МАТЕМАТИЧЕСКИЕ МЕТОДЫ КРИПТОГРАФИИ

UDC 519.7 DOI 10.17223/20710410/44/3

CRYPTANALYTIC CONCEPT OF FINITE AUTOMATON INVERTIBILITY WITH FINITE DELAY[1]

G. P. Agibalov

National Research Tomsk State University, Tomsk, Russia

E-mail: agibalov@mail.tsu.ru

The automaton invertibility with a finite delay plays a very important role in the analysis and synthesis of finite automata cryptographic systems. The automaton cryptanalitic invertibility with a finite delay τ is studied in the paper. From the cryptanalyst's point of view, this notion means the theoretical possibility for recovering, under some conditions, a prefix α of a length n in an unknown input sequence $\alpha\delta$ of an automaton from its output sequence γ of the length $n+\tau$ and perhaps an additional information such as parameters τ and n, initial (q), intermediate (θ) or final (t) state of the automaton or the suffix δ of the length τ in the input sequence. The conditions imposed on the recovering algorithm require for prefix α to be arbitrary and may require for the initial state q and suffix δ to be arbitrary or existent, that is, the variable α is always bound by the universal quantifier and each of variables q and δ may be bound by any of quantifiers — universal ($\forall$) or existential ($\exists$) one. The variety of information, which can be known to a cryptanalyst, provides many different types of the automaton invertibility and, respectively, many different classes of invertible automata. Thus, in the paper, an invertibility with a finite delay τ of a finite automaton A is the ability of this automaton to resist recovering or, on the contrary, to allow precise determining any input word α of a length n for the output word γ being the result of transforming by the automaton A in its initial state q the input word $\alpha\delta$ with the δ of length τ and with the known n, τ, A, γ and $\upsilon \subseteq \{\delta, q, \theta, t\}$ where q and δ may be arbitrary or some elements in their sets and θ and t are respectively intermediate and final states of A into which A comes from q under acting of input words α and $\alpha\delta$ respectively. According to this, the automaton A is called invertible with a delay τ if there exists a function $f(\gamma,\upsilon)$ and a triplet of quantifiers $\varkappa \in \{Q_1x_1Q_2x_2Q_3x_3 : Q_ix_i \in \{\forall q, \exists q, \forall\alpha, \forall\delta, \exists\delta\}, i \neq j \Rightarrow x_i \neq x_j\}$ such that $\varkappa[f(\gamma,\upsilon) = \alpha]$; in this case f is called a recovering function, $(\varkappa,\upsilon)$ — an invertibility type, $\varkappa$ — an invertibility degree, υ — an invertibility order of the automaton A and $\exists f \varkappa[f(\gamma,\upsilon) = \alpha]$ — an invertibility condition of type $(\varkappa,\upsilon)$ for the automaton A. So, 208 different types of the automaton A invertibility are defined at all. The well known types of (strong) invertibility and weak invertibility described for finite automata earlier by scientists (D. A. Huffman, A. Gill, Sh. Even, A. A. Kurmit, Z. D. Dai, D. F. Ye, K. Y. Lam, R. Tao and many others) in our theory belong to types $(\forall q\forall\alpha\forall\delta, \varnothing)$ and $(\forall q\forall\alpha\forall\delta, \{q\})$ respectively. For every invertibility type, we have defined a class of automata with this type of invertibility and described the inclusion relation on the set of all these classes. It has turned out that the graph of this relation is the union of twenty nine lattices with thirteen of them each containing sixteen

[1] The author is supported by the RFBR-grant no. 17-01-00354.

classes and sixteen lattices each containing thirteen classes. To solve the scientific problems (invertability tests, synthesis of inverse automata and so on) related to the different and concrete invertibility classes, we hope to continue these investigations.

Keywords: *finite automata, information-lossless automata, automata invertibility, cryptanalytic invertibility.*

Introduction

In the theory of analysis and synthesis of finite automaton cryptosystems, the invertibility property of finite automata takes the most important place. From cryptanalytic point of view, it means the theoretical possibility to recover a nonempty part of input word of an automaton using its output word and, possibly, some additional information about the automaton — about its transition and output functions, about its states — initial, intermediate or final, about its class, etc and about the rest of input word playing an auxiliary (often — official) role — about its length, value and location in the input word. A variety of this information kinds generates the different types of the automaton invertibility and, respectively, the different classes of the invertible automata. In this paper, we assume that the transition and output functions of the automata under consideration are completely known, a nonempty prefix of an input word need be recovered so that the length of the next part of the word following after the prefix and called the recovering or invertibility delay are also known.

So under the invertibility with a finite delay τ of a finite automaton A, we understand the property of A which allows uniquely compute its any input word α using an output word γ produced by the automaton A in an initial state q as its reaction to an input word $\alpha\delta$ with δ of the length τ, with the known τ, A, γ, and with the unknown, possibly, some or all the values from the list $\upsilon \subseteq \{\delta, q, \theta, t\}$, where q and δ can be arbitrary or some elements of their sets, θ and t are, respectively, intermediate and final states of the automaton A, into which it comes from q under the influence of input words α and $\alpha\delta$ respectively.

According to this, the automaton A is called *invertible* with the delay τ if there exist a function $f(\gamma, \upsilon)$ and a triplet of quantifiers $\varkappa \in \{Q_1x_1Q_2x_2Q_3x_3 : Q_ix_i \in \in \{\forall q, \exists q, \forall\alpha, \forall\delta, \exists\delta\}, i \neq j \Rightarrow x_i \neq x_j\}$ such that $\varkappa(f(\gamma, \upsilon) = \alpha)$; in this case, f is called recovering function, $(\varkappa, \upsilon)$ — invertibility type, $\varkappa$ — invertibility degree, υ — invertibility order of the automaton A and $\exists f \varkappa(f(\gamma, \upsilon) = \alpha)$ — invertibility condition for type $(\varkappa, \upsilon)$.

In the automata theory, a notion of information lossless automaton (ILA) are often used as a synonym to a notion of an invertible automaton. For the first time, ILAs were investigated by D. A. Huffman [1, 2] (his results can be also found in the monograph by A. Gill [3]), later — by Sh. Even [4] and also by A. A. Kurmit, who has described his own results on ILA in the detailed monograph [5] where ILA with a finite delay is considered with the known initial or final state and is called there ILA of a finite order, respectively, of I or of II type. In 1959, A. D. Zakrevsky [6] has proposed a symmetric cipher on the base of a strongly connected ILA with zero delay (with an output function being bijective for every state). For the sake of fairness, we need to say that first the similar automata were used by the Japanese during World War Two in their ciphering machine known as Purple [7]. Recently, the automaton invertibility became a research subject for Chinese scientists headed by professor R. Tao. They have produced FAPKC — Finite Automaton Public Key Cryptosystems based on memory finite automata which are invertible with finite delay and with known initial state [7 – 9].

The main results of the works enumerated above and related to the automaton invertibility with a finite delay are in reality the definitions and constructive tests of two types of invertibility — strong and weak (our types $(\forall q \forall \alpha \forall \delta, \varnothing)$ and $(\forall q \forall \alpha \forall \delta, \{q\})$ respectively) and algorithms for synthesis of inverse automata for them. These types of invertibility are really defined in the mentioned works through the automaton properties (classes) and afterwards it is proved that if an automaton belongs to a certain class, then the recovering its input word prefixes is possible. This looks like "a cart ahead of horse".

In our paper, a general definition for an arbitrary type of finite automaton invertibility is introduced. Every particular type of invertibility is obtained from this definition by setting particular values of the definition parameters which are the degree $\varkappa$ and the order υ of the invertibility. So, formally, 208 types of finite automaton invertibility with a delay are introduced in all, including types of strong and weak invertibility mentioned above from [3, 5, 9]. For every type of automaton invertibility with a fixed delay, we define the class of all finite automata invertible of this type and show that the set of all these classes partially ordered by the inclusion relation is the union of 29 lattices. The definition of the arbitrary type as well as of each particular type of an automaton invertibility is given in a completely clear and natural way, namely through the existence of a function recovering the unknown prefix of an automaton input word by using another known information. As for constructive tests for automaton invertibility of each type, they are supposed to be formulated and proved in terms of the automaton itself properties. A consequence of this fact is that the definitions of strong and weak invertibilities in monographs [3, 5, 9] are theorems in our theory. Besides, we have succeeded in defining and researching many such automaton invertibility types and classes which are not studied by other scientists. Of course, we don't exclude that not all these classes are of high importance from science point of view, but the only existence of them induces people to thorough studying them for the purpose of solving some theoretical and applied problems, including establishment of necessary and sufficient conditions for automaton invertibility of each type; building up a constructive test for belonging a finite automaton to an invertibility class; algorithmic synthesis of the automata in a given invertibility class; characterization of the invertible automata, to which inverse automata exist, and algorithmic synthesis of the last; development of the effective algorithms for recovering word prefix on the input of an invertible automaton in a particular invertibility class; creation of private and public key cryptosystems on the basis of invertible automata of different invertibility classes; algorithmic cryptanalysis of these cryptosystems.

The solutions of these problems and their research in computer experiments are supposed to perform by the author and his colleagues for several future years with regular publications of the results in the journal "Prikladnaya Diskretnaya Matematika" and their presentation on the International Conference "Computer Security and Cryptography" — SIBECRYPT.

1. Agreements

An arbitrary finite automaton is presented as $A = (X, Q, Y, \psi, \varphi)$, where X, Q and Y are its input alphabet, the set of states and the output alphabet respectively; ψ and φ — its functions, respectively, of transitions and outputs, $\psi : X \times Q \to Q$ and $\varphi : X \times Q \to Y$. The functions, being defined for pairs $xq \in X \times Q$, we extend to pairs $\alpha q \in X^* \times Q$ by induction on the length $|\alpha|$ of the word $\alpha \in X^*$, namely the functions $\psi : X^* \times Q \to Q$ and $\bar{\varphi} : X^* \times Q \to Y^*$ are defined as $\psi(\Lambda, q) = q$, $\psi(\alpha\beta, q) = \psi(\beta, \psi(\alpha, q))$, $\bar{\varphi}(\Lambda, q) = \Lambda$, $\bar{\varphi}(x, q) = \varphi(x, q)$ and $\bar{\varphi}(\alpha\beta, q) = \bar{\varphi}(\alpha, q)\bar{\varphi}(\beta, \psi(\alpha, q))$. Here and everywhere further, the symbol Λ denotes the empty word in any alphabet.

Thus, $\psi(\alpha, q)$ is a state, into which the automaton A comes from a state q under the influence of input word α, and $\bar{\varphi}(\alpha, q)$ is an output word which the automaton produces this time. The function $\varphi_q : X \to Y$, defined as $\varphi_q(x) = \varphi(x, q)$ for all $x \in X$, is called the output function of the automaton A in the state $q \in Q$.

We don't exclude partially defined automata from the consideration. For presentation of the information in them, we use the symbol ϖ, regarding it as any word of any length and over any alphabet. So the record $\varpi \in X^n$, for example, means that ϖ is a word of a length n in the alphabet X and the record $f(a) = \varpi$ — that a function value $f(x)$ is not defined for $x = a$. In comparison between two words in the same alphabet, we consider that the word ϖ equals a word α iff $|\varpi| = |\alpha|$. In particular, two words ϖ coincide iff their lengths are equal.

Further, we adopt the convention for any logical formula

$$F = Q_1 x_1 Q_2 x_2 \ldots Q_n x_n P(x_1, x_2, \ldots, x_n),$$

where $Q_1, Q_2, \ldots, Q_n$ are the symbols of quantifiers $\forall, \exists$ and the formula P doesn't contain quantifiers, to say that a n-tuple $c_1 c_2 \ldots c_n$ of values of variables $x_1, x_2, \ldots, x_n$ satisfies F if, for each $i = 1, 2, \ldots, n$, the value c_i is chosen in the range of the variable x_i in the following way: in case $Q_i = \forall$ — anyhow, in case $Q_i = \exists$ — (in dependence on already chosen c_j for $j < i$) so that $P(c_1, c_2, \ldots, c_n) = \texttt{true}$. By the definition of the truth of F, such a tuple exists if and only if $F = \texttt{true}$.

Finally, everywhere further, by the symbol τ we denote a non-negative integer called a delay and, in the absence of additional remarks, it is supposed that $a \in X$, $b \in X$, $\alpha \in X^*$, $\beta \in X^*$, $\delta \in X^\tau$, $\varepsilon \in X^\tau$, $q \in Q$, $s \in Q$.

2. Definition of invertibility with finite delay

Consider a finite automaton $A = (X, Q, Y, \psi, \varphi)$. Let q, α, δ be variables with values in Q, X^*, X^τ denoting, respectively, an initial state, a prefix (beginning) and suffix (ending) of an input word $\alpha\delta$ of the automaton A and $K = \{\forall q, \forall \alpha, \forall \delta, \exists q, \exists \delta\}$ be the set of universal and existential quantifiers which bind these variables. In reality, the quantifiers in K are $\forall q \in Q$, $\forall \alpha \in X^*$, $\forall \delta \in X^\tau$, $\exists q \in Q$, $\exists \delta \in X^\tau$ without previously fixed symbols indicating ranges of variables in question and omitted in K for conciseness of record. Besides, notice that K doesn't contain the quantifier $\exists \alpha$. This is because, for a cryptanalyst, the input word α can be any one.

Also let $\theta = \psi(\alpha, q)$, $t = \psi(\alpha\delta, q)$ and $V = \{\Lambda, q, \theta, t, \delta, q\theta, qt, q\delta, \theta t, \theta\delta, t\delta, q\theta t, q\theta\delta, qt\delta, \theta t\delta, q\theta t\delta\}$. It is seen that symbols θ and t denote an intermediate and final states, into which the automaton A comes from the state q after having received on its input the words α and $\alpha\delta$ respectively. The members of the set V are meant for describing what we call here an invertibility order of the automaton A. In fact, they are some functions in q, α, δ.

We say that the automaton A is *invertible with the delay* τ if there exist quantifiers K_1, K_2, K_3 in K with different variables from $\{q, \alpha, \delta\}$ as well as a function $f : Y^* \times V \to X^*$ and a tuple $v(q, \alpha, \delta) \in V$ such that the following formula is true

$$\Phi = K_1 K_2 K_3 (f(\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta)) = \alpha);$$

in this case, $(K_1 K_2 K_3, v)$ is called *invertibility type* of the automaton A, $K_1 K_2 K_3$ — *invertibility degree*, v — *invertibility order*, f — *recovering function* (for input prefix), τ — *recovery delay*, or *invertibility delay* and $\exists f[\Phi]$ — *invertibility condition* of this type for the automaton A.

Taking into account the commutativity of the same type quantifiers, in the table for the automaton A, we present invertibility conditions of all possible invertibility types with the delay τ. From this table, for an invertibility of a type $(K_1K_2K_3, v)$, the invertibility condition is obtained by attaching the quantifier prefix $\exists f K_1K_2K_3$ from the left column to the (so called) underlying expression $f(\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta)) = \alpha$ from the right column with the proper invertibility order v. Further, the invertibility condition with the quantifier prefix of a number i and its underlying expression of a number j in the table is denoted by $U_{i,j}$ or (if you need to know τ) $U_{i,j}[\tau]$. For example, $U_{1,1}[\tau] = \exists f\forall q\forall\alpha\forall\delta(f(\bar{\varphi}(\alpha\delta, q)) = \alpha)$, $U_{1,2}[\tau] = \exists f\forall q\forall\alpha\forall\delta(f(\bar{\varphi}(\alpha\delta, q), q) = \alpha)$, $U_{5,10}[\tau] = \exists f\exists q\forall\alpha\exists\delta(f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha, q), \delta) = \alpha)$ and so on. Thus, for any finite automaton, we have formally defined $208(= 13\cdot 16)$ invertibility types with any finite delay. But later, we will see that, for some of these types with different invertibility orders, the invertibility conditions can be equivalent and define the same type of invertible automata.

Conditions for different types of invertibility with a delay τ of the automaton A

No	Quantifier prefix $\exists f Q_1x_1Q_2x_2Q_3x_3$	No	Underlying expression $f(\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta)) = \alpha$
		1	$f(\bar{\varphi}(\alpha\delta, q)) = \alpha$
		2	$f(\bar{\varphi}(\alpha\delta, q), q) = \alpha$
		3	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha, q)) = \alpha$
1	$\exists f\,\forall q\,\forall\alpha\,\forall\delta$	4	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha\delta, q)) = \alpha$
2	$\exists f\,\forall q\,\forall\alpha\,\exists\delta$	5	$f(\bar{\varphi}(\alpha\delta, q), \delta) = \alpha$
3	$\exists f\,\forall q\,\exists\delta\,\forall\alpha$	6	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha, q)) = \alpha$
4	$\exists f\,\exists q\,\forall\alpha\,\forall\delta$	7	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha\delta, q)) = \alpha$
5	$\exists f\,\exists q\,\forall\alpha\,\exists\delta$	8	$f(\bar{\varphi}(\alpha\delta, q), q, \delta) = \alpha$
6	$\exists f\,\exists q\,\exists\delta\,\forall\alpha$	9	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha, q), \psi(\alpha\delta, q)) = \alpha$
7	$\exists f\,\forall\alpha\,\exists q\,\forall\delta$	10	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha, q), \delta) = \alpha$
8	$\exists f\,\forall\alpha\,\exists q\,\exists\delta$	11	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha\delta, q), \delta) = \alpha$
9	$\exists f\,\forall\alpha\,\forall\delta\,\exists q$	12	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha, q), \psi(\alpha\delta, q)) = \alpha$
10	$\exists f\,\forall\alpha\,\exists\delta\,\forall q$	13	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha, q), \delta) = \alpha$
11	$\exists f\,\forall\delta\,\exists q\,\forall\alpha$	14	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha\delta, q), \delta) = \alpha$
12	$\exists f\,\exists\delta\,\forall q\,\forall\alpha$	15	$f(\bar{\varphi}(\alpha\delta, q), \psi(\alpha, q), \psi(\alpha\delta, q), \delta) = \alpha$
13	$\exists f\,\exists\delta\,\forall\alpha\,\exists q$	16	$f(\bar{\varphi}(\alpha\delta, q), q, \psi(\alpha, q), \psi(\alpha\delta, q), \delta) = \alpha$

Having a function $f : Y^* \times V \to X^*$, we can define a function $f' : Y^* \times V \to X^*$ so that $f'(\gamma y, v) = f(\gamma, v)$ for all $\gamma \in Y^*$, $y \in Y$ and $v \in V$. Since $\bar{\varphi}(\alpha\delta x, q) = \bar{\varphi}(\alpha\delta, q)\varphi(x, \psi(\alpha\delta, q))$, the equality $f(\bar{\varphi}(\alpha\delta, q), v) = \alpha$ implies the equality $f'(\bar{\varphi}(\alpha\delta x, q), v) = \alpha$. By the principle of mathematical induction, this implication proves that if, for a type of invertibility, a finite automaton is invertible with a finite delay, then, for the same type of invertibility, the automaton is invertible with any greater integer delay.

In this work, by the invertibility, we only understand an invertibility of a finite automaton, of a certain type, of a certain order, and of a finite delay and usually don't mention these its attributes without a particular need.

3. Invertibility classes

For any $i \in \{1, 2, \ldots, 13\}$ and $j \in \{1, 2, \ldots, 16\}$, we say that an automaton $A = (X, Q, Y, \psi, \varphi)$ belongs to an *(invertibility) class* $C_{i,j}[\tau]$ if the condition $U_{i,j}[\tau]$ is true; in this case, the condition $U_{i,j}[\tau]$ is called the *invertibility condition in the class* $C_{i,j}[\tau]$ of the automaton A. The purpose of this paragraph is the description of the inclusion relation on the set of all invertibility classes with a particular delay, following from the property: if $U_{i,j}[\tau] \Rightarrow U_{k,l}[\tau]$, then $C_{i,j}[\tau] \subseteq C_{k,l}[\tau]$. There are two cases when the premise

$U_{i,j}[\tau] \Rightarrow U_{k,l}[\tau]$ in this property takes place and this fact is recognized immediately by the invertibility types $(K_1K_2K_3, v)$ in $U_{i,j}[\tau]$ and $(K'_1K'_2K'_3, v')$ in $U_{k,l}[\tau]$:

1) $i = k$, $j \neq l$ and all the elements in v are contained in v';
2) $i \neq k$, $j = l$ and $K_1K_2K_3P(q, \alpha, \delta) \Rightarrow K'_1K'_2K'_3P(q, \alpha, \delta)$ for any predicate P in three variables.

For instance, in the first case, $U_{i,5} \Rightarrow U_{i,13}$ and therefore, $C_{i,5} \subseteq C_{i,13}$ for all i and, in the second case, $U_{7,j} \Rightarrow U_{9,j}$ and therefore, $C_{7,j} \subseteq C_{9,j}$ for all j.

In the case 2, truth (or falsehood) of pointed out implication is easy established with the help of identically true formulas of predicate logic such as $\forall x S(x) \Rightarrow \exists x S(x)$, $\exists x \forall y R(x, y) \Rightarrow \forall y \exists x R(x, y)$ and the like.

The implication $U_{i,j}[\tau] \Rightarrow U_{k,l}[\tau]$, connecting the invertibility conditions for two automata, induces the inclusion relation $C_{i,j}[\tau] \subseteq C_{k,l}[\tau]$ between the invertibility classes of these automata. On every of sets $\{C_{i,j}[\tau] : j = 1, 2, \ldots, 16\}$, $i = 1, \ldots, 13$, and $\{C_{i,j}[\tau] : i = = 1, 2, \ldots, 13\}$, $j = 1, \ldots, 16$, this relation defines a lattice — a partially ordered set, in which, for every pair of elements, there exist the least upper and the greatest lower bounds. These lattices are shown in the Figs. 1 and 2.

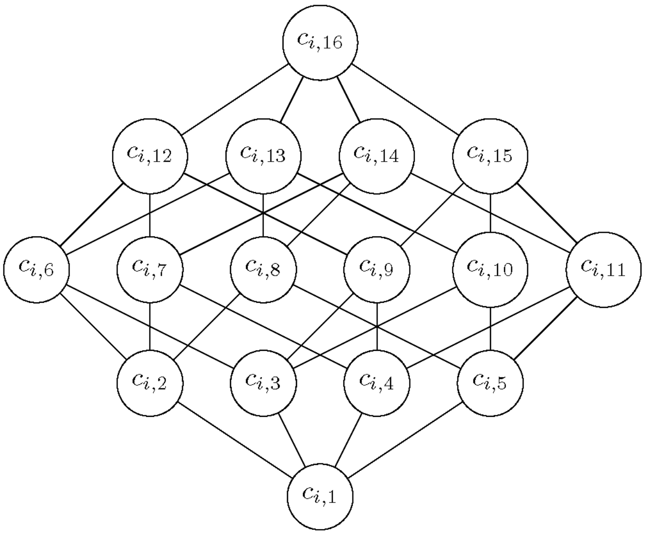

Fig. 1

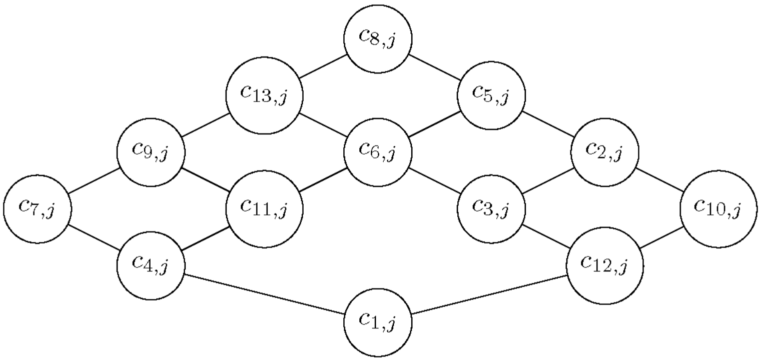

Fig. 2

In the case $\tau = 0$, the sequence δ in quantifier prefixes and underlying expressions in table is the empty word and, as a consequence, all these prefixes and expressions break up the blocks of equal entities, namely: the first — up the blocks $\bar{1} = \{1, 2, 3, 10, 12\}$, $\bar{4} = \{4, 5, 6, 11\}$ и $\bar{7} = \{7, 8, 9, 13\}$; the second — up the blocks $1' = \{1, 5\}$, $2' = \{2, 8\}$, $3' = \{3, 4, 9, 10, 11, 15\}$ and $6' = \{6, 7, 12, 13, 14, 16\}$. Hence $C_{i,j}[0] = C_{k,j}[0]$ for all $i, k \in I$, $I = \bar{1}, \bar{4}, \bar{7}$ and $j = 1, 2, \ldots, 16$, as well $C_{i,j}[0] = C_{i,l}[0]$ for all $j, l \in J$, $J = 1', 2', 3', 6'$ and $i = 1, 2, \ldots, 13$.

4. Automata invertibility problems

The automaton invertibility conditions which are contained in the definition of this notion for its different types are given in a non-constructive form and it is difficult to apply it in practice. Formulation and correct proof of constructive tests for invertibility of every type is the first in the row of problems related to the cryptanalytic notion of finite automaton invertibility.

From cryptographic point of view, in this row the problem of generating invertible automata of all possible types takes an important place. In different settings of this problem, many different requirements to automata under generation can present — with an equal probability in a certain class, with limited complexity, with great or, on the contrary, little invertibility delay and the like. Its solution seems to be impossible without a proper solution of the first problem.

The notion of finite automaton invertibility under consideration doesn't imply the obligatory existence of an inverse automaton to an invertible automaton. Moreover, it is possible that the function recovering an input prefix can not be finite-automated one for some types of automaton invertibility. In this case evidently the problem appears: given an invertible (of a certain type) automaton, find out whether it has an inverse automaton and if it has, then construct the inverse to it. The solution of this problem in turn implies the definition of inverse to any automaton of every invertibility class. In the absence of inverse automata to the automata of an invertibility class, we have the problem of constructing for them functions recovering prefixes of input sequences under known output sequences.

In subsequent investigations by the author and his colleagues, some of these problems are meant to be solved for some of invertibility classes defined.

5. Invertibility conditions

For investigating the properties of the automaton invertibility, the invertibility condition in its definition need to be re-formulated in a more constructive way and first of all to get out of request for explicit testing the existence of a recovering function. In this section, we present a test (Proposition 1) for automaton invertibility of any type $(\forall q \forall \alpha \forall \delta, v(q, \alpha, \delta))$ and give some necessary conditions (Proposition 2) for an automaton to be invertible of any type $(Q_1 q Q_2 \alpha Q_3 \delta, v(q, \alpha, \delta))$ both (test and necessary conditions) without explicit performance of a procedure of testing the existence of a recovering function. The propositions follow from the corresponding auxiliary lemmas about logical formulas. To formulate lemmas, we first introduce some needed symbols.

Let n be a positive integer; $Q_1, \ldots, Q_n$ be symbols of quantifiers, $Q_k \in \{\forall, \exists\}$, $k \in \{1, \ldots, n\}$; $x_1, \ldots, x_n, y_1, \ldots, y_n$ be different subject variables and D_i be the range of x_i and y_i for $i \in \{1, \ldots, n\}$. Also let $g(x_1, \ldots, x_n)$ be a function in variables $x_1, \ldots, x_n$ with a range D_g, $k_0 \in \{1, \ldots, n\}$, and $Q_{k_0} = \forall$. Finally, let $f : D_g \to D_{k_0}$ denotes an arbitrary function with the domain D_g and the range D_{k_0}. Consider a logical formula

$$Q_1 x_1 Q_2 x_2 \ldots Q_n x_n (f(g(x_1, x_2, \ldots, x_n)) = x_{k_0}) \tag{1}$$

in the normal form, that is, with the quantifier prefix $Q_1x_1Q_2x_2 \dots Q_nx_n$ and a underlying equality $f(g(x_1, x_2, \dots, x_n)) = x_{k_0}$ without quantifiers.

Lemma 1. In the case $Q_1 = \dots = Q_n = \forall$ the function f with the property (1) exists if and only if

$$\forall x_1 \dots \forall x_n \forall y_1 \dots \forall y_n (x_{k_0} \neq y_{k_0} \Rightarrow g(x_1, \dots, x_n) \neq g(y_1, \dots, y_n)). \tag{2}$$

Proof. Necessity. Take any $x_1, \dots, x_n, y_1, \dots, y_n$, where $x_{k_0} \neq y_{k_0}$. By the condition (1), $f(g(x_1, x_2, \dots, x_n)) \neq f(g(y_1, y_2, \dots, y_n))$. Therefore, in view of functionality of f, we obtain $g(x_1, x_2, \dots, x_n) \neq g(y_1, y_2, \dots, y_n)$.

Sufficiency. For any $x_1, \dots, x_n$, let $f(g(x_1, x_2, \dots, x_n)) = x_{k_0}$. This definition of f is correct since, by the condition (2), if for some $x_1, \dots, x_n, y_1, \dots, y_n$ the equality $g(x_1, x_2, \dots, x_n) = g(y_1, y_2, \dots, y_n)$ holds, then $x_{k_0} = y_{k_0}$. ■

Taking in lemma 1 $n = 3$, $x_1 = q$, $x_2 = \alpha$, $x_3 = \delta$, $y_1 = s$, $y_2 = \beta$, $y_3 = \varepsilon$, $g(x_1, \dots, x_n) = (\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta))$, $g(y_1, \dots, y_n) = (\bar{\varphi}(\beta\varepsilon, s), v(s, \beta, \varepsilon))$, and $k_0 = 2$, $x_{k_0} = \alpha$, $y_{k_0} = \beta$, we get

Proposition 1. The automaton A is invertible of the type $(\forall q \forall \alpha \forall \delta, v(q, \alpha, \delta))$, that is,

$$\exists f \forall q \forall \alpha \forall \delta (f(\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta)) = \alpha)$$

if and only if

$$\forall q \forall \alpha \forall \delta \forall s \forall \beta \forall \varepsilon (\alpha \neq \beta \Rightarrow (\bar{\varphi}(\alpha\delta, q), v(q, \alpha, \delta)) \neq (\bar{\varphi}(\beta\varepsilon, s), v(s, \beta, \varepsilon)).$$

Lemma 2. For any true quantifier logic formulas in a normal form

$$Q_1z_1 \dots Q_mz_mA(z_1, \dots, z_m) \quad \text{and} \quad R_1z_1 \dots R_mz_mB(z_1, \dots, z_m),$$

where $Q_i, R_i \in \{\forall, \exists\}$ and $Q_iR_i \neq \exists\exists$ for every $i \in \{1, \dots, m\}$, there exist some values $c_1, \dots, c_m$ of variables $z_1, \dots, z_m$ respectively such that $A(c_1, \dots, c_m) = B(c_1, \dots, c_m) = \texttt{true}$.

Proof. Applying the induction scheme by integer $t \geqslant 1$, we will show that for any such an integer $t \leqslant m$ the equalities $Q_{t+1}z_{t+1} \dots Q_mz_mA(c_1, \dots, c_t, z_{t+1}, \dots, z_m) = R_{t+1}z_{t+1} \dots R_mz_mB(c_1, \dots, c_t, z_{t+1}, \dots, z_m) = \texttt{true}$ take place and under $t = m$ we will obtain the state of the lemma.

For $t = 0$, the equality under proof is true by the condition. Assuming that it is true for any $t \leqslant j$ where j is an integer and $0 \leqslant j < m$, and taking as c_{j+1} any value of the variable z_{j+1} in the case $Q_{j+1} = R_{j+1} = \forall$ and a value of z_{j+1} under which $Q_{j+2}z_{j+2} \dots Q_mz_mA(c_1, \dots, c_{j+1}, z_{j+2}, \dots, z_m) = \texttt{true}$ or $R_{j+2}z_{j+2} \dots R_mz_mB(c_1, \dots, c_{j+1}, z_{j+2}, \dots, z_m) = \texttt{true}$ in the case $Q_{j+1} = \exists$ or $R_{j+1} = \exists$ respectively, we obtain that it is also true for $t = j + 1$. ■

Lemma 3. For any function g, if there exists a function f with the property (1), then

$$Q_1x_1 \dots Q_nx_nQ_1y_1 \dots Q_ny_n(x_{k_0} \neq y_{k_0} \Rightarrow g(x_1, \dots, x_n) \neq g(y_1, \dots, y_n)). \tag{3}$$

Proof. Formulas

$$Q_1x_1 \dots Q_nx_n(f(g(x_1, \dots, x_n)) = x_{k_0}), \quad Q_1y_1 \dots Q_ny_n(f(g(y_1, \dots, y_n)) = y_{k_0})$$

are equivalent. Therefore, by the condition (1)

$$Q_1x_1\dots Q_nx_n(f(g(x_1,\dots,x_n))=x_{k_0})\ \&\ Q_1y_1\dots Q_ny_n(f(g(y_1,\dots,y_n))=y_{k_0}).$$

Hence,

$$Q_1x_1\dots Q_nx_nQ_1y_1\dots Q_ny_n(f(g(x_1,\dots,x_n))=x_{k_0}\ \&\ f(g(y_1,\dots,y_n))=y_{k_0}). \tag{4}$$

Suppose, the state (3) is false and its negation, that is, the following state is true:

$$Q'_1x_1\dots Q'_nx_nQ'_1y_1\dots Q'_ny_n(x_{k_0}\neq y_{k_0}\ \&\ g(x_1,\dots,x_n)=g(y_1,\dots,y_n)), \tag{5}$$

where, for any $j\in\{1,\dots,n\}$, the symbol Q'_j is a dual quantifier, namely $\forall'=\exists$ and $\exists'=\forall$. By the lemma 2 related to the formulas (4) and (5), there exist values $a_1,\dots,a_n$ of the variables $x_1,\dots,x_n$ and values $b_1,\dots,b_n$ of variables $y_1,\dots,y_n$ respectively such that $f(g(a_1,\dots,a_n))=a_{k_0}$, $f(g(b_1,\dots,b_n))=b_{k_0}$ and $a_{k_0}\neq b_{k_0}$, $g(a_1,\dots,a_n)=g(b_1,\dots,b_n)$. A contradiction is obtained, namely: from one side, $a_{k_0}\neq b_{k_0}$, from another one, $a_{k_0}= f(g(a_1,\dots,a_n))=f(g(b_1,\dots,b_n))=b_{k_0}$. ■

Let x_1,x_2,x_3 and y_1,y_2,y_3 be the different variables from the sets $\{q,\alpha,\delta\}$ and $\{s,\beta,\varepsilon\}$ respectively such that if x_i is q, α or δ, then y_i is s, β or ε respectively, $Q_i\in\{\forall,\exists\}$, and if $x_i=\alpha$, then $Q_i=\forall$, $i=1,2,3$.

Proposition 2. If an automaton A is invertible of any type $(Q_1x_1Q_2x_2Q_3x_3, v(q,\alpha,\delta))$, that is, $\exists fQ_1x_1Q_2x_2Q_3x_3(f(\bar\varphi(\alpha\delta,q),v(q,\alpha,\delta))=\alpha)$, then

$$Q_1x_1Q_2x_2Q_3x_3Q_1y_1Q_2y_2Q_3y_3(\alpha\neq\beta\Rightarrow(\bar\varphi(\alpha\delta,q),v(q,\alpha,\delta))\neq(\bar\varphi(\beta\varepsilon,s),v(s,\beta,\varepsilon)).$$

Proof. The Proposition 2 follows from the Lemma 3 in the same way as the Proposition 1 follows from the Lemma 1. ■

REFERENCES

1. *Huffman D. A.* Canonical forms for information-lossless finite-state logical machines. IRE Trans. Circuit Theory, 1959, vol. 6, Spec. Suppl., pp. 41–59.
2. *Huffman D. A.* Notes on information-lossless finite-state automata. Nuovo Cimento, 1959, vol. 13, Suppl. 2, pp. 397–405.
3. *Gill A.* Introduction to the Theory of Finite-State Machines. N.Y., McGraw-Hill Book Company, 1962. 300 p.
4. *Even Sh.* On information-lossless automata of finite order. IEEE Trans. Electron. Comput., 1965, vol. 14, no. 4, pp. 561–569.
5. *Kurmit A. A.* Information Lossless Automata of Finite Order. N.Y., John Wiley Publ., 1974.
6. *Zakrevskiy A. D.* Metod avtomaticheskoy shifratsii soobshcheniy [The method for messages automatic encryption]. Prikladnaya Diskretnaya Matematika, 2009, no. 2(4), pp. 127–137. (in Russian)
7. *Agibalov G. P.* Konechnye avtomati v kriptografii [Finite automata in cryptography]. Prikladnaya Diskretnaya Matematika. Prilojenie, 2009, no. 2, pp. 43–73. (in Russian)
8. *Dai Z. D., Ye D. F., and Lam K. Y.* Weak invertibility of finite automata and cryptanalysis on FAPKC. LNCS, 1998, vol. 1514, pp. 227–241.
9. *Tao R.* Finite Automata and Application to Cryptography. N.Y., Springer, 2009. 406 p.