Methods for Design of Balanced Boolean Functions Satisfying Strict Avalanche Criterion (SAC)

Dr. N.G. BARDIS
Adjunct Assistant Professor
Department of Automation
Technological Education Institute of Halkis
34400 Psahna, Halkis, Evia, Greece

Dr.M. MITROULI
Department of Mathematics
National and Kapodistrian University of Athens
Panepistimiopolis, GR 15784 Athens, Greece

Dr. A.P. MARKOVSKYY
Department of Computer Engineering
National Technical University of Ukraine
37, Peremohy, pr. Kiev 252056, KPI 2003,
Ukraine

Dr. A. POLYMENOPOULOS
Principal Scientist, ADB Consulting, LLC
Greece

Abstract: - In this paper two methods of designing Balanced Boolean functions for cryptographical transformations are presented. The first of them is based on using orthogonal nonlinear components. The second method realizes a combinatorial approach. Both methods provide high nonlinearity for the obtained functions and both of them operating with Algebraic Normal Form. The advantage of the first method is the simplicity and the technological realization and of the second the significant greatest number of the generated functions compared to the known methods.

Key-Words: - Boolean functions, balanced functions, SAC-functions

1 Introduction

Every cryptographic transformation is based on an unsolved mathematical task. The impossibility to analytically solve system of nonlinear Boolean equations is considered to be one of these mathematical tasks. This is precisely the reason that the Boolean functions are used in cryptography. Based on the nonlinear Boolean transformations they have developed practically all block cipher, the stream cipher and the most part of the hash-algorithms. In a practical level the Boolean functions are not being used only in public key algorithms that have as mathematical base an unsolved task of the number theory.

The main advantage of the Boolean functions in cryptography is considered to be the high efficiency at their calculation by software and hardware means. So for algorithms with roughly the same cryptoresistance, that is based on the Boolean functions is executed thousands times faster than modern public key algorithms.

Attacking to the cryptographic algorithms, based on Boolean transformations can be considered as an effort of solving an equivalent nonlinear Boolean functions system.

The classical types of attacks to block cipher are linear and differential cryptanalysis [2,6]. Their utilization allows under certain condition to decrease enumeration while algorithm breaking.

The effectiveness of those types of attacks is determined definitely by the properties of the Boolean functions that are used in cryptographic algorithms. In order to provide higher cryptoresistance to the linear and differential cryptanalysis, the Boolean functions which are used in cryptographical transformations should be balanced, orthogonal, have high non-linearity and maximum differential entropy.

The high non linearity of the Boolean functional transformations makes the linear approximation non effective at linear cryptanalysis.

The high level of differential entropy of the Boolean transformations provides smoothing of XOR-profile which decreases the effectiveness of differential cryptoanalysis.

From the cryptographycal applications point of view, one of the most important properties of Boolean function is the Strict Avalanche Criterion – SAC. For the Boolean transformations that have this criterion, as shown above, the effectiveness of the differential cryptanalysis is decrease. Apart from this, SAC makes the elimination of the variables not effective at linear cryptanalysis [6].

The dynamic increase of information technology requires on one side the increase of cryptoresistance, due to the complexity of cryptographic transformations and on the other side the more rapid cryptographic processing.

The solving of these conflicting requirements can be achieved using in the cryptographycal structures which are reconfigurated by a key Boolean functions with high cryptographic characteristics of great number of variables.

Under these conditions it's very important from the practical point of view, to develop high technology methods of designing Boolean functions that satisfy cryptographic criteria and more specific reconfigurated Balanced high non linear SAC functions.

2 Basic Definitions and Properties of SAC-functions

The Boolean function f(x_1,…,x_n) of n variables can be represented in the form of true table or binary sequences of length 2^n and in algebraic normal form (ANF):

$$f(X) = a_0^0 \oplus \bigoplus_{j=1}^{n} a_j^1 \cdot x_j \oplus \bigoplus_{\substack{i=1\\ k=i+1}}^{i<n,k<=n} a_{ik}^2 \cdot x_i \cdot x_k \oplus \cdots a_{1\ldots n}^n \cdot x_1 \cdot \ldots \cdot x_n \quad (1)$$

The Hamming weight W (f(x_1,…,x_n) of a Boolean function f(x_1,…,x_n) of n variables is the total number of the values of "one" that the function attains on the 2^n possible tuples of the variables values that form the set Z

$$W(f(x_1,...,x_n)) = \sum_{x_1,...,x_n \in Z} f(x_1,...,x_n) \quad (2)$$

The Boolean function f(x_1,…,x_n) satisfies the total entropy maximum criterion, i.e., is balanced if it takes the values of "zero" and "one" with equal probability:

$$W(f(x_1,...,x_n)) = 2^{n-1} \quad (3)$$

The Boolean function f(x_1,…,x_n) satisfies the criterion of the conditional entropy maximum or Strict Avalanche Criterion (SAC), if alterating any of its n variables results in changing the value of the function with the probability of 0.5.

$$\forall x_j, j = 1,..., n : W(f(x_1,..., x_j,...,x_n) \oplus \oplus f(x_1,..., \overline{x}_j,..., x_n)) = 2^{n-1} \quad (4)$$

In this case the non-linearity, N (f(x_1,…,x_n)), of the Boolean function f(x_1,…,x_n) is determined as the minimal Hamming's distance to the linear functions:

$$N(f(x_1,...x_n)) = = \min_{a_k \in \{0,1\}, k=0,...,n} W(f(x_1,...x_n) \oplus (a_0 \oplus ... \bigoplus_{j=1,...,n} a_j \cdot x_j)) \quad (5)$$

The theoretical maximum of nonlinearity of Boolean functions of n variables is equal to:

$$N(f_b(x_1,...,x_n)) = 2^{n-1} - 2^{n/2-1} \quad (6)$$

Boolean functions that have a maximum nonlinearity (6) are called bent-functions. But bent-functions are not balanced and so they can not be immediately used in cryptographical transformations.

For balanced Boolean functions f(x_1,…,x_n) of n variables the value of nonlinearity N(f) with the constraint n>3 has limit superior [3]:

$$N(f) \le 2^{n-1} - 2^{n/2-1} - 2 \qquad \text{for even n}$$

$$N(f) \le \left\lfloor 2^{n-1} - 2^{(n-1)/2-1} \right\rfloor \qquad \text{for odd n}$$

where $\lfloor x \rfloor$ is the maximal integer which is less than or equal to x.

Order of nonlinearity of Boolean functions f(x_1,…,x_n) is maximum terms length in its ANF representation.

Taking any of the variables x_j, j∈{1,…,n} the Boolean function f(x_1,…,x_n) can always be represented as f(x_1,…,x_n)=α_j(x_1,…,x_{j-1},x_{j+1},…,x_n)⊕x_j·β_j(x_1,…,x_{j-1},x_{j+1},…,x_n), where α_j(x_1,…,x_{j-1},x_{j+1},…,x_n) and β_j(x_1,…,x_{j-1},x_{j+1},…,x_n) – Boolean functions, independent from variable x_j. If function β_j is balanced, then function f(x_1,…,x_n) satisfies SAC for the variable x_j., since f(x_1,…,x_j,…,x_n)⊕f(x_1,…,x_j⊕1,…,x_n)=β_j(x_1,…x_{j-1},x_{j+1},…,x_n). That way, the condition (4) about the correspondence of the function f(x_1,…,x_n) SAC is equivalent to:

$$\forall x_j \in \{x_1,..., x_n\} : W(\beta_j(x_1,..., x_{j-1}, x_{j+1},..., x_n)) = 2^{n-1} \quad (8)$$

Particularities of balanced Boolean SAC-functions manifest themselves in specific properties of their spectrum [8].

To obtain the spectrum F(w_1,…,w_n) of the Boolean function f(x_1,…,x_n), the direct Walsh transform should be performed according to the following formula:

$$F(\overline{W}) = \sum_{\overline{X}\in Z} f(\overline{X}) \cdot (-1)^{\overline{X}\cdot\overline{W}} \qquad (9)$$

The inverse Walsh transform, that obtains the Boolean function $f(x_1,\ldots,x_n)$ by its spectrum $F(w_1,\ldots,w_n)$, is achieved by the formula:

$$f(\overline{X}) = 2^{-n} \cdot \sum_{\overline{W}\in Z} F(\overline{W}) \cdot (-1)^{\overline{X}\cdot\overline{W}} \qquad (10)$$

The Boolean function $f(x_1,\ldots,x_n)$ correspondence to the SAC-criterion may be determined by its spectrum properties $F(w_1,\ldots,w_n)$: a function $f(x_1,\ldots,x_n)$ is a SAC if and only if its spectrum $F(w_1,\ldots,w_n)$ holds the condition:

$$\sum_{\overline{W}\in Z} (-1)^{\overline{W}} \cdot F^2(\overline{W}) = 2^n \cdot F(0,\ldots,0) - 2^{2\cdot n-2} \quad (11)$$

Note that the meaning of the spectrum $F(0,\ldots,0)$ equals the number of ones in the truth table of the function $f(x_1,\ldots,x_n)$, i.e. the function $f(x_1,\ldots,x_n)$ may be called balanced if its spectrum on the zero tuple $F(0,\ldots,0)$ is equal to 2^{n-1} . Taking this into account, a Boolean function $f(x_1,\ldots,x_n)$ is balanced and corresponds to SAC if its spectrum $F(w_1,\ldots,w_n)$ holds the condition:

$$\sum_{\overline{W}\in Z} (-1)^{\overline{W}} \cdot F^2(\overline{W}) = 2^{2n-1} \qquad (12)$$

Thus, if a Boolean function $f(x_1,\ldots,x_n)$ corresponds to the total and conditioned entropy maximum criteria, i.e. it is a balanced SAC-function, then the sum of its spectrum $F(w_1,\ldots,w_n)$ squares, which may be considered as analogous to the energy spectrum, has the maximal value [4].

3 Contemporary State of Balanced SAC functions Design

Due to the importance placed on the automated design of balanced SAC-functions for modern information processing, a number of approaches have been suggested during the last decade. The designing methods of balanced SAC-functions can be divided in to three groups.

In the first group, there are the genetic methods. These methods are based in designing new balanced SAC-functions from functions that are somehow previously obtained. More specifically, the Forre method [4], which is using the spectrum transformations, allows obtaining from one balanced SAC-function a certain set of such functions with the same number of variables. Analysis of formula (12) reveals, that it is possible, in principle, to construct all the spectra $F(w_1,\ldots,w_n)$ for which condition (12) is held. All the balanced SAC-functions may be obtained through the inverse Walsh transform of each constructed spectrum $F(w_1,\ldots,w_n)$ using (10). However, it should be stated [4] that a real Boolean function $f(x_1,\ldots,x_2)$ does not correspond to each spectrum $F(w_1,\ldots,w_n)$ that satisfies condition (12). In order to decrease the number of non-productive inverse transforms (10) and to assure high degrees of nonlinearity, it was suggested in [4] to somehow find a balanced SAC-function $f(x_1,\ldots,x_n)$ and to obtain its spectrum $F(w_1,\ldots,w_n)$ by Walsh transform. It was further suggested that the family of the spectra $F_1(w),\ldots,F_h(w)$, $h<2^n$, for which (12) is held and for which the real balanced SAC-functions corresponds, should be obtained through alteration of the signs of the components $F(w_1,\ldots,w_n)$ in an arbitrary way on all the 2^n tuples $w_1,\ldots,w_n$. The real balanced SAC-functions may be obtained by inverse Walsh transform.

From a processing aspect, the Forre method [4] does not correspond to the requirements imposed above for the design of balanced SAC-functions. This, because it operates with a function's truth tables and the spectra's value tables of whose capacity is proportional to 2^n. The inversion of the Walsh transform to expression (10) demands intensive computer time that is also proportional to 2^n.

Balanced SAC-functions of high nonlinearity may be obtained by de-concatenation of a bent-function [3], however obtaining the bent-functions themselves from a large number of variables is a rather difficult problem whose solution requires substantial computational and memory resources.

There are also methods that allow obtaining from one SAC-function, SAC-function with a higher number of variables. Particularly, if the Boolean function $f(x_1,\ldots,x_n)$ with n variables is balanced and satisfies SAC, then the Boolean function with n+1 variables - $\psi(x_1,\ldots,x_{n+1})=f(x_1,\ldots,x_j\oplus x_{n+1},\ldots,x_n)$, $j\in\{1,\ldots,n\}$ also will be balanced and SAC.

In the second group are the designing methods of SAC-functions that are based on different transformations with orthogonal systems of Boolean functions. Here there is a number of interesting approaches proposed. The design method of balanced SAC functions that has been suggested by Kurosawa K. and Satoh T., is one of the well known [5]. In essence, the method's idea consists of dividing n variables into two non-overlapping sets with s and t variables (n=s+t). Further on, a linear function $g(x_1,\ldots,x_s)$ of s variables and a binary matrix Q dimensionality equal to s × t are formed. In so doing, the number of one-components of the product $Q\cdot\gamma_1$ of matrix Q by any s-component vector γ_1 with one non-zero component and the product $\gamma_2 \cdot Q$ of any t-component vector γ_2 with one non-zero component is more than or equals one. The vector formed by the coefficients of the function $g(x_1,\ldots,x_s)$ is to be linear-independent of the vectors formed by the columns of matrix Q. A balanced SAC-function is formed according to the formula:

$$f(x_1,...,x_n)=[x_1,...,x_s]\cdot Q\cdot$$
$$\cdot[x_{s+1},...,x_n]^T \oplus g(x_1,...,x_s) \qquad (13)$$

The disadvantage of this method is the technological complexity, which is related to the use of matrixes of high dimensions. From the cryptography point of view the main disadvantage of the method the fact that it is based on superposition of linear functions. This allows through the transition to the representation of the function via the correspondent orthogonal basis to simplify the linear cryptanalysis. The advantage of the design methods of Boolean functions belonging to special classes based on the orthogonal systems is that they allow the transition in a more simple way from the design of a single function to the design of a system of high cryptoresistant Boolean functions.

The greatest disadvantage of the genetic methods and the methods based on orthogonal systems is the fact that they only allow to obtain a rather small amount of the total balanced SAC-functions. The main reason for this is that only a small number of balanced SAC-functions can be presented as a superposition of an orthogonal system functions.

In the third group are the combinatorial methods of designing balanced SAC-functions. This group of methods has the greater number of generated functions and is the least studied.

4 Method of designing Balanced SAC functions based on orthogonal transformations

In the basis of the suggested method of designing Balanced SAC functions the properties of orthogonal transformations are laid. But on the contrary to the known methods [5] it is suggested to use as basis nonlinear secondary orthogonal functions.

The substance of the suggested method is presented in the following consecutive steps:

1. Form an orthogonal linear system of the Boolean functions $\lambda_1,\lambda_2,\ldots,\lambda_n$ so that $\lambda_1 = x_1$, $\lambda_2 = x_2 \oplus x_3 \oplus \ldots \oplus x_n$.

2. Form three secondary nonlinear orthogonal Boolean functions φ_1 ,φ_2 and φ_3 in the following way:

$$\begin{aligned}\varphi_1 &= \lambda_3 \oplus \lambda_1 \cdot \lambda_2 \\ \varphi_2 &= \lambda_4 \oplus \phi(\lambda_6,...,\lambda_n) \\ \varphi_3 &= \lambda_5 \oplus \xi(\lambda_6,...,\lambda_n)\end{aligned} \qquad (14)$$

where $\phi(\lambda_6,\ldots,\lambda_n)$ and $\xi(\lambda_6,\ldots,\lambda_n)$ – arbitrary Boolean functions from the linear functions $\lambda_6,\ldots,\lambda_n$,

3. The Balanced SAC function $f(x_1,\ldots,x_n)$ is formed as:

$$f(x_1,...,x_n)=\varphi_1 \oplus \varphi_2 \cdot \varphi_3 \qquad (15)$$

Since this function $f(x_1,\ldots,x_n)$ is XORed of the Balanced function φ_1 and the conjunction of φ_2 and φ_3 that are orthogonal to φ_1, is Balanced.

We will show that the function $f(x_1,\ldots,x_n)$ satisfies SAC. For the variable x_1 the function $f(x1,x2,\ldots xn)\oplus f(x1\oplus 1,x2,\ldots,xn) = \lambda_2$. Since λ_2- is balanced, then the function $f(x_1,\ldots,x_n)$ satisfies SAC for the variable x_1. We examine the variables $x_j \neq x_1$. The functions φ_1, φ_2 and φ_3 can always be presented as:

$\varphi_1=\alpha_j\oplus(x_1\oplus\delta_j)\cdot x_j(\delta_j=1,\ x_j\ \lambda_3)$,

$\varphi_2=\beta_j(x_2,\ldots,x_{j-1},x_{j+1},\ldots,x_n)\oplus\mu_j(x_2,\ldots,x_{j-1},x_{j+1},\ldots,x_n)\cdot x_j$ and

$\varphi_3=\varepsilon_j(x_2,\ldots,x_{j-1},x_{j+1},\ldots,x_n)\oplus\rho_j(x_2,\ldots,x_{j-1},x_{j+1},\ldots,x_n)\cdot x_j$,

where β_j, μ_j , ε_j, ρ_j - are Boolean functions, independent of the variables x_1 and x_j.

Correspondently, the resulting Boolean function $f(x_1,\ldots,x_n)$ can be presented as:

$f(x_1,\ldots,x_n)=(\alpha_j\oplus\beta_j\cdot\varepsilon_j)\oplus(x_1\oplus\delta_j\oplus\mu_j\cdot\varepsilon_j\oplus\beta_j\cdot\rho_j\oplus\mu_j\cdot\rho_j)\cdot x_j$ thus

$f(x_1,\ldots,x_j,\ldots x_n)\oplus f(x_1,\ldots,x_j\oplus 1,\ldots,x_n)=x_1\oplus\delta_j\oplus\mu_j\cdot\varepsilon_j\oplus\beta_j\cdot\rho_j\oplus\mu_j\cdot\rho_j$.

This function is balanced since it is the XOR of the balanced function x_1 and the function $(\delta_j\oplus\mu_j\cdot\varepsilon_j\oplus\beta_j\cdot\rho_j\oplus\mu_j\cdot\rho_j)$, that is independent from x_1.

That way the function $f(x_1,\ldots,x_n)$ satisfies SAC and for all the others variables $x_j \neq x_1$, $j=1,\ldots,n$.

The suggested method with the use of nonlinear functions is suitable for designing functions with more than 4 variables.

An example of designing a balanced SAC function of 8 variables (n=8) is shown below.

Primary system of orthogonal linear functions:

$\lambda_1 = x_1$, $\lambda_2 = x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8$,
$\lambda_3 = x_2$, $\lambda_4 = x_3$, $\lambda_5 = x_4$,
$\lambda_6 = x_5$, $\lambda_7 = x_{5,}$ $\lambda_8 = x_7$

Secondary system of thee orthogonal nonlinear functions:

$\varphi_1=\lambda_3\oplus\lambda_1\cdot\lambda_2=x_2\oplus x_1\cdot(x_2\oplus x_3\oplus x_4\oplus x_5\oplus x_5\oplus x_6\oplus x_7\oplus x_8)$,
$\varphi_2=\lambda_4\oplus\phi(\lambda_6,\lambda_7,\lambda_8)=x_3\oplus\phi(x_5,x_6,x_7)=x_3\oplus x_5\cdot x_6\cdot x_7$,
$\varphi_3=\lambda_5\oplus\xi(\lambda_6,\lambda_7,\lambda_8)=x_4\oplus\xi(x_5,x_6,x_7)=x_4\oplus x_5\cdot x_7$

Resulting balanced SAC-function:

$f(x_1,\ldots x_8) = \varphi_1 \oplus \varphi_2\cdot\varphi_3 = x_2 \oplus x_1\cdot x_2 \oplus x_1\cdot x_3 \oplus x_1\cdot x_4 \oplus x_1\cdot x_5 \oplus x_1\cdot x_6 \oplus x_1\cdot x_7 \oplus x_1\cdot x_8 \oplus x_3\cdot x_4 \oplus x_3\cdot x_5\cdot x_7 \oplus \oplus x_5\cdot x_6\cdot x_7 \oplus x_4\cdot x_5\cdot x_6\cdot x_7$.

The resulting function is balanced SAC function with nonlinearity equal to 96 and the order of nonlinearity is 4.

5 Combinatorial Method for Balanced SAC-function Design

The idea of the method proposed for obtaining the ANF of a balanced Boolean function $f(x_1,x_2,\ldots,x_n)$ that corresponds to the Strict Avalanche Criterion, consists of performing the following sequential actions:

1. The set of variables $\{x_1,\ldots,x_n\}$ is divided into four subsets that do not overlap $\vartheta=\{x_1,\ldots,x_{t+1}\}$, $\Theta=\{x_{t+2},\ldots,x_{2t+1}\}$, Y and Ω so that unite of them is equal of all variable set, must be arbitrary select the number of variables which compose the set Θ - t. Than number of variables which composed the set ϑ must be- $N(\vartheta)=t+1$, Number of variables of sets Y and Ω may be select arbitrary too, while sets Y and Ω may be empty, in other case number of elements in set Y should be less then number of variables which composed the set Ω[омега]. ($N(Y)<N(\Omega)$).

Resulting Boolean balanced SAC-functions are formed as XOR of four intermediate functions, which should be constructed such manner.

2. In the Algebraic Normal Form the first of them - $\varphi(x_1\ldots,x_{2t+1})$ is formed as:

$$\varphi(x_1,\ldots,x_{2t+1}) == \bigoplus_{j=1,\ldots,t} x_j,\ldots,x_{j+t+1} \oplus x_{t+1} \bigoplus_{k=t+2,\ldots,2t+1} x_k$$

3. The second intermediate Boolean function $\beta(x_1,\ldots,x_{t+1})$ is formed as the XOR of an odd number of arbitrary chosen terms from variables of set ϑ.

4. If sets Y and Ω are not empty ($Y\neq\varnothing$, $\Omega\neq\varnothing$), the intermediate Boolean function γ is formed as the XOR of set Ψ of the products of variables that belong to set Y, by the variables that belong to set Ω. In so doing, each variable of sets Y and Ω is to enter as a multiplier at least into one of the products of set Ψ. If $Y=\varnothing$, then function γ set to 0.

5. The fourth intermediate Boolean function η ($x\in\vartheta$, $x\in\Omega$) is formed as the XOR of set Δ of the product of the variables of first and second sets (ϑ and Ω). In so doing, the variable of set Ω is to enter into an even number of the products of set Δ. The formation of functions γ and η is done is such a way so that each variable of set Ω should enter into the products that compose sets Ψ and Δ.

If set Ω is empty then intermediate functions γ and η set to 0.

6. The ANF of the balanced SAC-function f is formed as XOR of all intermediate functions:

$$f(x_1,\ldots,x_n) = \varphi(x_1,\ldots,x_{2t+1}) \oplus \beta(x_1,\ldots,x_{t+1}) \oplus \oplus \gamma(x_{2t+2},\ldots,x_n) \oplus \eta, \qquad (x\in\vartheta, x\in\Omega) \qquad (17)$$

The properties of the balancedness and SAC of a function formed in accordance with the stated method do not change if the following functions are XORed to it:

- An arbitrary function δ determined on variables of set Y;
- An arbitrary function ρ determined on variables of set Ω;

The suggested method is illustrated by the following example of designing a balanced SAC-function of eight variables (n=8) at t=2.

The number of variables that compose set ϑ is $N(\vartheta)=t+1=3$, i.e. $\vartheta=\{x_1,x_2,x_3\}$, the number of variables that compose the set Θ is $N(\Theta)=t=2$, and set $\Theta=\{x_4,x_5\}$. On the basis that inequality $N(Y)<N(\Omega)$ is to be held, sets Y and Ω are defined as: $Y=\{x_6\}$, $\Omega=\{x_7,x_8\}$:

$\vartheta = \{ x_1, x_2, x_3 \}, \Theta=\{x_4,x_5\}, Y=\{x_6\}, \Omega=\{x_7, x_8\}$

Intermediate functions:

$\varphi = x_1\cdot x_4 \oplus x_2\cdot x_5 \oplus x_3\cdot x_4 \oplus x_3\cdot x_5$

$\beta = x_1 \oplus x_2 \oplus x_1\cdot x_2\cdot x_3$

$\gamma = x_6\cdot x_7 \oplus x_6\cdot x_8$ ***(16)***

$\eta = x_1\cdot x_7 \oplus x_2\cdot x_7$, $\delta = x_6$, $\rho = x_8$

Result function:

$f = \varphi \oplus \beta \oplus \gamma \oplus \eta \oplus \delta \oplus \rho =$
$= x_1\cdot x_4 \oplus x_2\cdot x_5 \oplus x_3\cdot x_4 \oplus x_3\cdot x_5 \oplus x_1 \oplus x_2 \oplus x_1\cdot x_2\cdot x_3 \oplus \oplus x_6\cdot x_7 \oplus x_6\cdot x_8 \oplus x_1\cdot x_7 \oplus x_2\cdot x_7 \oplus x_6 \oplus x_8$

Function f is balanced and satisfies SAC.

Nonlinearity N(f) = 112.

It can be shown that the nonlinearity N(f) of functions $f(x_1,\ldots,x_n)$ designed according to the developed procedure in the following way:

$N(f)=2^{n-1}-2^{n/2}$ - even number of variables n

$N(f)=2^{n-1}-2^{n-(n-1)/2-1}=2^{n-1}-2^{(n+1)/2-1}$ -odd n (18)

For example, at n=8, the nonlinearity of the balanced SAC-functions designed by the suggested method makes $2^7-2^4=112$ (under that maximum is equal 118), and at n=9 $N(f)=2^8-2^4=240$.(that is equal maximum possible for odd number of variables).

Order of nonlinearity of functions, which can be formed by suggested method may be equal or less n-1 that corresponded to theoretic maximum value.

The developed method for synthesis of balanced SAC–functions may be extended to solving the problem of obtaining balanced SAC–functions of order k.

The idea of the method suggested for obtaining ANF balanced Boolean functions $f(x_1,x_2,...,x_n)$, that satisfy the Strict Avalanche Criterion of order k consists of performing the following actions:

1. The set of variables is divided into four subsets that do not overlap $\vartheta=\{x_1,\dots,x_t\}$, $\Theta=\{x_{t+1},\dots,x_{2t+2}\}$, Y and Ω. In this case the number of variables of first set ϑ, which we will call t should not be less than k+2 ($t \geq k+2$) and the number of variables in second set Θ should be 2 more than t: $N(\Theta)=t+2$.

While choosing sets Y and Ω, there are three possibilities:

- the first - is that the both sets are empty,
- the second - is that set Y is empty, and Ω is not empty,
- the third is that sets Y and Ω are not empty;

In the last case the following condition should be obey: $N(\Omega) \geq k+1$. Balanced SAC-function $f(x_1,\dots,x_n)$ of order k is formed as XOR of four intermediate functions.

2.(t+1) linear Boolean functions $\lambda_1(x_{t+1},\dots,x_{2t+2})$, $\lambda_2(x_{t+2},x_{t+4},\dots x_{2t+2})$,…, $\lambda_{t+1}(x_{t+1},x_{t+2},\dots, x_{2t},x_{2t+2})$ from variables of second set Θ are formed so that linear function $\lambda_j(j=1,\dots,t+1)$ number j be a XORed of all the variables of this set except the variable number j :

$$\lambda_j(x_{t+1},...,x_{t+j-1},x_{t+j+1},...,x_{2t+2}) = \bigoplus_{k=1,...,t+2} a_k \cdot x_{t+k}, \qquad (19)$$
$$\forall l \in \{1,...,t+2\}, l \neq j : a_l = 1, a_j = 0$$

3. The first intermediate functions Boolean function $\varphi(x_1,\dots,x_{2t+2})$ is formed as:

$$\varphi(x_1,...,x_{2t+2}) = \lambda_1 \oplus \bigoplus_{j=2,...,t+1} x_{j-1} \cdot \lambda_j \quad (20)$$

4. The ANF of an arbitrary chosen second intermediate Boolean function β that was determined on variables of first set ϑ is formed.

5. If both sets Y and Ω are not empty, then the third intermediate Boolean function γ is formed as the XOR of set Ψ of the products of variables that belong to the set Y and Ω. In so doing, each of variables of set Y is to enter as a multiplier, not less than k+1 times, into products that compose the set Ψ. If set Y is empty, then the third intermediate function γ set to 0.

6. Forth intermediate Balanced function $\eta(x\in\vartheta,x\in\Omega)$ is formed as the XOR of set Δ of the products of variables of first set ϑ, by variables of set Ω. The functions $\gamma(x_{2t+2},\dots,x_n)$ and $\eta(x\in\vartheta,x\in\Omega)$ should be formed in such a way so that each variable of set Ω is to enter not less than (k+1) times into products that compose sets Ψ and Δ. If set Ω is empty, then both intermediate functions γ and η set to 0.

7. Resulting balanced SAC-function f of order k is formed as XOR of all intermediate functions:

$$f(x_1,...,x_n) = \varphi(x_1,...,x_{2\cdot t+1}) \oplus \beta(x_1,...,x_{t+1}) \oplus$$
$$\oplus \gamma(x_{2\cdot t+2},...,x_n) \oplus \eta \qquad (x \in \vartheta, x \in \Omega) \qquad (21)$$

The suggested method is illustrated by the following example for designing a balanced SAC-function of the first order (k=1) from 9 variables (n=9).

Reasoning that number t of variables that compose first set ϑ should be greater or equal k+2, so suppose that t equal 3. Correspondently first set include variables from one to tree.

The number of variables that compose second subset Θ is $N(\Theta)=t+2=5$, correspondently the subset Θ itself is $\{x_4, x_5, x_6, x_7, x_8\}$. Let subset Y is empty, and subset Ω include one variable x_9.

In accordance with the presented procedure, linear functions from variables of second set Θ formed as:

$\lambda_1=x_5 \oplus x_6 \oplus x_7 \oplus x_8$, $\lambda_2=x_4 \oplus x_6 \oplus x_7 \oplus x_8$,
$\lambda_3=x_4 \oplus x_5 \oplus x_7 \oplus x_8$, $\lambda_4=x_4 \oplus x_5 \oplus x_6 \oplus x_8$.

Correspondently first intermediate function $\varphi(x_1,\dots,x_8)$ is formed as:

$\varphi(x_1,\dots,x_8)=\lambda_1\oplus\lambda_2\cdot x_1\oplus\lambda_3\cdot x_2\oplus\lambda_4\cdot x_3$.

Arbitrary choosing second intermediate function $\beta(x_1,x_2,x_3)= x_1\cdot x_2\cdot x_3$. Third intermediate function $\gamma=0$ because of set Y is empty. Forth intermediate function $\eta(x_1,x_2,x_3,x_9)$ is equal $x_1\cdot x_9 \oplus x_2\cdot x_9$. The resulting balanced Boolean SAC-function $f(x_1,\dots,x_9)$ is formed as:

$f(x_1,\dots,x_9)= x_5 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_1\cdot x_4 \oplus x_1\cdot x_6 \oplus x_1\cdot x_7 \oplus x_1\cdot x_8 \oplus x_2\cdot x_4 \oplus x_2\cdot x_5 \oplus x_2\cdot x_7\oplus x_2\cdot x_8\oplus \oplus x_3\cdot x_4 \oplus x_3\cdot x_5 \oplus x_3\cdot x_6 \oplus x_3\cdot x_8 \oplus x_1\cdot x_2\cdot x_3 \oplus x_1\cdot x_9 \oplus x_2\cdot x_9$.

The function that is formed is balanced and corresponds to SAC. With fixation of any one variable $x_1,\dots,x_9$ the formed function $f(x_1,\dots,x_9)$ is transformed into balanced SAC-functions from 8 variables. In particular, it is not difficult to make sure that if $x_1=0$ function f is transformed to function $g(x_2,\dots,x_9) = x_5 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_2\cdot x_4 \oplus x_2\cdot x_5 \oplus x_2\cdot x_7 \oplus x_2\cdot x_8 \oplus x_3\cdot x_4 \oplus x_3\cdot x_5 \oplus x_3\cdot x_6 \oplus x_3\cdot x_8 \oplus x_2\cdot x_9$ which is balanced SAC-function.

Compared to the known methods for obtaining cryptographically strong functions, the suggested one requires much less computational resources. More specific compared to one of the most effective methods suggested by Kurosawa – show that the presented one provides performance by about two orders higher.

6 Conclusion

Two methods of designing balanced Boolean SAC functions are suggested. The first one is based on the orthogonal transformations. In the contrary of the known methods [5], the suggested method relies on the superposition of non linear function that correspondently makes linear cryptanalysis difficult through the transition to a different coordinates system. The second method develops the idea of the combinatorial approach to design balanced SAC functions.

The suggested methods for the designing of balanced Boolean SAC-functions operates only with ANF and removes the processing limitation for obtaining functions from a large number of variables (experiments carried out proved the ability to generate balanced SAC-functions from hundreds of variables). Also, the methods makes it possible to obtain a function in the most appropriate form - the method suggested does not require the resource-intensive stage of Boolean function minimization. Compared to other known methods for the design of balanced Boolean functions the suggested is much more feasible in practice and needs much lower resources for its implementation in computers because it does not require the time-consuming operations of Walsh spectrum transforms or the search for linear independent vectors or bent–functions. Experimental studies proved the performance of the second method to be nearly 3 orders higher compared to [5] for n=128.

The significant advantage of the second one of the proposed methods compared to the known ones is that it allows the generation of an appreciably larger number of balanced SAC-functions from all the possible ones at a given number of n variables. For example, for n=4, the suggested method may design approximately 200 functions, while the method [5] only 72 functions.

References:

[1] Bardis E.G., Bardis N.G., Markovskyy A.P., Spyropoulos A.K. Design of Boolean Function from a Great Number of Variables Satisfying Strict Avalanche Criterion.// Recent Advances in Signal Processing and Communication.WSESP,1999. P.107-112.

[2] Biham E., Shamir A. Differential cryptoanalysis of DES-like cryptosystems // Advances in Cryptology – Crypto'90 Proceeding, Lecture Notes in Computer Sciences, 537– 1990- P.2-21.

[3] Cusic T.W. On construction of balanced correlation immune function, in sequences and their application. // Proceeding of SETA'98- Springer Discrete Mathematics and Theoretical Computer Sciences,-1999-P.184-190.

[4] Forre R. The strict avalanche criterion: spectral properties of Boolean functions and extend definition // Advances in Cryptology – Crypto'88 Proceeding, Lecture Notes in Computer Sciences, 403 – 1988- P.450-468.

[5] Kurosawa K., Satoh T. Design of SAC/PC(l) of Order k Boolean Functions and Three Other Cryptographic Criteria.// Advances in Cryptology –Eurocrypt'97 Proceeding, Lecture Notes in Computer Science 1233-1997-P.433-449.

[6] Matsui M. Linear Cryptanalysis Method for DES Cipher // Advances in Cryptology- Eurocrypt'93 Proceedings, Lecture Notes in Computer Science 765, P. 386-397. Spriger-Varlag,1994.

[7] Polymenopoulos A., Bardis E.G., Bardis N.G., Markovskaja N.A., "Design and Implementation of Boolean Balanced Functions Satisfying Strict Avalanche Criterion (SAC)", WSES – "On Problems in Applied Mathematics and Computational Intelligence", ISBN: 960-8052-30-0, 2001, pp. 12-16.

[8] Webster A.F., Tavares S.E. On the design of S-boxed. // Advances in Cryptology – Crypto'85, Proceeding, Notes in Computer Science, 332 – 1986 P.523-535.