Computer Science Journal of Moldova, vol.**11**, no.3(33), 2003

Cryptographic primitives based on cellular transformations

B.V. Izotov

Abstract

Design of cryptographic primitives based on the concept of cellular automata (CA) is likely to be a promising trend in cryptography. In this paper, the improved method performing data transformations by using invertible cyclic CAs (CCA) is considered. Besides, the cellular operations (CO) as a novel CAs application in the block ciphers are introduced. Proposed CCAs and COs, integrated under the name of cellular transformations (CT), suit well to be used in cryptographic algorithms oriented to fast software and cheap hardware implementation.

Key words: cryptographic primitive, cellular automata, cellular operations, block cipher.

1 Introduction

Design of *cryptographic primitives* is one of the principal problems in creating effective block ciphers and hash functions. Here we face a conventional tradeoff between security, speed and implementation cost of encryption. In this connection, we consider one of the possible approaches to designing the block ciphers.

The cryptographic primitives are usually performed by (i) table transformations (substitutions and permutations) or (ii) operations represented in a set of computer instructions. In the first case, the complexity of such transformations with large input block forces one to split them into the small parts (S-boxes). This splitting is usually inconvenient for primitive implementation, and sometimes defines

©2003 by B.V. Izotov

prerequisites to attack the cipher. In the second case, in spite of software simplicity of computer instructions, these operations either (i) become complex in hardware implementation (for example, addition and multiplication modulo 2^n) or (ii) perform the weak elementary transformations (for example, cyclic rotation or bitwise addition and multiplication modulo 2).

Meanwhile, there exist wide capabilities to create the high-performance cryptographic primitives operating with whole input block and having both the operation and table transformation characteristics. In the paper we introduce one of the possible ways to implement such primitives that are made up using the *elementary bitwise operations*. The features of these primitives are: (i) property to perform the transformations on whole input block at once, (ii) capability to control the cryptographic characteristics, (iii) and suitability for fast software and cheap hardware implementation.

2 The Types of Cryptographic Primitives

To begin with, let us recall that each iterative block cipher [1] usually consists of a sequence of the relatively simple *round transformations* (*rounds*) and evaluates some encryption function $Y=\mathbf{E}(\boldsymbol{K}^*, X)$, where $\boldsymbol{K}^* \in \mathrm{GF}(2)^N$ is an initial *secret key*, $X \in \mathrm{GF}(2)^n$ and $Y \in \mathrm{GF}(2)^n$ are *input* and *output* binary blocks.

Each round transformation can be presented as the function $X(i+1)=\mathbf{F}(\boldsymbol{K}_i, X(i))$, where $i \in \{0, \ldots, r\text{-}1\}$ is a number of round, $X(i) \in \mathrm{GF}(2)^n$, $X(0)=X$, $X(r)=Y$, and $\boldsymbol{K}_i=\Psi(i, \boldsymbol{K}^*) \in \mathrm{GF}(2)^m$ is an extended *round key* received as a result of the key extension procedure. In this case, the block cipher performs the iterative transformation $Y=\mathbf{F}(\boldsymbol{K}_{r-1}, \mathbf{F}(\boldsymbol{K}_{r-2}, \ldots, \mathbf{F}(\boldsymbol{K}_1, \mathbf{F}(\boldsymbol{K}_0, X))\ldots))$. So in fact, the security of such cipher is determined only by the properties of both the round function $\mathbf{F}$ and the procedure Ψ forming the extended round keys. In this connection, while considering the block ciphers and their primitives we shall use equation $Y=\boldsymbol{F}(\boldsymbol{K}, X)$ as a generalized round transformation with some round key $\boldsymbol{K}$.

To produce effective encryption and decryption, the transforma-

tion $\mathbf{F}$ must be invertible and should have an inverse transformation $X=\mathbf{F}^{-1}(\boldsymbol{K}, Y)$ with low complexity and necessary cryptographic properties. These requirements are reached by special round structure and peculiarity of the basic primitives. Each key-independent primitive, transforming input block X to output block Y, can be presented by a *vector Boolean function* (BF) $Y=\mathbf{G}(X)=(f_0(X), \ldots, f_{n-1}(X))$. In case of key-dependent primitive, such presentation has the form $Y=\mathbf{G}(K, X)=(f_0(K, X), \ldots, f_{n-1}(K, X))))$, where K is an element (*subkey*) of the round key $\boldsymbol{K}$. The functions forming the output block Y we shall call *generating BF* and introduce notation $y_i=f_i(X)$ or $y_i=f_i(K, X)$, where $\forall i \in\{0, 1, \ldots, n\text{-}1\}$ $y_i \in$GF(2). Later on for all BF, without loss of generality, we shall use only *an algebraic normal form* based on two Boolean operations – "$\oplus$" (XOR) and "&" (conjunction).

With respect to these operations the cryptographic primitives can be divided into linear and nonlinear ones, depending on linearity or nonlinearity of the transformations performed. In this connection, we shall call the primitive $Y=\mathbf{G}(X)$ to be *linear* if $\forall X_1, X_2 \in$GF(2)^n $\mathbf{G}(X_1 \oplus X_2)=\mathbf{G}(X_1)\oplus\mathbf{G}(X_2)$. Each component of such primitive is determined by an *affine* BF $y = a \oplus a_0x_0\oplus\ldots\oplus a_{n-1}x_{n-1}$, where $(a, a_0, \ldots, a_{n-1}) \in$GF(2)$^{n+1}$. Linear primitives possess a determinate *propagation (diffusion) property* to change the output bits in response of input bits changing (*error propagation*). As a result, the value of any affine BF always changes its value after an odd number of input arguments is changed, and doesn't change output value otherwise. The particular cases of the linear transformations are permutations, each of which keeps equal the Hamming weight of output and input blocks. Usually the linear transformation is performed by means of table transformation or by binary matrix multiplication.

The nonlinear primitives contain the nonlinear generating BFs and determine complexity of relation between input and output bits (*confusion property*). These primitives play a prime part in security of the block ciphers, concerning the differential and linear cryptoanalysis [1]. Concept of nonlinearity is usually characterized by two primitive's properties: (i) *an algebraic nonlinearity* (maximum degree of the

algebraic normal form (*algebraic degree*) among all generating BFs), and (ii) *a nonlinearity value concerning to the affine BFs* (minimum Hamming distance from the *truth table* of each linear combination of the generating BFs to the truth table of each affine BF). Traditionally the nonlinear primitives are implemented as table transformations or nonlinear algebraic operations realizable with computer commands.

The nonlinear transformation is effective if it has both large nonlinearity and high error propagation. *Bent functions* possess the best properties of such kind [1, 6], having maximum nonlinearity value $2^{l-1} - 2^{(l/2)-1}$, where l is a number of the bent function arguments. So these functions suit well to be used as components of the cryptographic primitives.

To assign the goals of our study let us consider some negative properties of traditional cryptographic primitives used in the iterative block ciphers.

In particular, a sequence of small-sized S-boxes ($\leq$ 8 bits) is not optimal case concerning to cryptographic properties of the aggregate transformation on a whole input block.

Other conventional type of the cryptographic primitives is the algebraic operations that also have essential lacks. In particular, the hardware implementation of such operations requires recursive evaluation of output bits from the least to the most significant bits that results in essential time delaying or increasing implementation cost.

The pointed negative properties of the traditional linear and nonlinear primitives imply the necessity of looking for new alternative ways of the block cipher design.

3 Cryptographic primitives designed on the base of elementary bitwise operations

A promising method of constructing the cryptographic primitives oriented to effective software and hardware implementation is one founded on the relatively simple transformations using only five elementary bitwise operations $\{\oplus, \&, \ll k, \leftarrow k, \rightarrow k\}$. This basis includes the follow-

ing elements:

1. Two operations on two vector operands $X_1, X_2 \in \mathrm{GF}(2)^n$: (i) an addition $(X_1 \oplus X_2)$ and (ii) a multiplication $(X_1 \& X_2 = X_1 X_2)$ modulo 2.

2. Three operations on one vector operand $X \in \mathrm{GF}(2)^n$: (i) a cyclic rotation (on the given positions k) to the left $(X^{\ll k})$; (ii) a logic shift to the left $(X^{\leftarrow k})$, and (iii) a logic shift to the right $(X^{\rightarrow k})$ (during the logic shift the released positions are filled by zeroes).

The peculiarities of the bitwise operations are (i) broad availability and efficiency of their implementation on any computer platforms, (ii) low hardware complexity, (iii) and high speed of transformation performed on all bits of input blocks at once.

The idea to use similar operations as elementary units in cryptographic design has been introduced earlier by S. Wolfram [2] and successfully used in the block ciphers 3-Way [1], BaseKing, and also in the cryptochip Subterranean [4]. These algorithms contain elementary *cellular automata* (CA) [2] formed on the basis of three bitwise operations$\{\oplus, \&, \ll k\}$.

For further study of our method let us introduce some notation concerning CAs. In a wide sense, CAs are known as effective tools generating pseudo-random sequences used in different scientific areas. The feature of CA $Y = \mathbf{G}(X)$ over the field GF(2) during one step of activity consists in a mapping of each input *cell* (bit) of n-dimensional binary input block $X \in \mathrm{GF}(2)^n$ to a corresponding cell of output block $Y \in \mathrm{GF}(2)^n$, depending on states of input cell and some *neighboring* input cells. By our notation, the cells in output block are determined with the generating BFs $\forall i \in 0, 1, \ldots, n-1$ $y_i = f_i(X^{(l_1,\ i,\ l_2)})$, where $X^{(l_1,\ i,\ l_2)} = (x_{i-l_1}, x_{i-l_1+1}, \ldots, x_i, \ldots, x_{i+l_2-1}, x_{i+l_2})$ and l_1, l_2 are left and right radiuses of CA $(0 \le l_1 + l_2 \le n-1)$ that may be dependent on i $(l_1 = l_1(i),\ l_1 = l_2(i))$.

In this design, each generating BF f_i includes the different arguments $X^{(l_1,\ i,\ l_2)}$ also depending on i. However it is more convenient if the structure of this function should be considered irrespective of i with regard to some arguments belonging to an unified vector $Z = (z_0, \ldots, z_p) \in \mathrm{GF}(2)^p$, where $p \le l_1 + l_2$. Such BF $y = f_i(Z)$

we shall call a *prototype* of the generating BF $y_i = f_i(X^{(l_1,\ i,\ l_2)})$. A correspondence of the prototype arguments to the arguments of the generating BF is called *a transition mapping* σ_i: $Z \to X^{(l_1,\ i,\ l_2)}$. For example, the prototype corresponding to the generating BFs $y_i = x_{i-3} \oplus x_{i-1}x_{i-3} \oplus x_i \oplus x_{i+2} \oplus x_{i+1}x_{i+2}$ can be represented by both the simple BF $y = z_0 \oplus z_0z_3 \oplus z_1 \oplus z_2 \oplus z_2z_4$ and the transition mapping $\sigma_i : (z_0,\ z_1,\ z_2,\ z_3,\ z_4) \to (x_{i-3},\ x_i,\ x_{i+2},\ x_{i-1},\ x_{i+1})$.

Since the single prototype can correspond to several generating BFs, then a number of prototypes in CA can be varied from 1 up to n. Thus, any CA is determined by a set of the generating BFs $\{f_i(X^{(l_1,\ i,\ l_2)})\}$, but at the same time a complexity of CA and its cryptographic properties are convenient to be considered regarding a set of prototypes and transition mappings $\{\{f_i(Z)\}, \{\sigma_i\}\}$.

Each generating BF is said to perform a *local mapping* of neighborhood cells to the proper output cell. The aggregated transformation of input block X to output block Y is known to be a *global mapping* of CA. The properties of the global mapping in many respects depend on conditions of the local mapping at the boundaries of input block. In this connection, there are two types of CA: (i) *a cyclic CA* (CCA), (ii) and *CA with initial boundary conditions* (ICA). To the best of our knowledge, nowadays only CCAs have been used in cryptography. In such CCAs the arguments of the generating BFs are cyclically rotated on the boundary of input block, i. e. arithmetic operations on indices are executed modulo n: $\forall i \in \{0,\ 1,\ \ldots,\ n\text{-}1\}$ $X^{(l_1,\ i,\ l_2)} = (x_{(i-l_1)\ mod\ n},\ \ldots,\ x_i,\ \ldots,\ x_{(i+l_2)\ mod\ n})$.

As for ICA, all its neighboring cells, overstepping the boundary of input block, get the proper values from a fixed binary block $X^{(0)}$ of *initial conditions*. It means that only customary arithmetic operations on indices are executed. Therefore in case $i + l > n$, missing arguments of the ICA generating BFs f_i have to be sequentially substituted by the given components of the block $X^{(0)}$ (for more details see section 5).

If l_1=0 or l_2=0, then such CA will be *one-sided* and we shall call it *a right* or *a left CA*, accordingly. This CA is performed by the generating BFs $y_i = f_i(x_i,\ \ldots,\ x_{i+l}) = f_i(X^{(i,\ l)})$ or $y_i = f_i(x_{i-l},\ \ldots,\ x_i) = f_i(X^{(l,\ i)})$.

The CA may contain not only one input block X (*one-dimensional CA*) but also several input blocks $X_1, X_2, \ldots, X_k \in \mathrm{GF}(2)^n$ (*multi-dimensional CA*). The CA of such kind corresponds to a transformation $Y=\mathbf{G}(X_1, X_2, \ldots, X_k)$ performed by the generating BFs $\forall i \in \{0, 1, \ldots, n\text{-}1\}\ y_i=f_i(X_1^{(l_{1,1},\ i,\ l_{1,2})}, X_2^{(l_{2,1},\ i,\ l_{2,2})}, \ldots, X_k^{(l_{k,1},\ i,\ l_{k,2})})$.

According to the general definition of the primitive nonlinearity, the CA is called linear if all its prototypes $f_i(Z)$ are affine BFs, and nonlinear otherwise.

If the prototypes $f_i(Z)$ or transition mappings σ_i are not identical, then CA is called *hybrid* or *nonuniform*. It seems that the greatest concern in applying to the block cipher is introduced by *uniform* CA having both identical prototypes ($f_i(Z)=f(Z)$) and identical transition mappings ($\sigma_i=\sigma$).

The trivial example of *uniform* CCA with radius k can be presented by the cyclic rotation of input block on k positions to the left: $Y = X^{\ll k}$. Such CCA has the elementary generating BFs $y_i=x_{(i+k)\ mod\ n}$ corresponding to the prototype $y=z_0$, and the transition mapping is set by a substitution $z_0 \to x_{(i+k)\ mod\ n}$.

It is clear that nonuniform CA likely to be more sophisticated cryptographic primitive than uniform CA. However the last one, being performed in the base of elementary bitwise operations, has the important capability to be faster due to the transformation of whole input block at once. At the same time, the principal problem of such primitives is connected with their bijectivity playing a prime part in the block ciphers design concerning invertibility and other cryptographic properties of the round transformation $\mathbf{F}$.

4 Linear primitives on the basis of cellular automata

For the right (or left) uniform linear CCAs, having the generating BFs $\forall i \in \{0, 1, \ldots, n\text{-}1\}\ y_i=a_0x_i \oplus a_1x_{i+1} \oplus \ldots \oplus a_{n-1}x_{(i+n-1)\ mod\ n}$, the necessary and sufficient conditions for their invertibility have been obtained and a general method of designing invertible CCAs have been

considered [5]. These conditions are determined by using apparatus of *CA characteristic polynomials* and *CA state generating functions* in accordance with the technique described below [5].

For the mentioned CCA let us consider a characteristic polynomial over the field GF(2) $A(z)=a_0 \oplus a_1 z \oplus \ldots \oplus a_{n-1} z^{n-1}$, and let us set a state generating function $X(z)=x_0 \oplus x_1 z \oplus \ldots \oplus x_{n-1} z^{n-1}$ for input block $X = (x_0, x_1, \ldots, x_{n-1})$. Then for output block $Y = (y_0, y_1, \ldots, y_{n-1})$ we can obtain the following state generating function $Y(z)=y_0 \oplus y_1 z \oplus \ldots \oplus y_{n-1} z^{n-1} = A(z)X(z) \bmod (1+z^n)$. The *necessary and sufficient* condition for such CCA to be invertible is that $\mathrm{LCD}(\mathrm{A}(z), (1+z^n))=1$. Thus, the characteristic polynomial $\mathrm{B}(z)=b_0 \oplus b_1 z \oplus \ldots \oplus b_{n-1} z^{n-1}$ of inverse CCA is evaluated from the expanded Euclidean algorithm by the formula $\mathrm{A}(z)\mathrm{B}(z)+(1+z^n)\mathrm{C}(z)=1$ that is equivalent to the equation $\mathrm{A}(z)\mathrm{B}(z)=1 \bmod (1+z^n)$.

The *necessary* condition for invertibility of uniform linear CCA over the field GF(2) is that the characteristic polynomial $\mathrm{A}(z)$ contains an odd number of terms [5]. Indeed, since for any n the "one" is a root of the equation $(1+z^n)=0$, then we always obtain its factorization in the form $(1+z^n)=(1+z)\mathrm{D}(z)$, where $\mathrm{D}(z)$ is a polynomial of a degree n-1. Further, if $\mathrm{A}(z)$ contains an even number of terms, then $\mathrm{A}(1)=0$. This implies equation $\mathrm{A}(z)=(1+z)\mathrm{A}'(z)$. Therefore we have $\mathrm{LCD}(\mathrm{A}(z), (1+z^n)) = \mathrm{LCD}((1+z)\mathrm{A}'(z), (1+z)\mathrm{D}(z)) \neq 1$, i. e. the condition of CCA invertibility is failed. Hence the oddness of a number of terms in $\mathrm{A}(z)$ is the necessary condition for the right (or left) uniform linear CCA to be invertible.

Until recently the uniform linear and nonlinear CCAs were applied only as the invertible primitives integrated in a classic *substitutional-permutation network* (SP-network). The feature of these networks is that for their invertibility each linear and nonlinear component primitive included in the network should be invertible. As a rule, the SP-network is usually constructed on the base of the primitives having the identical dimensions of input blocks. However nowadays design of the invertible uniform nonlinear CCAs is essentially limited by both the possible size of input blocks (in particular case $n \neq 2^k$) and the radius l of the generating BFs ($l \leq 3$). These limitations substantially decrease

cryptographic properties of CCAs global mapping.

According to such SP-networks structure based on CCAs, not only elementary nonlinear, but also the simple linear CCAs with non-typical block-sizes ($n \neq 2^k$) have been used in the known SP-networks. For example, the cryptochip Subterranean has the block-size n=257 and based on both the invertible uniform nonlinear CCA with the generating BFs $y_i = f(X^{(i,\ 2)}) = x_i \oplus x_{i+2} \oplus x_{i+1}x_{i+2} \oplus 1$ and invertible uniform linear CCA with the characteristic polynomial $\mathrm{A}(z) = 1 \oplus z^3 \oplus z^8$.

At the same time, to perform the most effective linear transformation the uniform linear CCA should meet the following conditions:

1. The block-size of CCA input block should be equal to a dimension of the block transformed that usually equals to a power of 2 ($n = 2^k$).

2. The mutually inverse characteristic polynomials $\mathrm{A}(z)$ and $\mathrm{B}(z)$ should have the same amount of terms, which one should equal approximately to $n/2$.

According to the following statement, in case $n = 2^k$ each second characteristic polynomial $\mathrm{A}(z)$ is invertible that provides the broad capabilities to find linear transformations satisfying to the requirements pointed.

Theorem 1. *If dimension of the characteristic polynomial of the uniform linear CCA over the field* GF(2) *is* $n = 2^k$,*then the oddness of a number of terms in this polynomial is not only a necessary, but also a sufficient condition of CCA invertibility.*

Proof. The necessary and sufficient condition for the uniform linear CCA to be invertible is that $\mathrm{LCD}(\mathrm{A}(z), (1+z^n)) = 1$, where $\mathrm{A}(z)$ is the CCA characteristic polynomial. As it was mentioned earlier, for invertibility of such CCA the corresponding characteristic polynomial $\mathrm{A}(z)$ must include an odd number of terms. So factorization of $\mathrm{A}(z)$ on the simple polynomials doesn't contain the factor $(1+z)$.

While $n = 2^k$, the equation $(1+z^n) = (1+z)^n$ always holds for polynomials over the field GF(2). Indeed, the polynomial $(1+z)^{2^k}$ is a product of 2^k elementary factors $(1+z)$. If these factors would be aggregate by pairs, a result product would be in the form of 2^{k-1} polynomials $(1+z)^2 = (1+2z+z^2) = (1+z^2)$. On the second step we shall receive prod-

uct in the form of 2^{k-2} polynomials $(1+z^2)^2=(1+2z^2+z^4)=(1+z^4)$ due to aggregating the factors $(1+z^2)$ by pairs. Having conducted of k such steps, we obtain the initial polynomial $(1+z^{2^k})$. Since a factorization of the characteristic polynomial with an odd number of terms $\mathrm{A}_{odd}(z)$ does not contain the factor $(1+z)$, then using previous result we obtain that the equality $\mathrm{LCD}(\mathrm{A}_{odd}(z),\ (1+z^{2^k}))=\mathrm{LCD}(\mathrm{A}_{odd}(z),\ (1+z)^{2^k})=1$ always holds. This completes the proof.

In accordance with definition of the uniform linear CCA, the matrix of the corresponding linear transformation is *circulant*. In this matrix each subsequent string from the initial one, being defined as a *generating string*, is cyclically moved on one position to the right (or to the left). Hence from the theorem 1 it follows:

Corollary 1. *Any circulant matrix of the size* $2^k \times 2^k$ *over the field* GF(2) *is invertible if and only if its generating string contains an odd number of ones.*

As three examples of the mutually inverse uniform linear CCAs for n=32, we illustrate the CCAs determined by the following pairs of characteristic polynomials:

1. $\mathrm{A}^{(1)}(z) = 1 \oplus z^2 \oplus z^3 \oplus z^6 \oplus z^7 \oplus z^{10} \oplus z^{12} \oplus z^{16} \oplus z^{17} \oplus z^{18} \oplus$
$\oplus z^{20} \oplus z^{22} \oplus z^{24} \oplus z^{25} \oplus z^{26} \oplus z^{28} \oplus z^{31},$
$\mathrm{B}^{(1)}(z) = 1 \oplus z \oplus z^4 \oplus z^5 \oplus z^6 \oplus z^7 \oplus z^{12} \oplus z^{13} \oplus z^{18} \oplus z^{19} \oplus z^{21} \oplus$
$\oplus z^{22} \oplus z^{24} \oplus z^{25} \oplus z^{27} \oplus z^{30} \oplus z^{31};$
2. $\mathrm{A}^{(2)}(z) = 1 \oplus z^2 \oplus z^3 \oplus z^6 \oplus z^7 \oplus z^{10} \oplus z^{12} \oplus z^{16} \oplus z^{17} \oplus z^{18} \oplus$
$\oplus z^{20} \oplus z^{22} \oplus z^{24} \oplus z^{25} \oplus z^{26} \oplus z^{28} \oplus z^{30},$
$\mathrm{B}^{(2)}(z) = 1 \oplus z \oplus z^2 \oplus z^5 \oplus z^7 \oplus z^9 \oplus z^{10} \oplus z^{11} \oplus z^{14} \oplus z^{15} \oplus z^{18} \oplus$
$\oplus z^{19} \oplus z^{21} \oplus z^{23} \oplus z^{27} \oplus z^{28} \oplus z^{30};$
3. $\mathrm{A}^{(3)}(z) = 1 \oplus z^2 \oplus z^3 \oplus z^4 \oplus z^6 \oplus z^8 \oplus z^9 \oplus$
$\oplus z^{10} \oplus z^{14} \oplus z^{17} \oplus z^{18} \oplus z^{19} \oplus z^{20} \oplus z^{24} \oplus z^{28},$
$\mathrm{B}^{(3)}(z) = z^2 \oplus z^9 \oplus z^{10} \oplus z^{14} \oplus z^{15} \oplus z^{16} \oplus z^{18} \oplus$
$\oplus z^{20} \oplus z^{21} \oplus z^{22} \oplus z^{25} \oplus z^{26} \oplus z^{29} \oplus z^{30} \oplus z^{31}.$

Owing to the structure of the uniform linear CCAs these primitives may easily be performed on the basis of two bitwise operations $\{\oplus, \ll k\}$. Each generating string of the circulant matrices (32×32) corresponding to the characteristic polynomials $\mathrm{A}^{(1)}(z)$, $\mathrm{B}^{(1)}(z)$, $\mathrm{A}^{(2)}(z)$, and $\mathrm{B}^{(2)}(z)$ contains 17 ones, and each generating string corresponding to characteristic polynomials $\mathrm{A}^{(3)}(z)$ and $\mathrm{B}^{(3)}(z)$ contains 15 ones. For example, mutually inverse CCAs for polynomials $\mathrm{A}^{(1)}(z)$ and $\mathrm{B}^{(1)}(z)$ have the following representations:

$$\begin{aligned} Y = \mathbf{G}(X) = X \oplus X^{\ll 2} \oplus X^{\ll 3} \oplus X^{\ll 6} \oplus X^{\ll 7} \oplus X^{\ll 10} \oplus X^{\ll 12} \oplus \\ \oplus X^{\ll 16} \oplus \oplus X^{\ll 17} \oplus X^{\ll 18} \oplus X^{\ll 20} \oplus X^{\ll 22} \oplus X^{\ll 24} \oplus X^{\ll 25} \oplus \\ \oplus X^{\ll 26} \oplus X^{\ll 28} \oplus X^{\ll 31}, \\ Y = \mathbf{G}^{-1}(X) = X \oplus X^{\ll 1} \oplus X^{\ll 4} \oplus X^{\ll 5} \oplus X^{\ll 6} \oplus X^{\ll 7} \oplus X^{\ll 12} \oplus \\ \oplus X^{\ll 13} \oplus X^{\ll 18} \oplus X^{\ll 19} \oplus X^{\ll 21} \oplus X^{\ll 22} \oplus X^{\ll 24} \oplus X^{\ll 25} \oplus \\ \oplus X^{\ll 27} \oplus X^{\ll 30} \oplus X^{\ll 31}. \end{aligned}$$

Note that the mentioned earlier cyclic rotation $Y{=}\mathbf{G}(X){=}X^{\ll k}$, being a trivial example of uniform linear CCA, has the characteristic polynomial $\mathrm{A}(z){=}z^k$. From the equation $z^k\mathrm{B}(z){=}1 \bmod (1{+}z^n)$ it follows that $\mathrm{B}(z){=}z^{n-k}$. Therefore the inverse CCA looks like $Y{=}\mathbf{G}^{-1}(X){=}X^{\ll n-k}$.

The main feature of the uniform linear CCAs is the suitability for fast software and cheap hardware implementation. In hardware based on schematic cells, each of which has m inputs, such CCAs are performed with a small critical path defined by formula $t_{\mathrm{G}} \approx (\lfloor \log_m L \rfloor + 1)t_{\oplus}$, where $\lfloor\ \rfloor$ means the integer part of number, L is the number of items ($X^{\ll}$) in the linear transformation, and $t_{\oplus}$ is the critical path of elementary bitwise operation "$\oplus$". Such CA has identical critical path of the direct and inverse transformation. In the mentioned first and second examples, we have $t_{\mathrm{G}}{=}5t_{\oplus}$, and in the third case $t_{\mathrm{G}}{=}4t_{\oplus}$. For $m{=}4$ the critical path of these CCAs decrease up to $3t_{\oplus}$ and $2t_{\oplus}$, accordingly.

5 Cellular operations as cryptographic primitives

As we noted earlier, the invertibility of the nonlinear CAs is an enough sophisticated problem concerning the CAs usage in the block ciphers. The main difficulties consist (i) in obtaining such transformation to be one-to-one (bijective) or (ii) in estimating reduction of this primitive output domain. This problem has been studied for elementary nonlinear CAs by using the apparatus of *generating functions* [2, 4]. As a result, the conditions of bijectivity for some elementary uniform CAs, having the limitations on both radius (l) and input block-size (n), were obtained. But still there exists an opened problem of such transformations to be bijective for practically significant case $n{=}2^k$. To overcome this difficulty we offer to use cellular operations (CO) as new cryptographic primitives developed on the base of ICA.

The title of these primitives has been formed by analogy with customary algebraic operations having one or two binary operands. Such operations can be presented by the formula $Y{=}\mathbf{S}(X_1, X_2)$, where $X_1, X_2, Y \in \mathrm{GF}(2)^n$. Each subsequent value of output bit in nonlinear algebraic operations of such type depends upon corresponding input bit and on all previous bits of input block in accordance with the formula $\forall i \in\{0, 1, \ldots, n\text{-}1\}$ $y_i{=}f_i(x_0^{(1)}, \ldots, x_i^{(1)}, x_0^{(2)}, \ldots, x_i^{(2)})$. Besides, in detail such function is represented in a recursive form providing difficulties in hardware implementation. In particular, for addition module 2^n, which one may be considered as CO starting point [7], each i-th output bit is recursively determined by the generating BF $y_i{=}x_i^{(1)}{\oplus}x_i^{(2)}{\oplus}u_{i-1}$, where u_{i-1} ($u_{-1}{=}0$) is a *carry bit* formed in accordance with recurrence $u_{i-1}{=}\, x_{i-1}^{(1)}(x_{i-1}^{(2)}{\oplus}u_{i-2}){\oplus}x_{i-1}^{(2)}u_{i-2}$. By solving this equation, we obtain the following probability: $P(u_i = 1) = (1/2)(1 - (1/2^{i-1}))$. Therefore the carry bit's probabilistic dependency on previous input bits decrease exponentially. Thus, from the probabilistic viewpoint each i-th generating BF essentially depends not on a set of all arguments $\{x_{i-j}\}$, where $0 \leq j \leq i$, but only on the restricted amount of the neighboring about the left arguments. This property doesn't yield optimal cryptographic

effect of the resulting transformation in spite of large time delay while such operation is being performed in low-cost hardware.

In this connection, the main advantage of CO is the property to get rid of recursiveness in calculations. So we design the COs in such manner that the generating BFs and a number of their arguments may be selected practically arbitrarily. This COs peculiarity allows to verify cryptographic properties of the transformation as a whole. We form such one-dimensional CO by the following rules:

1. The CO corresponds to the right (or left) ICA. In particular, for the right CO each i-th cell of output block Y is determined by the generating BFs $\forall i \in\{0, \ldots, n\text{-}1\}$ $y_i{=}f_i(X^{(i,\ l)}){=}$ $x_i{\oplus}\varphi_i(x_{i+1}, \ldots, x_{i+l}){=}x_i{\oplus}\varphi_i(X^{(i+1,\ l-1)})$, where each φ_i (*a basic BF*) is a variable part of the generating BF.

2. While $i + l > n$, according to the definition of ICA the elements of any given but fixed initial condition $X^{(0)}{=}(0, \ldots, 0, x_1^{(0)}, x_2^{(0)}, \ldots, x_l^{(0)}) \in \mathrm{GF}(2)^n$ are consecutively used as a missing part of φ_i arguments.

So in fact, input block X is additionally extended on l fixed bits and is formed by aggregating X and $X^{(0)}$: $X \cup X^{(0)} = (x_0, \ldots, x_{n-1}, x_1^{(0)}, \ldots, x_l^{(0)}) \in \mathrm{GF}(2)^{n+l}$.

By this notation, CO performs the transformation $Y{=}\mathbf{G}(X \cup X^{(0)}) \in \mathrm{GF}(2)^n$ having the following generating BFs $\forall i \in\{0, 1, \ldots, n\text{-}1\}$ $y_i{=}f_i((X \cup X^{(0)})^{(i,\ l)}) = x_i \oplus \varphi_i((X \cup X^{(0)})^{(i+1,\ l-1)})$.

Due to CO design it is clear that each output bit of the proposed primitives depend on some neighboring input bits only. Besides, these primitives possess the extra property of generating BFs to become more complicate while moving from the edge ($i{=}n-1$) to the middle of the transformed block ($i{=}n-l$) within the bounds of radius (l). This CO's attribute as against nonlinear uniform CCA brings about a low nonlinearity of some initial generating BFs. Remarkably that such property coincides with similar property of nonlinear algebraic operations. In our case, we meet a tradeoff between average level of primitive nonlinearity and its output domain reduction. Indeed, as against CCA our CO slightly loses in average nonlinearity, but obtains important cryptographic property in accordance with the following statement.

Theorem 2. *Every one-dimensional linear and nonlinear CO* $Y = \mathbf{G}(X \cup X^{(0)}) \in \mathrm{GF}(2)^n$ *for any* n *and for arbitrary basic BFs* φ_i *always performs a bijective transformation* $X \to Y (X, Y \in \mathrm{GF}(2)^n)$.

The proof is trivial and follows from the uniqueness of the solution of a set of equations from x_{n-1} up to x_0 by Gaussian reduction:

$$
\begin{aligned}
x_{n-1} &= y_{n-1} \oplus \varphi_{n-1}\left((X \cup X^{(0)})^{(n,\ l-1)}\right) = \\
&= y_{n-1} \oplus \varphi_{n-1}(x_1^{(0)},\ x_2^{(0)},\ \ldots,\ x_l^{(0)}), \\
x_{n-2} &= y_{n-2} \oplus \varphi_{n-2}\left((X \cup X^{(0)})^{(n-1,\ l-1)}\right) = \\
&= y_{n-2} \oplus \varphi_{n-2}(x_{n-1},\ x_1^{(0)},\ \ldots,\ x_{l-1}^{(0)}), \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
x_1 &= y_1 \oplus \varphi_1\left((X \cup X^{(0)})^{(2,\ l-1)}\right) = y_1 \oplus \varphi_1(x_2,\ x_3,\ \ldots,\ x_l), \\
x_0 &= y_0 \oplus \varphi_0\left((X \cup X^{(0)})^{(1,\ l-1)}\right) = y_0 \oplus \varphi_0(x_1,\ x_2,\ \ldots,\ x_{l-1}).
\end{aligned}
$$

To transform several input blocks at once or to combine the elements of the round key $\boldsymbol{K}$ with input blocks, we propose to make use a multi-dimensional CO with the global mapping $Y{=}\mathbf{G}(X_1 \cup X_1^{(0)},\ X_2,\ \ldots,\ X_k)$ formed by the local mappings with the generating BFs $y_i{=}f_i((X_1 \cup X_1^{(0)})^{(i,\ l_1)},\ X_2,\ \ldots,\ X_k){=}x_i \oplus \varphi_i((X_1 \cup X_1^{(0)})^{(i+1,\ l_1-1)},\ X_2,\ \ldots,\ X_k)$, where $Y \in \mathrm{GF}(2)^{n_1}$ and $\forall j \in \{1,\ \ldots,\ k\}$ $X_j{=}(x_0^{(j)},\ \ldots,\ x_{n_j-1}^{(j)}) \in \mathrm{GF}(2)^{n_j}$.

It is remarkable that notation of a multi-dimensional CO may be considered as some extension of a multi-dimensional ICA concept. Indeed, as against ICA each input block $\forall j \in 2, \ldots, k$ X_j of proposed CO is presented by a binary vector of any given dimension n_j, so that not only neighboring but any components of this vector can be used as arguments of the basic BFs φ_i.

For multi-dimensional CO we shall prove the following cryptographically important statement:

Theorem 3. *Every multi-dimensional CO* $Y{=}\mathbf{G}((X_1 \cup X_1^{(0)}),\ X_2,\ \ldots,\ X_k)$ $(\forall j \in \{1,\ \ldots,\ k\}$ $X_j \in \mathrm{GF}(2)^{n_j}$,

$Y \in \mathrm{GF}(2)^{n_1}$) *formed by the generating BFs* $\forall i \in \{1, \ldots, n_1\text{-}1\}$ $y_i = x_i \oplus \varphi_i((X_1 \cup X_1^{(0)})^{(i+1,\, l_1-1)}, X_2, \ldots, X_k)$ *and by fixed initial conditions* $X_1^{(0)}$, *always performs* ***regular*** *mapping*: $(X_1, X_2, \ldots, X_k) \to Y(\mathrm{GF}(2)^{n_1+n_2+\ldots+n_k} \to \mathrm{GF}(2)^{n_1})$. *It means that, while aggregated input block* $(X_1, X_2, \ldots, X_k)$ *passing all* $2^{n_1+n_2+\ldots+n_k}$ *values in domain* $\mathrm{GF}(2)^{n_1+n_2+\ldots+n_k}$, *output block* Y *gets all* 2^{n_1} *different values strictly* $2^{n_2+\ldots+n_k}$ *times in domain* $\mathrm{GF}(2)^{n_1}$.

Proof. The necessary and sufficient condition for any *surjective* mapping $\mathrm{GF}(2)^n \to \mathrm{GF}(2)^m$ $(n > m)$ to be *regular* is a balanceness of each BF having been obtained by an arbitrary *linear combination* of the generating BFs [3]. A concept of the balanced function means that *truth table* of each such linear combination should contain identical number of zeroes and ones. In this case, BF gets the values 0 or 1 with the probability 1/2.

Further, for an arbitrary vector $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{n-1}) \in \mathrm{GF}(2)^n$ let us consider linear combination of CO generating BFs:

$$Q_\alpha(X_1, \ldots, X_k) = \sum_{i=0}^{n-1} \alpha_i f_i \left(\left(X_1 \cup X_1^{(0)} \right)^{(i,l_1)} \right) =$$

$$= \sum_{i=0}^{n-1} \alpha_i \varphi_i \left(\left(X_1 \cup X_1^{(0)} \right)^{(i+1,l_1-1)}, X_2, \ldots, X_k \right) \oplus \sum_{i=1}^{n-1} \alpha_i x_i^{(1)}.$$

Suppose k is the minimum value, at which one $\alpha_k = 1$; then $\mathbf{Q}_\alpha = x_k^{(1)} \oplus \mathbf{Q}'_\alpha(x_{k+1}^{(1)}, \ldots, x_{n_1-1}^{(1)}, X_1^{(0)}, X_2, \ldots, X_k)$. This formula corresponds to a superposition of random both the $x_k^{(1)}$ and the data $\mathbf{Q}'_\alpha$. For this reason, a proper result can be obtained by standard evaluation. Since we have the probability $\mathrm{P}(x_k^{(1)}=0) = \mathrm{P}(x_k^{(1)}=1) = 1/2$ and $\mathbf{Q}'_\alpha$ doesn't depend on $x_k^{(1)}$, then we have the following probability:

$$P(\mathbf{Q}_\alpha = 0) = P(\mathbf{Q}'_\alpha = 0/x_k^{(1)} = 0)P(x_k^{(1)} = 0) +$$
$$+ P(\mathbf{Q}'_\alpha = 1/x_k^{(1)} = 1)P(x_k^{(1)} = 1) = (1/2)(P(\mathbf{Q}'_\alpha = 0/x_k^{(1)} = 0) +$$
$$+ P(\mathbf{Q}'_\alpha = 1/x_k^{(1)} = 1)) = (1/2)(P(\mathbf{Q}'_\alpha = 0) + P(\mathbf{Q}'_\alpha = 1)) = 1/2.$$

Thus, each linear combination $\mathbf{Q}_\alpha$ is a balanced BF. This completes the proof.

Corollary 2. Suppose we have any multi-dimensional CO $Y=\mathbf{G}((X_1 \cup X_1^{(0)}), X_2, \ldots, (X_j \cup X_j^{(0)}), \ldots, X_k)$, where $X_j \in \mathrm{GF}(2)^{n_j}$, $n_j=n_1$, and $\forall i \in \{1, \ldots, n_1\text{-}1\}$ $y_i=x_i^{(1)} \oplus x_i^{(j)} \oplus \varphi_i((X_1 \cup X_1^{(0)})^{(i+1,\ l_1-1)}, X_2, \ldots, (X_j \cup X_j^{(0)})^{(i+1,\ l_j-1)}, \ldots, X_k)$; then as a result of theorems 2 and 3 the global mapping $\mathbf{G}$ is bijective separately on X_1 and on X_j. From here it follows the way to design the multi-dimensional CO that one is simultaneously bijective on any input block X_j, where $n_j=n_1$.

It seems that uniform COs using only one prototype of the basic BFs $\varphi_i(Z)=\varphi(Z)$ and only identical transition mappings $\sigma_i=\sigma$ are more suitable for cryptographic applications. These operations possess the principle property of uniform CCA to transform simultaneously all bits of input data considered in a vector form. As we shall illustrate in the further CO example, the feature of such operations is that the cyclic rotation is substituted by adding ($\oplus$) the logically shifted ($\leftarrow k$) input block X to the logically shifted ($\rightarrow k$) block of initial conditions $X^{(0)}$. Thus, uniform CO presented as *a vector algebraic normal form* of the generating BFs can be performed on the basis of four vector operations $\{\oplus, \&, \leftarrow k, \rightarrow k\}$. In hardware such primitives are carried out with critical path $t_{\mathrm{G}} \approx (\lfloor \log_m M \rfloor + 1) t_\oplus$, where M is a number of input variables contained in the given prototype $\varphi(Z)$, and m is a number of inputs in elementary schematic cell [8].

Obviously, the CO is made in such manner that its nonlinearity and error propagation property can be changed by varying prototype $\varphi(Z)$ and/or transition mapping σ.

For example, we shall consider the practical construction of CO $Y=\mathbf{G}(X \cup X^{(0)}) \in \mathrm{GF}(2)^n$ having the following generating BFs: $y_i=f((X \cup X^{(0)})^{(i,\ 7)})= x_i \oplus \varphi((X \cup X^{(0)})^{(i+1,\ 6)}= x_i \oplus x_{i+1}x_{i+3} \oplus x_{i+2}x_{i+5} \oplus x_{i+6}x_{i+7} \oplus x_{i+1}x_{i+2}x_{i+6}$ with initial condition $X^{(0)}=(0, \ldots, 0, x_1^{(0)}, \ldots, x_7^{(0)}) \in \mathrm{GF}(2)^n$. Here the unique prototype $\varphi(Z)$ is determined by the bent function $y=z_0z_3 \oplus z_1z_4 \oplus z_2z_5 \oplus z_0z_1z_2$, and the transition mapping σ is set by the

substitution $(z_0, z_1, z_2, z_3, z_4, z_5) \to (x_{i+1}, x_{i+2}, x_{i+6}, x_{i+3}, x_{i+5}, x_{i+7})$.

This bijective nonlinear primitive has the following representation on the basis of four vector operations$\{\oplus, \&, \leftarrow k, \to k\}$:

$$Y = \mathbf{G}^{(1)}(X) = X \oplus (X^{\leftarrow 1} \oplus X^{(0)\to 6})(X^{\leftarrow 3} \oplus X^{(0)\to 4}) \oplus$$
$$\oplus (X^{\leftarrow 2} \oplus X^{(0)\to 5})(X^{\leftarrow 5} \oplus X^{(0)\to 2}) \oplus (X^{\leftarrow 6} \oplus X^{(0)\to 1})(X^{\leftarrow 7} \oplus X^{(0)}) \oplus$$
$$\oplus (X^{\leftarrow 1} \oplus X^{(0)\to 6})(X^{\leftarrow 2} \oplus X^{(0)\to 5})(X^{\leftarrow 6} \oplus X^{(0)\to 1}).$$

Using initial condition $X^{(0)} = (0, \ \ldots, \ 0, \ 1, \ 1, \ 1, \ 1, \ 1, \ 1, \ 1)$ and modified input block $X'=X \oplus E$, where $E=(1, \ 1, \ \ldots, \ 1)$, it is possible to construct more simple bijective CO on the basis of only three vector operations $\{\oplus, \ \&, \ \leftarrow k\}$:

$$Y = \mathbf{G}^{(2)}(X \oplus E) = (X \oplus E) \oplus (X^{\leftarrow 1} \oplus E)(X^{\leftarrow 3} \oplus E) \oplus (X^{\leftarrow 2} \oplus E)$$
$$(X^{\leftarrow 5} \oplus E) \oplus (X^{\leftarrow 6} \oplus E)(X^{\leftarrow 7} \oplus E) \oplus (X^{\leftarrow 1} \oplus E)(X^{\leftarrow 2} \oplus E)(X^{\leftarrow 6} \oplus E).$$

In case $X^{(0)} = (0, \ldots, 0)$, CO has even more simple implementation on the basis of the same vector operations $\{\oplus, \&, \leftarrow k\}$:

$$Y = \mathbf{G}^{(3)}(X) = X \oplus X^{\leftarrow 1}X^{\leftarrow 3} \oplus X^{\leftarrow 2}X^{\leftarrow 5} \oplus X^{\leftarrow 6}X^{\leftarrow 7} \oplus$$
$$\oplus X^{\leftarrow 1}X^{\leftarrow 2}X^{\leftarrow 6}.$$

6 Iterative block ciphers based on the cellular transformations

The different CAs, including their new class – cellular operations, may be generalized as cellular transformations (CT). These cryptographic primitives can be effectively used as components of the iterative block ciphers and hash functions. While the invertible uniform linear CCAs can be applied as universal elements of any cryptoschemes, the uniform linear and nonlinear COs possess some features imposing limitations on their usage in the block ciphers. It is connected with high complexity of CO's inverse transformation, which one usually corresponds to a nonuniform CO. The given circumstance allows using CO only in the

special block cipher cryptoschemes that don't require invertibility of this primitive for decryption. As an example of such cryptoschemes, let us consider the well-known *Feistel scheme* to be widely applied in the block cipher design.

We recall that round transformation $\mathbf{F}$ formed with *traditional* Feistel scheme [6] has the left and the right transformation branches. For all rounds except the last one this transformation is determined by the following equations: $Y_L = \mathbf{f}(K, X_L) \oplus X_R$ and $Y_R = X_L$, where (i) $\mathbf{f}(\boldsymbol{K}, X_L)$ is a round function, (ii) X_L, X_R $\in$GF$(2)^{n/2}$ are left and right halves (*subblocks*) of input block X $\in$GF$(2)^n$, (iii) and Y_L, Y_R $\in$GF$(2)^{n/2}$ are left and right subblocks of output block Y $\in$GF$(2)^n$. On the last round the transformation $\mathbf{F}$ corresponds to the equations Y_R=$\mathbf{f}(\boldsymbol{K}, X_L) \oplus X_R$ and Y_L=X_L, i. e. as against the main rounds the swapping of subblocks Y_L and Y_R is not fulfilled here. The principle feature of this scheme is that for invertibility of the round transformation $\mathbf{F}$ the round function $\mathbf{f}$ is not required to be invertible. Meanwhile, to provide more secure encryption it is desirable for this function to be bijective [1]. During decryption inverse round transformation $\mathbf{F}^{-1}$ is performed under the same scheme as the direct transformation $\mathbf{F}$. For all decryption procedure it means that (i) subblocks Y_L, Y_R are considered as input subblocks, (ii) and the round keys are used in the reverse order.

As examples of the round functions belonging to the round transformations Y=$\mathbf{F}(\boldsymbol{K}, X)$ of the multi-purpose block cipher Y=$\mathbf{E}(\boldsymbol{K}^*, X)$ based on CTs, we introduce the following functions $\mathbf{f}(\boldsymbol{K}, X_L)$ (see Fig. 1) having convenient software and cheap hardware representation:

1. $\mathbf{f}$=$\mathbf{G}_3\{\mathbf{G}_2[(\mathbf{G}_1((\boldsymbol{K}\oplus X_L) \cup X_1^{(0)}))^{\ll\lambda_1} \cup X_2^{(0)}]\}$,
2. $\mathbf{f}$=$\mathbf{G}_2[(\mathbf{G}_1(\mathbf{G}_3\{\boldsymbol{K}\oplus X_L\}\cup X_1^{(0)}))^{\ll\lambda_1} \cup X_2^{(0)}]$,
3. $\mathbf{f}$=$\mathbf{G}_2[\mathbf{G}_3\{\mathbf{G}_1((\boldsymbol{K}\oplus X_L)^{\ll\lambda_1} \cup X_1^{(0)})\}\cup X_2^{(0)}]$,
4. $\mathbf{f}$=$\mathbf{G}_3\{(\mathbf{G}_1((\boldsymbol{K}\oplus X_L)^{\ll\lambda_1} \cup X_1^{(0)})\}$,
5. $\mathbf{f}$=$\mathbf{G}_1((\mathbf{G}_3\{\boldsymbol{K}\oplus X_L\})^{\ll\lambda_1} \cup X_1^{(0)})$, where

- $\mathbf{G}_1$ is the right, and $\mathbf{G}_2$ is the left uniform nonlinear CO; $\forall i$=1, 2 each $\mathbf{G}_i$ has both the unique prototype of the basic BFs $\varphi^{(i)}(Z)$ and the unique transition mapping $\sigma^{(i)}$ (see section 5);

- $X_1^{(0)}$, $X_2^{(0)}$ are the blocks of initial conditions, which ones may be the functions of both the round key $\boldsymbol{K}$ and the round number j(see section 5);

- $\mathbf{G}_3$ is the uniform linear CCA formed by the proper characteristic polynomial $\mathrm{A}_1(z)$ (see section 4);

- $\lambda_1=\lambda_1(j,\, K)$ is an elementary integer function dependent on both the round number $0\leq j\, \leq r-1$ $(0\leq\lambda_1\leq(n/2)-1)$ and elements K of the round key $\boldsymbol{K}$;

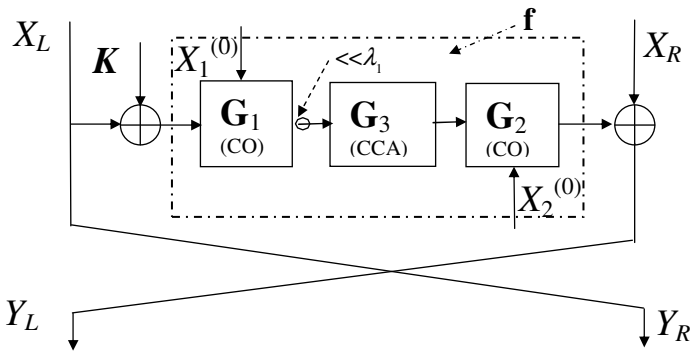

Figure 1. Example of traditional Feistel scheme using the cellular transformations only

To improve the cryptographic properties and to optimize the hardware implementation of the traditional block cipher, we shall consider a *modified* Feistel scheme. For this reason, we make the round transformation to be more complicated, but coordinated by critical path of the round primitives. In particular, such transformation $\mathbf{F}$ on all rounds except the last one can be defined by two equations: $Y_L=\mathbf{f}_3(K_3,\, (\mathbf{f}_1(K_1,\, X_L)\, \oplus\, \mathbf{f}_2(K_2,\, X_R,\, X_L)),\, X_L)$ and $Y_R=X_L$, where (i) K_1, K_2, K_3 are the round subkeys being the different components of the round key $\boldsymbol{K}$, (ii) $\mathbf{f}_1$ is a transformation corresponding to the round function $\mathbf{f}$ of the traditional Feistel scheme, (iii) $\mathbf{f}_2$, $\mathbf{f}_3$ are transformations convertible on X_R and $X'_R=\mathbf{f}_1\oplus\mathbf{f}_2$, respectively. In this scheme both the function $\mathbf{f}_2$ and $\mathbf{f}_3$ can be considered as the transformations of subblocks X_R and X'_R dependent on input subblock X_L or as the

transformations controlled by subblock X_L.

Inverse round transformation $\mathbf{F}^{-1}$ of the modified Feistel scheme has the following form: X_R=$\mathbf{f}_2^{-1}(K_2,\ (\mathbf{f}_1(K_1,\ Y_R)\oplus\mathbf{f}_3^{-1}(K_3,\ Y_L,\ Y_R))$ and $X_L\ = Y_R$. Similarly to the traditional Feistel scheme, the transposition of the left and the right subblocks on the last round of both the direct and the inverse transformations is not applied here.

For the block ciphers constructed with the given modified Feistel scheme the following example of the primitives consisting in the j-th round transformations can be offered (see Fig. 2):

$\mathbf{f}_1$=$\mathbf{f}(K_1 \oplus X_L)$, where $\mathbf{f}$ is a function corresponding to the round function of the traditional Feistel scheme;

$\mathbf{f}_2$=$\mathbf{G}'_2\{(K_2 \oplus X_R)^{\lll\lambda_2}\}$;

$\mathbf{f}_3$=$\mathbf{G}'_3\{K_3,\ \mathbf{f}(K_1 \oplus X_L)\ \oplus\ \mathbf{f}_2(K_2 \oplus X_R,\ X_L)\}^{\lll\lambda_3}$, where

- $\mathbf{G}'_2$ and $\mathbf{G}'_3$ are invertible uniform linear CCAs determined by the characteristic polynomials $\mathrm{A}_2(z)$ and $\mathrm{A}_3(z)$ (for inverse functions $\mathbf{f}_2^{-1}$ and $\mathbf{f}_3^{-1}$ the uniform linear CCAs $\mathbf{G}_2'^{-1}$ and $\mathbf{G}_2'^{-1}$ are determined by the corresponding characteristic polynomials $\mathrm{B}_2(z)\ \ \mathrm{B}_3(z)$);

- λ_2=$\lambda_2(j,\ X_L,\ K_3)$ and λ_3=$\lambda_3(j,\ X_L,\ K_2)$ are elementary integer functions, where $\forall m$=2, 3 $0 \le \lambda_m \le (n/2)-1$ (for inverse $\mathbf{f}_2^{-1}$ and $\mathbf{f}_3^{-1}$ we have λ'_2=$(n/2)-\lambda_3(j,\ Y_L,\ K_2)$ and λ'_3=$(n/2)-\lambda_2(j,\ Y_L,\ K_3)$).

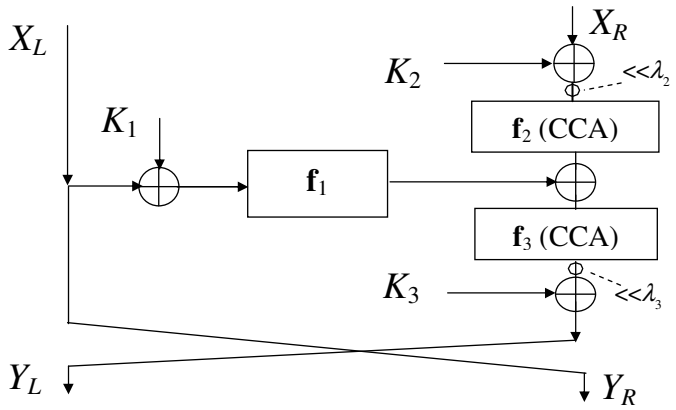

Figure 2. Example of modified Feistel scheme using the cellular transformations only

As a result, the cryptographic properties and flexibility of the proposed block ciphers depend on a set of the following parameters: $\{\varphi^{(1)}(Z),\ \varphi^{(2)}(Z),\ \sigma^{(1)},\ \sigma^{(2)},\ \mathrm{A}_1(z),\ \mathrm{A}_2(z),\ \mathrm{A}_3(z),\ \lambda_1,\ \lambda_2,\ \lambda_3,\ X_1^{(0)},\ X_2^{(0)},\ r\}$. The structure of these parameters and the methods for their optimization (in accordance with a cryptographic security) are out of the scope of our paper and should be the matter of further theoretical and experimental study.

Besides of the parameters pointed, a quality of the block cipher directly depends on the key extension procedure $\boldsymbol{K}_j=\Psi(j,\ \boldsymbol{K}^*)$ forming round keys on the base of the secret key $\boldsymbol{K}^*$ $(0\leq j\ \leq r-1)$. Such procedures are intended to provide the appropriate statistic properties of round keys irrespectively of the secret key. It yields the necessary foundation (i) to ensure the ability for security estimations of the block cipher, (ii) and to complicate searching the cipher's weak keys.

The practice shows that the key extension procedure is rationally to be formed as iterative one using elements of the round transformation. In this case, input data of the current iteration may be obtained by usage of the special constants, the elements of the secret key, and the previously defined round keys. Such approach provides high profitability for software and hardware implementations of the block ciphers.

As two examples of such procedure for the mentioned cipher (see Fig. 2), it is possible to offer a way of the round keys $\boldsymbol{K}_j=(K_1^{(j)},\ K_2^{(j)},\ K_3^{(j)})$ to be obtained on the base of the secret key $\boldsymbol{K}^*=(\boldsymbol{Q}_1,\ \boldsymbol{Q}_2,\ \boldsymbol{Q}_3)\in\mathrm{GF}(2)^{3n/2}$. These keys can be formed by the following recurrences containing elements of round functions $\mathbf{f}$ introduced before:

$$
\begin{aligned}
1.\quad K_1^{(j+1)} &= \mathbf{G}_3\{\mathbf{G}_2[((\mathbf{G}_1(K_3^{(j)}\cup X_1^{(0)}))^{\ll\lambda(j)}\oplus K_2^{(j)})\cup X_2^{(0)}]\},\\
K_2^{(j+1)} &= \mathbf{G}_3\{\mathbf{G}_2[((\mathbf{G}_1(K_1^{(j)}\cup X_1^{(0)}))^{\ll\lambda(j)}\oplus K_3^{(j)})\cup X_2^{(0)}]\},\\
K_3^{(j+1)} &= \mathbf{G}_3\{\mathbf{G}_2[((\mathbf{G}_1(K_2^{(j)}\cup X_1^{(0)}))^{\ll\lambda(j)}\oplus K_1^{(j)})\cup X_2^{(0)}]\};
\end{aligned}
$$

$$
\begin{array}{lrcl}
2. & K_1^{(j+1)} & = & \mathbf{G}_3\{(\mathbf{G}_2(K_2^{(j)} \cup X_2^{(0)}))^{\ll\lambda(j)} \oplus K_3^{(j)}\}, \\
 & K_2^{(j+1)} & = & \mathbf{G}_3\{(\mathbf{G}_2(K_3^{(j)} \cup X_2^{(0)}))^{\ll\lambda(j)} \oplus K_1^{(j)}\}, \\
 & K_3^{(j+1)} & = & \mathbf{G}_3\{(\mathbf{G}_2(K_1^{(j)} \cup X_2^{(0)}))^{\ll\lambda(j)} \oplus K_2^{(j)}\},
\end{array}
$$

where $K_1^{(0)}=(\mathbf{G}_2((\boldsymbol{Q}_1 \oplus R_1) \cup X_1^{(0)}))^{\ll(n/2)-1}$, $K_2^{(0)}=(\mathbf{G}_2((\boldsymbol{Q}_2 \oplus R_2) \cup X_1^{(0)}))^{\ll(n/2)-1}$, $K_3^{(0)}=(\mathbf{G}_2((\boldsymbol{Q}_3 \oplus R_3) \cup X_1^{(0)}))^{\ll(n/2)-1}$ and R_1, R_2, $R_3 \in \mathrm{GF}(2)^{n/2}$ are the binary constants having the uniform probability distribution of zeros and ones. For example, such constants can be selected as a sequence of n bits contained in a binary representation of the well-known irrational numbers π, e, etc.

7 Conclusion

The method of the block ciphers and other cryptographic algorithms design introduced on the base of CTs only may be considered as one of the promising directions in constructing the effective cryptographic systems. The necessary properties of such systems can be reached by combining the different CCAs and COs as new cryptographic primitives.

In some cases, the CTs can be applied instead of table linear transformations, S-blocks, and algebraic operations conventionally used in cryptography. The limitations on the COs usage are connected with the difficulty to perform inverse CO, which one have usually to be the nonuniform CO. At the same time, this feature of COs can play a positive part in designing secure hash functions.

The proposed principle schemes of the block ciphers based on CTs only and defined by a set of parameters may be considered as a baseline for further study and elaboration of the fast cryptographic algorithms having necessary cryptographic properties and suitable for universal implementation.

References

[1] B. Schneier, *Applied Cryptography Second Edition: protocols, algorithms, and source code in C. John Wiley and Sons.* Inc. New York, 1996, 758 pp.

[2] S. Wolfram. *Cryptography with Cellular Automata* // Advances in Cryptology - Crypto'85. Proceedings. Springer Verlag. 1985. pp. 429–432.

[3] J. Seberry, X-M. Zhang, Y. Zheng. *Nonlinearly Balanced BFs and Their Propogation Chracteristics* // Advances in Cryptology - Crypto'93 Proceedings. LNCS. Springer Verlag. 1994. pp. 49–60.

[4] L. Claesen, J. Daemen, M. Genoe, G. Peeters, Subterranean: *A 600 Mbit/sec Cryptographic VLSI Chip*, Proc. ICCD'93: VLSI in Computers and Processors, IEEE Computer Society Press. 1993. pp. 610–613.

[5] J. Daemen, R. Govaerts and J. Vandewalle. *A New Approach Towards Block Ciphers Design* // Advances in Cryptology - FSE'94 Proceedings. LNCS. Springer Verlag. 1994. pp. 18–33.

[6] H. Feistel. *Cryptography and Computer privacy* // Scientific American. 1973. Vol.228. N.5. pp. 15–23.

[7] B.V. Izotov, A.A. Moldovyan, N.A. Moldovyan. *Controlled Operations as a Cryptographic Primitive* // MMM-ACNS'2001 Proc. LNCS. Springer-Verlag. 2001. Vol. 2052. pp. 230–241.

[8] N. Sklavos, A.A. Moldovyan, O. Koufopavlou. *Encryption and Data Dependent Permutations: Implementation Coast and Performance Evaluation* // MMM-ANCS'2003 Proc. LNCS. Springer-Verlag. 2003. Vol. 2776. pp. 337–348.

Received September 3, 2003

B.V. Izotov,

Specialized Center of Program Systems "SPECTR",
Kantemirovskaya str., 10, St. Petersburg 197342, Russia;
ph./fax.7-812-2453743,
E–mail: *ibv@cobra.ru*