(x_1\right), x_2 = \theta \end{cases} ,$$

$$y_1 = \begin{cases} x_2 \cdot \left(x_1 \cdot x_2^{\alpha}\right)^{\beta}, x_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), x_1 = \theta \end{cases} = \begin{cases} x_1^{\beta} \cdot x_2^{\alpha\beta+1}, x_1 \neq \theta \\ \widehat{\pi}_2\left(x_2\right), x_1 = \theta \end{cases} .$$

In [83] experimental study of "B" type permutations in the case $m = 4$ was carried out. According to 9 the parameter values α and β belong to the following sets: $\alpha \in \{1,2,4,8\}$, $\beta \in \{7,11,13,14\}$. Let us show that when α is fixed, there exists unique β at which the permutation will not have a high differential δ-uniformity.

Proposition 10 ([85]). *Let $m = 4$ and $\pi_1 = x^{\alpha}$, $\pi_2 = x^{\beta}$ where α, β satisfies the equality GCD $\left(\alpha, 2^4 - 2\right) = 1$, GCD $\left(\beta, 2^4 - 2\right) = 1$. Then if $\alpha\beta + 1 \neq 14 \pmod{15}$, then $\delta_{F_B} \geqslant 2^m - 2$.*

Thus, the following cases are possible, which were experimentally found in the work [83].

1. $\pi_1(x) = x$, $\pi_2(x) = x^{13}$,
2. $\pi_1(x) = x^2$, $\pi_2(x) = x^{14}$,
3. $\pi_1(x) = x^4$, $\pi_2(x) = x^7$,
4. $\pi_1(x) = x^8$, $\pi_2(x) = x^{11}$.

In this case, cases 2 and 4 are the inverse of cases 1 and 3, respectively. For these cases, if $\widehat{\pi}_i$ is chosen correctly, the [83] obtained permutations with the following cryptographic properties:

- nonlinearity — 108,

- differential δ-uniformity — 6,
- algebraic degree — 7.

Consider the family of permutations that generalize the permutations of type "A" and "B" when fixing the monomial parameters of the permutations considered earlier. Let us examine a family of permutations whose parameters are a quadruple of powers $(\alpha, \beta, \gamma, \delta)$ and permutations $\widehat{\pi}_i$, $i \in \overline{1{,}2}$:

$$\begin{aligned} G_1(x_1, x_2) = y_1 &= \begin{cases} x_1^{\alpha} \cdot x_2^{\beta}, & x_2 \neq \theta \\ \widehat{\pi}_1(x_1), & x_2 = \theta \end{cases}, \\ G_2(x_1, x_2) = y_2 &= \begin{cases} x_1^{\gamma} \cdot x_2^{\delta}, & x_1 \neq \theta \\ \widehat{\pi}_2(x_2), & x_1 = \theta \end{cases}. \end{aligned} \tag{20}$$

For the equation (20) to specify a bijective transformation, it is sufficient that the equation

$$\begin{cases} G_1(x_1, x_2) = a_1 \\ G_2(x_1, x_2) = a_2 \end{cases}$$

had a solution for arbitrary $a_1, a_2 \in \mathbb{F}_2^m$.

Consider the case $m = 4$. By Fermat's little theorem there are a total of 8 monomial permutations of the field $\mathbb{F}_{2^4}$ that are not equal to each other. Using the lemma 1 we can constrain the parameter values $(\alpha, \beta, \gamma, \delta)$ in the equation (**??**) using a computer, similar to [83]. As in [83], if parameters $\widehat{\pi}_i$, $i \in \overline{1{,}2}$ are chosen correctly, we obtain permutations that have the following cryptographic properties:

- nonlinearity — 108,
- differential δ-uniformity — 6,
- algebraic degree — 7.

It has been experimentally verified that the above nonlinearity indices are achieved, for example, when $\widehat{\pi}_i(x) = x^d$, $d \in \{7{,}11{,}13{,}14\}$.

In [91] it is shown that the generalized construction is also an F-construction. For the case $m = 4$, the classification of parameter values $(\alpha, \beta, \gamma, \delta)$ using the lemma 1 is performed and it is shown that permutations with the above cryptographic properties can only be constructed using the parameters given in [85].

It is easy to show that for a large number of parameter values of the considered parametric families of permutations the value of graph algebraic immunity equals to 2, which can potentially lead to the use of algebraic methods of analysis of symmetric encryption algorithms, [23]. Consider $(2m{,}m)$-functions s_1', s_2', s_1'', s_2'' of the form:

1. $y = x_1 \cdot x_2 \oplus \text{Ind}_\theta(x_2)$;
2. $y = x_1 \cdot x_2^{-1} \oplus \text{Ind}_\theta(x_2)$;
3. $y = x_1^{-2} \cdot x_2 \oplus \text{Ind}_\theta(x_2)$.

For each of these functions we can give a function $g(x_1, x_2, y)$ whose nonzero linear combination of coordinate functions equals 0, which has an algebraic degree at most two, indicating that the value of the graph algebraic immunity of permutations with such coordinate functions will equal 2. Indeed, for the first function $g(x_1, x_2,x_3) = x_2 \cdot y \oplus x_1 \cdot x_2^2 = 0$, for the second — $g(x_1, x_2,x_3) = x_2 \cdot y \oplus x_1 = 0$, for the third — $g(x_1, x_2,x_3) = y \cdot x_2 \oplus y \cdot x_1 = \text{Ind}_\theta(x_2)$. In the case of monomial choice of substitutions π_i, $i = 1,2$ for a parametric family of type "B" all substitutions will have the value $AI_{gr}(F) = 2$, as for the parametric family of type "A" with monomial choice π_i, $i = 1,2$ with linear substitution π_1, and in the case of nonlinear choice when $\widehat{\pi}_2 = x^{-1}$.

Thus, three parametric families of nonlinear bijective transformations are proposed; for the most interesting from the practical point of view case $m = 4$ some parameters are fixed, allowing to produce permutations possessing "good" cryptographic properties at a suitable choice of $\widehat{\pi}_i$, $i = 1,2$. If parameters $\widehat{\pi}_i$, $i \in \overline{1,2}$ are chosen correctly, we obtain permutations having the following cryptographic properties:

- nonlinearity — 108,
- differential δ-uniformity — 6,
- algebraic degree — 7;
- graph algebraic immunity — 3.

Consider the method of construction $\widehat{\pi}_i$, $i = 1,2$ using a heuristic algorithm based on the well-known genetic algorithm [32]. This approach was previously used successfully in the implementation of the spectral-linear and spectral-differential methods of permutation construction [60]. A detailed description of the algorithm, its correctness, and optimization methods are described in detail in [90]. Briefly, the essence of the proposed algorithm consists in successive multiplication of the permutations $\widehat{\pi}_i$, $i = 1,2$ by transpositions and selection among the obtained permutations of the $\mathbb{F}_2^8$ space of the best on nonlinearity, differential uniformity and corresponding values in linear and difference spectra. Thus, the current generation of permutations generates some number of new pairs, of which only a small number of the "best" survive. No crossing is performed, only "random mutations" within $\widehat{\pi}_i$, $i = 1,2$.

Consider the construction of the nonlinear bijective transformation of the Kuznyechik algorithm, whose representation was taken as the basis for the parametric families of permutations studied in this dissertation.

Proposition 11 ([86]). *For the* π *permutation of the Kuznyechik algorithm, there are the following* $(\mathsf{A}_i, \mathsf{B}_i)$, $i = 1{,}2$, *subgroups of* $\mathbb{F}_2^8$

- $\mathsf{A}_1 = \left\{\alpha^{-1}\left(\mathtt{0xd} \cdot x \| x\right) \middle| \, x \in \mathbb{F}_{2^4}\right\}$, $\mathsf{B}_1 = \left\{\beta\left(\mathtt{0x0} \| y\right) \middle| \, y \in \mathbb{F}_{2^4}\right\}$,
- $\mathsf{A}_2 = \left\{\alpha^{-1}\left(x \| \mathtt{0x0}\right) \middle| \, x \in \mathbb{F}_{2^4}\right\}$, $\mathsf{B}_2 = \left\{\beta\left(y \| \mathtt{0x0}\right) \middle| \, y \in \mathbb{F}_{2^4}\right\}$,

such that there exist $a, b \in \mathbb{F}_2^8$: $\pi(\mathsf{A}_i \oplus a) = \mathsf{B}_i \oplus b$.

Definition 5 ([86]). *A pair of subgroups of the space* $\mathbb{F}_q$ (A, B) *will be called I-pair for the permutation* $\pi\colon \mathbb{F}_q \to \mathbb{F}_q$ *if there exist* $a{,}b \in \mathbb{F}_q$ *such that*

$$\pi(\mathsf{A} \oplus a) = \mathsf{B} \oplus b.$$

We will call A *and* B *the LI and RI sets for* π, *respectively.*

The path $\operatorname{span}(S)$ is a linear span of the set S. Then, using the ideas proposed in [52], we can propose the following algorithm for finding I-pairs for permutation $\pi\colon \mathbb{F}_q \to \mathbb{F}_q$:

```
1. $i := 0$
2. **for every** $a, b \in \mathbb{F}_q$:
    a. $\mathsf{A}_i \leftarrow \{0\}$;
    b. $\mathsf{B}_i \leftarrow \operatorname{span}\left(\pi\left(\mathsf{A}_i \oplus a\right) \oplus b\right)$;
    c. $\mathsf{A}_i \leftarrow \operatorname{span}\left(\pi^{-1}\left(\mathsf{A}_i \oplus b\right) \oplus a\right)$;
    d. **if** $\mathsf{A}_i = \operatorname{span}(\mathsf{A}_i)$ **then:**
        – **if** $|\mathsf{A}_i| \neq 2^8$, **print**$(A_i = \mathsf{A}_i \oplus a, B_i = \mathsf{B}_i \oplus b)$, $i \leftarrow i + 1$;
        – **for every** $x \in \mathbb{F}_2^8 \backslash \mathsf{A}_i$: $\mathsf{A}_i \leftarrow \operatorname{span}\left(\mathsf{A}_i \cup x\right)$, **go to step** (2.b);
```

In proposition 11 we found two I pairs of sets $(\mathsf{A}_i, \mathsf{B}_i)$ for permutation π; every set consists of 16 elements. Using algorithm 1 one can find such pairs of sets of any size. We implemented it and founded:

- 2 I pairs $(\mathsf{A}_i, \mathsf{B}_i)$, $|\mathsf{A}_i| = |\mathsf{B}_i| = 16$;
- 1 943 I pairs $(\mathsf{A}_i, \mathsf{B}_i)$, $|\mathsf{A}_i| = |\mathsf{B}_i| = 4$;
- 2 730 I pairs $(\mathsf{A}_i, \mathsf{B}_i)$, $|\mathsf{A}_i| = |\mathsf{B}_i| = 2$.

For permutations of the proposed parametric families, there are also I-pairs of power 2^m each kind:

1. $\mathsf{A}'_1 = \{(\theta \| x) | \, x \in \mathbb{F}_{2^m}\}$, $\mathsf{B}'_1 = \{(\theta \| y) | \, y \in \mathbb{F}_{2^m}\}$,
2. $\mathsf{A}'_2 = \{(x \| \theta) | \, x \in \mathbb{F}_{2^m}\}$, $\mathsf{B}'_2 = \{(y \| \theta) | \, y \in \mathbb{F}_{2^m}\}$,

Thus, it is necessary to be able to provide a reasonable estimate of the security of symmetric encryption algorithms with respect to invariant attacks.

The fourth chapter of the thesis is devoted to the influence of the design features of the proposed parametric families of perrmutations on the security of symmetric encryption algorithms. We will use the notations from the first section.

Definition 6 ([92]). *We will call matrices of type II as matrices $C \in (\mathbb{F}_2)_{n'm,n'm}$ of the form*

$$C = \begin{pmatrix} C_{1,1} & \dots & C_{1,m} \\ \vdots & \ddots & \vdots \\ C_{m,1} & \dots & C_{m,m} \end{pmatrix},$$

where $C_{i,j} \in (\mathbb{F}_2)_{n',n'}$ are nonsingular, $i,j = 1, \dots, m$.

Consider a symmetric encryption algorithm based on an XSL network structure with a linear transformation given by a matrix of type II. Let us describe one approach to find sets $G_K \subset \mathbb{F}_2^n$, invariant with respect to the composition of transformations $\mathrm{X}[K] \circ \mathrm{L} \circ \mathrm{S}$. Let there be a pair of families of sets $(\mathcal{A}, \mathcal{B})$, where

$$\mathcal{A} = \{A_1, A_2, \dots, A_{e_a}\}, \, A_i \subseteq \mathbb{F}_2^{n'},$$

$$\mathcal{B} = \{B_1, B_2, \dots, B_{e_b}\}, \, B_i \subseteq \mathbb{F}_2^{n'},$$

and for every $i \in \{1, \dots, e_a\}$ there exists $j \in \{1, \dots, e_b\}$ such that $A_i^{\pi} \subseteq B_j$. Consider the families $\mathcal{A}^m$ and $\mathcal{B}^m$ — the direct product of the sets $\mathcal{A}$ and $\mathcal{B}$ respectively. Then for any element $A_{i_1} \times \ldots \times A_{i_m} \in \mathcal{A}^m$ there exists an element $B_{j_1} \times \ldots \times B_{j_m} \in \mathcal{B}^m$ such that

$$(A_{i_1} \times \ldots \times A_{i_m})^{\mathrm{S}} = \left(A_{i_1}^{\pi} \times \ldots \times A_{i_m}^{\pi}\right) \subseteq B_{j_1} \times \ldots \times B_{j_m}.$$

The set G_K will be searched among subsets of the set $\mathcal{A}^m$, that is, elements of the set G_K are sets of the form $A_{i_1} \times A_{i_2} \times \ldots \times A_{i_m} \in \mathcal{A}^m$.

Let $\mathcal{C}$ be a family of sets such that for any element $B_{j_1} \times \ldots \times B_{j_m} \in \mathcal{B}^m$ there exists an element C of the family $\mathcal{C}$ for which the inclusion

$$(B_{j_1} \times \ldots \times B_{j_m})^{\mathrm{L}} \subseteq C.$$

Let there also be $K \in \left(\mathbb{F}_2^{n'}\right)^m$ such that $\mathcal{C}^{\mathrm{X}[K]} = \mathcal{A}^m$, that is, the following diagram is true:

$$\mathcal{A}^m \xrightarrow{\mathrm{S}} \mathcal{B}^m \xrightarrow{\mathrm{L}} \mathcal{C} \xrightarrow{\mathrm{X}[K]} \mathcal{A}^m. \tag{21}$$

We will do all further reasoning under the assumption that the diagram is feasible 21. In this case, the equality is obviously satisfied: $|\mathcal{C}| = |\mathcal{A}^m|$. Indeed, consider the element $A_{i_1} \times A_{i_2} \times \ldots \times A_{i_m} \in \mathcal{A}^m$, $i_1, \ldots, i_m \in \{1,\ldots,e_a\}$. Let $K = (k_1, k_2, \ldots, k_m)$. Then

$$(A_{i_1} \oplus k_1) \times (A_{i_2} \oplus k_2) \times \ldots \times (A_{i_m} \oplus k_m) \in \mathcal{C}.$$

Thus, set C consists of a direct product of sets of the form $A_j \oplus k_i$, $j \in \{1,\ldots,e_a\}$, $i \in \{1,\ldots,m\}$.

Proposition 12 ([92]). *Let there be a symmetric encryption algorithm based on an XSL-network whose linear transformation $L = (l_{a,b})_{m\times m}$, $l_{a,b} \in GL_{n'}(2)$, $a,b = 1,\ldots,m$, is given by matrix of type II. Consider the sets*

$$B = B_{i_1} \times B_{i_2} \times \ldots \times B_{i_m} \in \mathcal{B}^m, i_1, \ldots, i_m \in \{1,\ldots,e_b\}$$

and

$$C = C_{j_1} \times C_{j_2} \times \ldots \times C_{j_m} \in \mathcal{C}, j_1, \ldots, j_m \in \{1,\ldots,e_a\},$$

such that $B^{\mathrm{L}} \subseteq C$, and for some key $K \in \left(\mathbb{F}_2^{n'}\right)^m$ the diagram 21 is fulfilled. Then for any $j \in \{j_1, \ldots, j_m\}$ the inequality $|C_j| \geqslant \max\limits_{i\in\{i_1,\ldots,i_m\}} |B_i|$.

It follows from the trueness of the diagram 21 that for any $i_1, \ldots, i_m \in \{1, \ldots, e_a\}$ there exist such $j_1, \ldots, j_m \in \{1, \ldots, e_a\}$ that the diagram will be true:

$$A_{i_1} \times \ldots \times A_{i_m} \xrightarrow{\mathrm{X}[K]\circ\mathrm{L}\circ\mathrm{S}} A_{j_1} \times \ldots \times A_{j_m}.$$

Let us define on the family $\mathcal{A}^m$ an oriented graph Γ with marked arcs as follows. The vertices of this graph are elements of the family $\mathcal{A}^m$, and the vertices $X,Y \in \mathcal{A}^m$, are connected by an arc marked K if and only if there exists a key K such that $X^{\mathrm{X}[K]\circ\mathrm{L}\circ\mathrm{S}} \to Y$. In this case, if $\mathbb{F}_2^{n'} \in \mathcal{A}$, then it is obvious that for an arbitrary key K

$$\left(\left(\mathbb{F}_2^{n'}\right)^m\right)^{\mathrm{X}[K]\circ\mathrm{L}\circ\mathrm{S}} = \left(\mathbb{F}_2^{n'}\right)^m$$

there is a cycle, which we will call trivial. To construct the set G_K it is necessary to be able to find non-trivial cycles in the graph Γ. In particular G_K is the set of vertices of graph Γ lying on cycles of length 1 (loops) labeled K. However, further we propose to consider the more general case when a loop consists of more than one vertex. Suppose that in a graph Γ there exists a nontrivial cycle of length r given by a subfamily of the family $\mathcal{A}^m$. This is equivalent to the fact that some subset $\mathcal{A}^m$ is invariant with respect to r rounds of the algorithm under consideration for some keys $K_1, \ldots, K_r$. Let us find the necessary conditions for the existence of a nontrivial loop and propose a constructive algorithm for finding it.

Let $\mathcal{A}' \subset \mathcal{A}^m$ — the set of vertices of graph Γ defining some nontrivial cycle of length r. Denote $\mathcal{B}' = (\mathcal{A}')^{\mathrm{S}}$, $\mathcal{C}' = (\mathcal{B}')^{\mathrm{L}}$. Thus, for each $A \in \mathcal{A}'$ there exists a key K and a set $C \in \mathcal{C}'$ such that $C^{\mathrm{X}[K]} = A$.

Proposition 13 ([92]). *Suppose there is a symmetric encrytpion algorithm based on an XSL-network whose linear transformation $L = (l_{a,b})_{m\times m}$, $l_{a,b} \in GL_{n'}(2)$, $a,b = 1,\ldots,m$, is given by a type II matrix, elements of the family $\mathcal{A}'$ define some nontrivial loop of the graph Γ, $A_{a_1} \times \ldots \times A_{a_m} \in \mathcal{A}'$, и*

$$B_{b_1} \times \ldots \times B_{b_m} \in \mathcal{B}', B_{b_1} \times \ldots \times B_{b_m} = \mathrm{S}\left(A_{a_1} \times \ldots \times A_{a_m}\right),$$

$$C_{c_1} \times \ldots \times C_{c_m} \in \mathcal{C}', C_{c_1} \times \ldots \times C_{c_m} = \mathrm{L}\left(B_{b_1} \times \ldots \times B_{b_m}\right).$$

Then

1. $|A_{a_1}| = |B_{b_1}| = |C_{c_1}|$,
2. $|A_{a_1}| = |A_{a_2}| = \ldots = |A_{a_m}|$,
3. $|B_{b_1}| = |B_{b_2}| = \ldots = |B_{b_m}|$,
4. $|C_{c_1}| = |C_{c_2}| = \ldots = |C_{c_m}|$.

The following proposition allows us to specify an algorithm for finding cycles in a Γ graph or prove that there are no nontrivial cycles.

Proposition 14 ([92]). *Let for a symmetric encryption algorithm based on XSL-network whose linear transformation $L = (l_{a,b})_{m\times m}$, $l_{a,b} \in GL_{n'}(2)$, $a,b = 1,\ldots,m$, is given by matrix of type II, elements of $\mathcal{A}'$ family define some non-trivial loop of graph Γ. Let also*

$$B = B_{i_1} \times B_{i_2} \times \ldots \times B_{i_m} \in \mathcal{B}',$$

$$C = C_{j_1} \times C_{j_2} \times \ldots \times C_{j_m} \in \mathcal{C}',$$

where $C = B^{\mathrm{L}}$. Then

1. *for arbitrary $v \in \{1,\ldots,m\}$ the set B_{i_v} is a coset on some subgroup of $\mathbb{F}_2^{n'}$;*
2. *for arbitrary $v \in \{1,\ldots,m\}$ the set C_{j_v} is a coset on some subgroup of $\mathbb{F}_2^{n'}$;*

Corollary 5 ([92]). *Let in the conditions of the previous proposition*

$$A = A'_{i_1} \times A'_{i_2} \times \ldots \times A'_{i_m} \in \mathcal{A}'.$$

Then for any $j \in \{1,\ldots,m\}$ the set A'_{i_j} a coset on some subgroup of $\mathbb{F}_2^{n'}$.

Thus, we are first of all interested in such pairs of sets (A,B) that $A^{\pi} = B$, $A = H_A \oplus h_A$, $B = H_B \oplus h_B$, and H_A, H_B — subspaces of $\mathbb{F}_2^{n'}$, $h_A, h_B \in \mathbb{F}_2^{n'}$.

Suppose there are M pairs of such sets (A_i, B_i), $i \in \{1,\ldots,M\}$, with $A_i = H_{A,i} \oplus h_{A,i}$, $B_i = H_{B,i} \oplus h_{B,i}$, $|A_i| = |A_j|\ \forall i,j \in \{1,\ldots,M\}$. The necessity of the same size of the sets A_i is due to a similar requirement for the sets that form a cycle in the graph Γ. Consider the vector $h \in \left(\mathbb{F}_2^{n'}\right)^m$:

$$h = \left(h_{B,i_1}, h_{B,i_2}, \ldots, h_{B,i_m}\right), i_1, \ldots, i_m \in \{1,\ldots,M\}.$$

The total of such vectors is $|M|^m$.

For each $v,w = 1,\ldots,m$, calculate the set $C(h,v,w) \subset \mathbb{F}_2^{n'}$:

$$C(h,v,w) = \left\{ \sum_{b=1}^{m} h_{B,i_b} \cdot l_{b,w} + y \cdot l_{v,w} \middle| y \in H_{B,v} \right\}$$

and check if there exists such $g(w) \in \mathbb{F}_2^{n'}$ and such $j(w) \in \{1,\ldots,M\}$, depending on w, that

$$C(h,v,w) \oplus g(w) = A_{j(w)}.$$

That is, for different v but the same w the set $A_{j(w)}$ and the element $g(w)$ must be the same.

Theorem 1 ([92]). *Let for a symmetric encryption algorithm based on XSL-network whose linear transformation $L = (l_{a,b})_{m\times m}$, $l_{a,b} \in GL_{n'}(2)$, $a,b = 1,\ldots,m$, is given by a matrix of type II, the elements of the family $\mathcal{A}'$ specify some non-trivial loop in the graph Γ. Let also $A_i = H_{A,i} \oplus h_{A,i}$, $H_{A,i}$ — a subspace of $\mathbb{F}_2^{n'}$, $h_{A,i} \in \mathbb{F}_2^{n'}$, $i \in \{1,\ldots,M\}$, with $|A_i| = |A_j|\ \forall i,j \in \{1,\ldots,M\}$. For the set $A \in \mathcal{A}'$,*

$$A = A_{i_1}\{times A_{i_2} \times \ldots \times A_{i_m}, i_1, \ldots, i_m \in \{1,\ldots,M\}$$

there exists a key K such that $A^{\mathrm{L}\circ\mathrm{S}\circ\mathrm{X}[K]} = A' \in \mathcal{A}'$,

$$A' = A_{j_1} \times A_{j_2} \times \ldots \times A_{j_m}, j_1, \ldots, j_m \in \{1,\ldots,M\}$$

then and only if for every $w \in \{1,\ldots,m\}$ there exists a vector $g(w) \in \mathbb{F}_2^{n'}$ and number $j(w) \in \{1,\ldots,M\}$ such that for any $v \in \{1,\ldots,m\}$ the equality is satisfied

$$C(h,v,w) \oplus g(w) = A_{j(w)},$$

where

$$C(h,v,w) = \left\{ \sum_{b=1}^{m} h_{B,i_b} \cdot l_{b,w} \oplus y \cdot l_{v,w} \,\middle|\, y \in H_{B,v} \right\}.$$

Using the proved theorem, it is possible constructively to enumerate invariants for the round transformation of an algorithm based on the XSL-network of the considered kind.

Using the results of the theorem 1, it is possible to propose the following approach to prove the impossibility of applying the proposed attack for the Kuznyechik algorithm. Let $(\mathsf{A}_i, \mathsf{B}_i)$ I-pair for substitution π. Consider

$$B_i^{(j)} = \underbrace{\{\theta\} \times \ldots \times \{\theta\}}_{j-1} \times \mathsf{B}_i \times \{\theta\} \times \ldots \times \{\theta\},$$

$$\mathrm{L}\left(B_i^{(j)}\right) = C_i^{(j)} = \left\{ \left(c_{i,k}^{(j,1)}, \ldots, c_{i,k}^{(j,m)}\right), k = 1, \ldots, |\mathsf{B}_i| \right\}.$$

Then according to the theorem 1 each set

$$C_i^{(j,l)} = \left\{ c_{i,k}^{(j,l)}, k = 1, \ldots, |\mathsf{B}_i| \right\}$$

is some LI set A_d for π. Then

$$\exists\, c_1, c_2 \in \mathbb{F}_{2^4} \colon \pi\left(\mathsf{A}_d \oplus c_1\right) \oplus c_2$$

is a subgroup of $(\mathbb{F}_q, \oplus)$. The following is true

Proposition 15 ([86]). *Let* π *be a permutation,* L *be a linear and* S *be a nonlinear transformation of the Kuznyechik algorithm. Then for every I pair* $(\mathsf{A}_i, \mathsf{B}_i)$*,* $|\mathsf{B}_i| > 1$*, for permutation* π *and for every* $j = \{1, \ldots, m\}$*, there is* $l = \{1, \ldots, m\}$ *so that* $C_i^{(j,l)}$ *is not a subset of any subgroup* A_d *so that*

$$\exists\, c_1, c_2 \in \mathbb{F}_{2^4} \colon \pi\left(\mathsf{A}_d \oplus c_1\right) \oplus c_2$$

is a subgroup of $(\mathbb{F}_q, \oplus)$.

This dissertation was carried out at the Department of Computer Security at the Tikhonov Moscow Institute of Electronics and Mathematics of HSE, Moscow, Russia.

List of papers on the topic of the dissertation work presented for the defense (with the personal contribution of the candidate). The main aspects of the thesis are presented in 13 scientific papers. 9 of them are included in the list of eligible journals refers to a list of journals featuring publications that may be considered for the provision of Academic Merit Bonuses, research productivity assessment and other procedures.:

1. *Fomin, D. B.* A timing attack on CUDA implementations of an AES-type block cipher / D. B. Fomin // Mat. Vopr. Kriptogr. 2016. Vol. 7, no. 2. P. 121—130
2. *Fomin, D. B.* New Classes of 8-bit Permutations Based on a Butterfly Structure / D. B. Fomin // Mat. Vopr. Kriptogr. 2019. Vol. 10, no. 2. P. 169—180
3. *Fomin, D. B.* Construction of permutations on the space V_{2m} by means of $(2m,m)$-functions / D. B. Fomin // Mat. Vopr. Kriptogr. 2020. Vol. 11, no. 3. P. 121—138. (In Russian)
4. *Fomin, D. B.* On the algebraic degree and differential uniformity of permutations on the space V_{2m} constructed via $(2m, m)$-functions / D. B. Fomin // Mat. Vopr. Kriptogr. 2020. Vol. 11, no. 4. P. 133—149. (In Russian)
5. *Trifonov, D. I.* Invariant subspaces in SPN block cipher / D. I. Trifonov, D. B. Fomin // Prikl. Diskr. Mat. 2021. Vol. 54. P. 58—76. (In Russian)
   /D.B. Fomin's owns the results following the proof of statement 4, pp. 67–71./
6. A compact bit-sliced representation of Kuznyechik S-box / O. D. Avraamova, D. B. Fomin, V. A. Serov, [et al.] // Mat. Vopr. Kriptogr. 2021. Vol. 12, no. 2. P. 21—38
   /D.B. Fomin's own formulation of the problem, as well as the results of section 3.3./
7. *Kovrizhnykh, M. A.* On differential uniformity of permutations derived using a generalized construction / M. A. Kovrizhnykh, D. B. Fomin // Mat. Vopr. Kriptogr. 2022. Vol. 13, no. 2. P. 37—52
   /D.B. Fomin's own formulation of the problem and research methodology./
8. *Fomin, D. B.* On the impossibility of an invariant attack on Kuznyechik / D. B. Fomin // Journal of Computer Virology and Hacking Techniques. 2022. Vol. 18, no. 1. P. 61—67
9. *Kovrizhnykh, M. A.* Heuristic algorithm for obtaining permutations with given cryptographic properties using a generalized construction / M. A. Kovrizhnykh, D. B. Fomin // Prikl. Diskr. Mat. 2022. Vol. 57. P. 5—21. (In Russian)
   /D.B. Fomin's own formulation of the problem, research methodology, and the results of section 3.1./.

At the same time, 5, 8 and 9 articles from this list are indexed in the international database Scopus.

The author's publications on the topic of the dissertation in other journals:

10. *Fomin, D. B.* O podhodah k postroeniyu nizkoresursnyh nelinejnyh preobrazovanij / D. B. Fomin // Obozrenie prikladnoj i promyshlennoj matematiki. 2018. Vol. 25, no. 4. P. 379—381. (In Russian)
11. *Fomin, D. B.* Hardware implementation of one class of 8-bit permutations / D. B. Fomin, D. I. Trifonov // Prikl. Diskr. Mat. Suppl. 2019. Vol. 12. P. 134—137. (In Russian)

    /Fomin D.B. owns a method for selecting nonlinear bijective transformations of a special kind./
12. *Fomin, D. B.* On the way of constructing differentially 2δ-uniform permutations over $\mathbb{F}_{2^{2m}}$ / D. B. Fomin // Prikl. Diskr. Mat. Suppl. 2021. Vol. 14. P. 51—55. (In Russian)
13. *Kovrizhnykh, M. A.* On a heuristic approach to constructing bijective vector Boolean functions with given cryptographic properties / M. A. Kovrizhnykh, D. B. Fomin // Prikl. Diskr. Mat. Suppl. 2021. Vol. 14. P. 181—184. (In Russian)

    /D.B. Fomin’s own formulation of the problem and research methodology./

Conclusion

The main results of this work are as follows.

1. A new timing attack on block cipher algorithms, implemented on GPU, is developed. The fundamental possibility of applying timing attacks on symmetric algorithms implemented on GPU is shown. The practical applicability of the proposed method of analysis for the AES algorithm is also shown.
2. Parametric families of permutations are proposed, estimated for them such cryptographic properties as nonlinearity, algebraic degree, differential δ-uniformity. Algorithms for constructing nonlinear bijective transformations from the considered parametric families are developed.
3. Proposed a new invariant attack for block ciphers based on XSL-network.
4. The inefficiency of applying the proposed attack for the Kuznyechik algorithm is shown.

References

1. GOST 34.11-2018. Information technology. Cryptographic data security. Hash-function. — M. : Standartinform, 2018. — 25 p. — (International Standard). — (In Russian).
2. GOST 34.12-2018. Information technology. Cryptographic data security. Block ciphers. — M. : Standartinform, 2018. — 14 p. — (International Standard). — (In Russian).
3. Information Security Doctrine of the Russian Federation. — Rossiyskaya Gazeta, 2016. — URL: https://rg.ru/documents/2016/12/06/doktrina-infobezobasnost-site-dok.html ; (In Russian).
4. Methodological document. Methodology for assessing threats to information security. — M. : FSTEC of Russia, 2021. — 83 p. — (In Russian).
5. R 1323565.1.012-2017. Information technology. Cryptographic data security. Principles of creation and modernization for cryptographic modules. — M. : Standartinform, 2018. — 23 p. — (Recommendations for standardization). — (In Russian).
6. *Aziz*, *A.* A look-up-table implementation of AES / A. Aziz, N. Ikram //. — 01/2007. — P. 187—191.
7. *Bao*, *Z.* Bitsliced Implementations of the PRINCE, LED and RECTANGLE Block Ciphers on AVR 8-bit Microcontrollers. / Z. Bao, P. Luo, D. Lin // IACR Cryptology ePrint Archive. — 2015. — Vol. 2015. — P. 1118. — URL: http://dblp.uni-trier.de/db/journals/iacr/iacr2015.html#BaoLL15.
8. *Bernstein*, *D. J.* Cache-timing attacks on AES : tech. rep. / D. J. Bernstein. — 2005.
9. *Biham*, *E.* A Fast New DES Implementation in Software. / E. Biham // FSE. Vol. 1267 / ed. by E. Biham. — Springer, 1997. — P. 260—272. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse97.html#Biham97a.
10. *Biham*, *E.* Differential Cryptanalysis of DES-like Cryptosystems. / E. Biham, A. Shamir // J. Cryptology. — 1991. — Vol. 4, no. 1. — P. 3—72. — URL: http://dblp.uni-trier.de/db/journals/joc/joc4.html#BihamS91.

11. *Biryukov*, *A.* Reverse-Engineering the S-Box of Streebog, Kuznyechik and STRIBOBr1. / A. Biryukov, L. Perrin, A. Udovenko. — 2016. — http://eprint.iacr.org/2016/071.

12. *Biryukov*, *A.* Reverse-Engineering the S-Box of Streebog, Kuznyechik and STRIBOBr1. / A. Biryukov, L. Perrin, A. Udovenko. — 2016. — URL: http://dblp.uni-trier.de/db/conf/eurocrypt/eurocrypt2016-1.html#BiryukovPU16.

13. Differential Cache-Collision Timing Attacks on AES with Applications to Embedded CPUs. / A. Bogdanov, T. Eisenbarth, C. Paar, [et al.] // CT-RSA. Vol. 5985 / ed. by J. Pieprzyk. — Springer, 2010. — P. 235—251. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/ctrsa/ctrsa2010.html#BogdanovEPW10.

14. Strong 8-bit Sboxes with efficient masking in hardware extended version. / E. Boss, V. Grosso, T. Güneysu, [et al.] // J. Cryptographic Engineering. — 2017. — Vol. 7, no. 2. — P. 149—165. — URL: http://dblp.uni-trier.de/db/journals/jce/jce7.html#BossGGL0017.

15. An APN permutation in dimension six / K. Browning, J. Dillon, M. McQuistan, [et al.]. — 2010.

16. *Buchfuhrer*, *D.* The complexity of Boolean formula minimization. / D. Buchfuhrer, C. Umans // J. Comput. Syst. Sci. — 2011. — Vol. 77, no. 1. — P. 142—153. — URL: http://dblp.uni-trier.de/db/journals/jcss/jcss77.html#BuchfuhrerU11.

17. *Burov*, *D. A.* On the existence of special nonlinear invariants for round functions of XSL-ciphers / D. A. Burov // Diskr. Mat. — 2021. — Vol. 33, no. 2. — P. 31—45. — (In Russian).

18. *Canteaut*, *A.* Construction of Lightweight S-Boxes using Feistel and MISTY structures (Full Version). / A. Canteaut, S. Duval, G. Leurent // IACR Cryptology ePrint Archive. — 2015. — Vol. 2015. — P. 711. — URL: http://dblp.uni-trier.de/db/journals/iacr/iacr2015.html#CanteautDL15 ; http://eprint.iacr.org/2015/711.

19. *Carlet*, *C.* Vectorial Boolean functions for cryptography / C. Carlet // Boolean Models and Methods in Mathematics, Computer Science, and Engineering. — 2006.

20. Building Your Private Cloud Storage on Public Cloud Service Using Embedded GPUs / W. Cheng, F. Zheng, W. Pan, [et al.] // Security and Privacy in Communication Networks. — Cham : Springer International Publishing, 2018. — P. 512—528.

21. *Chichaeva*, *A. A.* Search for effectively implemented permutations with optimal cryptographic characteristics / A. A. Chichaeva // Ruscrypto 2021. — 2021. — (In Russian).

22. *Christophe Clavier*, *L. R.* Systematic and Random Searches for Compact 4-Bit and 8-Bit Cryptographic S-Boxes / L. R. Christophe Clavier // IACR Cryptology ePrint Archive. — 2019.

23. *Courtois*, *N.* Cryptanalysis of Block Ciphers with Overdefined Systems of Equations / N. Courtois, J. Pieprzyk. — 2002. — https://eprint.iacr.org/2002/044. Cryptology ePrint Archive, Report 2002/044.

24. *Cruz Jiménez*, *R. A. de la*. Generation of 8-bit S-Boxes having almost optimal cryptographic properties using smaller 4-bit S-Boxes and finite field multiplication / R. A. de la Cruz Jiménez. — www.cs.haifa.ac.il/~orrd/LC17/paper60.pdf.

25. *Cruz Jiménez*, *R. A. de la*. On some methods for constructing almost optimal S-Boxes and their resilience against side-channel attacks / R. A. de la Cruz Jiménez. — 2018. — URL: https://eprint.iacr.org/2018/618 ; https://eprint.iacr.org/2018/618. Cryptology ePrint Archive, Paper 2018/618.

26. *Daemen*, *J.* Rijndael for AES. / J. Daemen, V. Rijmen // AES Candidate Conference. — National Institute of Standards, Technology, 2000. — P. 343—348. — URL: http://dblp.uni-trier.de/db/conf/aes/aes2000.html#DaemenR00.

27. *Dobbertin*, *H.* Construction of Bent Functions and Balanced Boolean Functions with High Nonlinearity. / H. Dobbertin // FSE. Vol. 1008 / ed. by B. Preneel. — Springer, 1994. — P. 61—74. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse94.html#Dobbertin94.

28. First-Round and Last-Round Power Analysis Attack Against AES Devices / S. D. Putra, A. D. W. Sumari, I. Asrowardi, [et al.] // 2020 International Conference on Information Technology Systems and Innovation (ICITSI). — 2020. — P. 410—415.

29. *Fomin*, *D. B.* Implementation of an XSL block cipher with MDS-matrix linear transformation on NVIDIA CUDA / D. B. Fomin // Математические вопросы криптографии. — 2015. — Vol. 6, no. 2. — P. 99—108.

30. *Freyre-Echevarria*, *A.* On the Generation of Cryptographically Strong Substitution Boxes from Small Ones and Heuristic Search / A. Freyre-Echevarria // 10th Workshop on Current Trends in Cryptology (CTCrypt 2021). Pre-proceedings. — 2021. — P. 112—128.

31. *Gao*, *Y.* Side-Channel Attacks With Multi-Thread Mixed Leakage / Y. Gao, Y. Zhou // IEEE Transactions on Information Forensics and Security. — 2021. — Vol. 16. — P. 770—785.

32. Genetic Programming – An Introduction / W. Banzhaf, P. Nordin, R. E. Keller, [et al.]. — San Francisco, CA, USA : Morgan Kaufmann Publishers, 1998.

33. Block Ciphers That Are Easier to Mask: How Far Can We Go? / B. Gérard, V. Grosso, M. Naya-Plasencia, [et al.] // CHES. Vol. 8086 / ed. by G. Bertoni, J.-S. Coron. — Springer, 2013. — P. 383—399. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/ches/ches2013.html#GerardGNS13.

34. LS-Designs: Bitslice Encryption for Efficient Masked Software Implementations. / V. Grosso, G. Leurent, F.-X. Standaert, [et al.] // FSE. Vol. 8540 / ed. by C. Cid, C. Rechberger. — Springer, 2014. — P. 18—37. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse2014.html#GrossoLSV14.

35. *Gullasch*, *D.* Cache Games - Bringing Access-Based Cache Attacks on AES to Practice. / D. Gullasch, E. Bangerter, S. Krenn // IEEE Symposium on Security and Privacy. — IEEE Computer Society, 2011. — P. 490—505. — URL: http://dblp.uni-trier.de/db/conf/sp/sp2011.html#GullaschBK11.

36. *Heys*, *H. M.* A Tutorial on Linear and Differential Cryptanalysis. / H. M. Heys // Cryptologia. — 2002. — Vol. 26, no. 3. — P. 189—221. — URL: http://dblp.uni-trier.de/db/journals/cryptologia/cryptologia26.html#Heys02.

37. *Hlavicka*, *J.* A Heuristic Boolean Minimizer / J. Hlavicka, P. Fiser // ICCAD’01. — 2001.

38. *Iwai*, *K.* Acceleration of AES encryption on CUDA GPU. / K. Iwai, N. Nishikawa, T. Kurokawa // Int. J. Netw. Comput. — 2012. — Vol. 2, no. 1. — P. 131—145. — URL: http://dblp.uni-trier.de/db/journals/ijnc/ijnc2.html#IwaiNK12.

39. *Jakobsen*, *T.* Attacks on Block Ciphers of Low Algebraic Degree / T. Jakobsen, L. R. Knudsen // Journal of Cryptology. — 2001. — Vol. 14, no. 3. — P. 197—210. — URL: https://doi.org/10.1007/s00145-001-0003-x.

40. Optimizing Implementations of Lightweight Building Blocks. / J. Jean, T. Peyrin, S. M. Sim, [et al.] // IACR Trans. Symmetric Cryptol. — 2017. — Vol. 2017, no. 4. — P. 130—168. — URL: http://dblp.uni-trier.de/db/journals/tosc/tosc2017.html#JeanPST17.

41. *Jiang*, *Z. H.* A complete key recovery timing attack on a GPU. / Z. H. Jiang, Y. Fei, D. R. Kaeli // HPCA. — IEEE Computer Society, 2016. — P. 394—405. — URL: http://dblp.uni-trier.de/db/conf/hpca/hpca2016.html#JiangFK16.

42. *Jiang*, *Z. H.* Exploiting Bank Conflict-based Side-channel Timing Leakage of GPUs. / Z. H. Jiang, Y. Fei, D. R. Kaeli // TACO. — 2020. — Vol. 16, no. 4. — 42:1—42:24. — URL: http://dblp.uni-trier.de/db/journals/taco/taco16.html#JiangFK20.

43. A Timing Side-Channel Attack on a Mobile GPU. / E. Karimi, Z. H. Jiang, Y. Fei, [et al.] // ICCD. — IEEE Computer Society, 2018. — P. 67—74. — URL: http://dblp.uni-trier.de/db/conf/iccd/iccd2018.html#KarimiJFK18.

44. *Kipper*, *M. S.* Implementing AES on GPU: Final Report / M. S. Kipper, J. Slavkin, D. Denisenko //. — 2011.

45. *Knudsen*, *L. R.* Non-Linear Approximations in Linear Cryptanalysis / L. R. Knudsen, M. J. B. Robshaw // Advances in Cryptology — EUROCRYPT ’96 / ed. by U. Maurer. — Berlin, Heidelberg : Springer Berlin Heidelberg, 1996. — P. 224—236.

46. *Kocher*, *P. C.* Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems / P. C. Kocher // Lecture Notes in Computer Science. — 1996. — Vol. 1109. — P. 104—113. — URL: citeseer.ist.psu.edu/kocher96timing.html.

47. *Komissarov*, *S. M.* On algorithmic implementation of 16-bit S-boxes with ARX and Butterfly structures / S. M. Komissarov // Prikl. Diskr. Mat. Suppl. — 2019. — Vol. 12. — P. 101—107. — (In Russian).

48. A performance prediction model for the CUDA GPGPU platform / K. Kothapalli, R. Mukherjee, M. S. Rehman, [et al.] // 2009 International Conference on High Performance Computing (HiPC). — 2009. — P. 463—472.

49. *Krasovsky*, *A. V.* Actual and Historical State of Side Channel Attacks Theory / A. V. Krasovsky, E. A. Maro // Proceedings of the 12th International Conference on Security of Information and Networks. — Sochi, Russia : Association for Computing Machinery, 2019. — (SIN '19). — URL: https://doi.org/10.1145/3357613.3357627.

50. *Kundi*, *D.-e.-S.* Implementation of T-box/T-1-Box Based AES Design on Latest Xilinx FPGA / D.-e.-S. Kundi, A. Aziz // Mehran University Research Journal of Engineering & Technology ISSN 0254-7821. — 2015. — Oct. — Vol. 34. — P. 441—446.

51. *Kutzner*, *S.* Enabling 3-Share Threshold Implementations for all 4-Bit S-Boxes. / S. Kutzner, P. H. Nguyen, A. Poschmann // ICISC. Vol. 8565 / ed. by H.-S. Lee, D.-G. Han. — Springer, 2013. — P. 91—108. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/icisc/icisc2013.html#KutznerNP13.

52. *Leander*, *G.* On Invariant Attacks / G. Leander //. — 2019. — Invited talk.

53. *Lim*, *C. H.* A Revised Version of Crypton - Crypton V1.0. / C. H. Lim // FSE. Vol. 1636 / ed. by L. R. Knudsen. — Springer, 1999. — P. 31—45. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse99.html#Lim99.

54. *Lim*, *C. H.* CRYPTON: A New 128-bit Block Cipher - Specification and Analysis / C. H. Lim. — 1998.

55. *Lo*, *O.* Correlation Power Analysis on the PRESENT Block Cipher on an Embedded Device. / O. Lo, W. J. Buchanan, D. Carson // ARES / ed. by S. Doerr, M. Fischer, S. Schrittwieser, [et al.]. — ACM, 2018. — 21:1—21:6. — URL: http://dblp.uni-trier.de/db/conf/IEEEares/ares2018.html#LoBC18.

56. *Malyshev*, *F. M.* Methods of linear and differential relations in cryptography / F. M. Malyshev // Diskr. Mat. — 2022. — Vol. 34, no. 1. — P. 36—63. — (In Russian).

57. *Matsuda*, *S.* Lightweight Cryptography for the Cloud: Exploit the Power of Bitslice Implementation. / S. Matsuda, S. Moriai // CHES. Vol. 7428 / ed. by E. Prouff, P. Schaumont. — Springer, 2012. — P. 408—425. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/ches/ches2012.html#MatsudaM12.

58. *Matsui*, *M.* New Block Encryption Algorithm MISTY. / M. Matsui // FSE. Vol. 1267 / ed. by E. Biham. — Springer, 1997. — P. 54—68. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse97.html#Matsui97.

59. *Matsui*, *M.* The First Experimental Cryptanalysis of the Data Encryption Standard. / M. Matsui // CRYPTO. Vol. 839 / ed. by Y. Desmedt. — Springer, 1994. — P. 1—11. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/crypto/crypto94.html#Matsui94.

60. *Menyachikhin*, *A. V.* Spectral-linear and spectral-differential methods for generating S-boxes having almost optimal cryptographic parameters / A. V. Menyachikhin // Математические вопросы криптографии. — 2017. — Vol. 8, no. 2. — P. 97—116.

61. Mixed Bases for Efficient Inversion in F((22)2)2 and Conversion Matrices of SubBytes of AES. / Y. Nogami, K. Nekado, T. Toyota, [et al.] // IEICE Trans. Fundam. Electron. Commun. Comput. Sci. — 2011. — Vol. 94—A, no. 6. — P. 1318—1327. — URL: http://dblp.uni-trier.de/db/journals/ieicet/ieicet94a.html#NogamiNTHM11.

62. *Nishikawa*, *N.* Implementation of Bitsliced AES Encryption on CUDA-Enabled GPU / N. Nishikawa, H. Amano, K. Iwai // Network and System Security / ed. by Z. Yan, R. Molva, W. Mazurczyk, [et al.]. — Cham : Springer International Publishing, 2017. — P. 273—287.

63. Mixed Bases for Efficient Inversion in $\mathbb{F}((2^2)^2)2$ and Conversion Matrices of SubBytes of AES / Y. Nogami, K. Nekado, T. Toyota, [et al.] //. — 08/2010. — P. 234—247.

64. *NVIDIA Corporation*. NVIDIA CUDA C Programming Guide. Design Guide / NVIDIA Corporation. — 2022. — URL: https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf ; Version 11.8.

65. *Olofsson*, *M.* VLSI Aspects on Inversion in Finite Fields : PhD thesis / Olofsson Mikael. — 02/2002.

66. *Perrin*, *L.* Cryptanalysis, Reverse-Engineering and Design of Symmetric Cryptographic Algorithms. : PhD thesis / Perrin Léo. — University of Luxembourg, 2017.

67. *Perrin*, *L.* Cryptanalysis of a Theorem: Decomposing the Only Known Solution to the Big APN Problem (Full Version). / L. Perrin, A. Udovenko, A. Biryukov // IACR Cryptology ePrint Archive. — 2016. — Vol. 2016. — P. 539. — URL: http://dblp.uni-trier.de/db/journals/iacr/iacr2016.html#PerrinUB16 ; http://eprint.iacr.org/2016/539.

68. *Raddum*, *H.* Algebraic Analysis of the Simon Block Cipher Family / H. Raddum //. Vol. 9230. — 08/2015. — P. 157—169.

69. *Rebeiro*, *C.* Bitslice Implementation of AES / C. Rebeiro, D. Selvakumar, A. S. L. Devi // Cryptology and Network Security / ed. by D. Pointcheval, Y. Mu, K. Chen. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2006. — P. 203—212.

70. *Rijmen*, *V.* The KHAZAD Block Cipher / V. Rijmen, P. Barreto. — 2000.

71. *Rudell*, *R.* Multiple-Valued Logic Minimization for PLA Synthesis / R. Rudell // Technical report, EECS Department, University of California, Berkeley. — 1986.

72. Rolled architecture based implementation of AES using T-Box / P. V. S. Shastry, N. Somani, A. Gadre, [et al.] //. — 08/2012. — P. 626—630.

73. Side-channel power analysis of a GPU AES implementation / C. Luo, Y. Fei, P. Luo, [et al.] // 2015 33rd IEEE International Conference on Computer Design (ICCD). — 2015. — P. 281—288.

74. *Stallings*, *W.* The Whirlpool Secure Hash Function. / W. Stallings // Cryptologia. — 2006. — Vol. 30, no. 1. — P. 55—67. — URL: http://dblp.uni-trier.de/db/journals/cryptologia/cryptologia30.html#Stallings06.

75. ICEBERG : An Involutional Cipher Efficient for Block Encryption in Reconfigurable Hardware. / F.-X. Standaert, G. Piret, G. Rouvroy, [et al.] // FSE. Vol. 3017 / ed. by B. K. Roy, W. Meier. — Springer, 2004. — P. 279—299. — (Lecture Notes in Computer Science). — URL: http://dblp.uni-trier.de/db/conf/fse/fse2004.html#StandaertPRQL04.

76. *Sun*, *B.* New Cryptanalysis of Block Ciphers with Low Algebraic Degree / B. Sun, L. Qu, C. Li // Fast Software Encryption / ed. by O. Dunkelman. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2009. — P. 180—192.

77. Highly efficient GF(28) inversion circuit based on hybrid GF representations. / R. Ueno, N. Homma, Y. Nogami, [et al.] // J. Cryptographic Engineering. — 2019. — Vol. 9, no. 2. — P. 101—113. — URL: http://dblp.uni-trier.de/db/journals/jce/jce9.html#UenoHNA19.

78. Finding Optimal Bitsliced Implementations of 4 x 4-bit S-boxes / M. Ullrich, C. D. Cannière, S. Indesteege, [et al.]. — 2011. — Ecrypt II.

79. *Zhou*, *Y.* Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing / Y. Zhou, D. Feng. — 2005. — URL: http://eprint.iacr.org/2005/388.

Author's publications on the topic of this dissertation

80. A compact bit-sliced representation of Kuznyechik S-box / O. D. Avraamova, D. B. Fomin, V. A. Serov, [et al.] // Mat. Vopr. Kriptogr. — 2021. — Vol. 12, no. 2. — P. 21—38.

81. *Fomin*, *D. B.* A timing attack on CUDA implementations of an AES-type block cipher / D. B. Fomin // Mat. Vopr. Kriptogr. — 2016. — Vol. 7, no. 2. — P. 121—130.

82. *Fomin*, *D. B.* Construction of permutations on the space V_{2m} by means of $(2m,m)$-functions / D. B. Fomin // Mat. Vopr. Kriptogr. — 2020. — Vol. 11, no. 3. — P. 121—138. — (In Russian).

83. *Fomin*, *D. B.* New Classes of 8-bit Permutations Based on a Butterfly Structure / D. B. Fomin // Mat. Vopr. Kriptogr. — 2019. — Vol. 10, no. 2. — P. 169—180.

84. *Fomin*, *D. B.* O podhodah k postroeniyu nizkoresursnyh nelinejnyh preobrazovanij / D. B. Fomin // Obozrenie prikladnoj i promyshlennoj matematiki. — 2018. — Vol. 25, no. 4. — P. 379—381. — (In Russian).

85. *Fomin*, *D. B.* On the algebraic degree and differential uniformity of permutations on the space V_{2m} constructed via $(2m, m)$-functions / D. B. Fomin // Mat. Vopr. Kriptogr. — 2020. — Vol. 11, no. 4. — P. 133—149. — (In Russian).

86. *Fomin*, *D. B.* On the impossibility of an invariant attack on Kuznyechik / D. B. Fomin // Journal of Computer Virology and Hacking Techniques. — 2022. — Vol. 18, no. 1. — P. 61—67.

87. *Fomin*, *D. B.* On the way of constructing differentially 2δ-uniform permutations over $\mathbb{F}_{2^{2m}}$ / D. B. Fomin // Prikl. Diskr. Mat. Suppl. — 2021. — Vol. 14. — P. 51—55. — (In Russian).

88. *Fomin*, *D. B.* Hardware implementation of one class of 8-bit permutations / D. B. Fomin, D. I. Trifonov // Prikl. Diskr. Mat. Suppl. — 2019. — Vol. 12. — P. 134—137. — (In Russian).

89. *Kovrizhnykh*, *M. A.* Heuristic algorithm for obtaining permutations with given cryptographic properties using a generalized construction / M. A. Kovrizhnykh, D. B. Fomin // Prikl. Diskr. Mat. — 2022. — Vol. 57. — P. 5—21. — (In Russian).

90. *Kovrizhnykh*, *M. A.* On a heuristic approach to constructing bijective vector Boolean functions with given cryptographic properties / M. A. Kovrizhnykh, D. B. Fomin // Prikl. Diskr. Mat. Suppl. — 2021. — Vol. 14. — P. 181—184. — (In Russian).

91. *Kovrizhnykh*, *M. A.* On differential uniformity of permutations derived using a generalized construction / M. A. Kovrizhnykh, D. B. Fomin // Mat. Vopr. Kriptogr. — 2022. — Vol. 13, no. 2. — P. 37—52.

92. *Trifonov*, *D. I.* Invariant subspaces in SPN block cipher / D. I. Trifonov, D. B. Fomin // Prikl. Diskr. Mat. — 2021. — Vol. 54. — P. 58—76. — (In Russian).