A New Statistical Testing for Symmetric Ciphers and Hash Functions

Eric Filiol

INRIA, Projet CODES,
B.P. 105, 78153 Le Chesnay Cédex, FRANCE
Eric.Filiol@inria.fr

Abstract. This paper presents a new statistical testing of symmetric ciphers and hash functions which allow us to detect biases in a few of these systems. We first give a complete characterization of the Algebraic Normal Form (ANF) of random Boolean functions by means of the Möbius transform. Output bits of a cryptosystem are here described by a set of Boolean functions. The new testing is based on the comparison between their Algebraic Normal Form and those of purely random Boolean functions. Detailed testing results on several cryptosystems are presented. As a main result we show that AES, DES, Snow, and Lili-128 fail the tests wholly or partly and thus present strong biases.

Keywords: Boolean function, statistical testing, symmetric cipher, randomness, hash function, Möbius transform, Walsh Transform.

1 Introduction

Randomness is the ground property of cryptography. For the attacker, any quantities produced by a given cryptosystem must look as unpredictable as possible. It means that these quantities have to be of sufficient size and "be random" in the sense that the probability of any particular value being selected must be as weak as possible to preclude a cryptanalyst from gaining advantage through optimed search strategy based on such probability [15, p 169].

From a general point of view, any symmetric cipher and any hash function must be designed as a pseudorandom bit generator (PRBG) relatively to each of its output bits.

Two important requirements are then to be satisfied: the output sequences of a PRBG must be statistically indistinguishable from truly random sequences and the output bits must be unpredictable to an attacker with limited computing facilities. Therefore, many different statistical tests have been proposed and are usually implemented to evaluate these two requirements. Historically, we must cite Golomb's randomness postulates [11]. These tests have been designed as necessary but not sufficient tests to check if a shift register sequence statistically behaves properly. Yet statistically good according to these postulates, this kind of sequence has been shown very predictable when using the Berlekamp-Massey algorithm [16]. This is the illustration that randomness is uniquely defined relatively to the statistical tests we may use.

Many other statistical tests have been proposed in order to improve what may be considered as "random". Among many others, let us cite those that are mainly implemented: frequency test, serial test, poker test, runs test and autocorrelation test [8, 13], Maurer's universal statistical test [17], (for a more detailed bibliography on statistical tests used in cryptography see [15, pp 188-189]).

All the recently proposed symmetric cryptosystems and hash functions can be considered as satisfying all the known randomness requirements. Now the essential part of the cryptanalyst's work is to find an exploitable bias, due to an unknown design flaw, that none of the up-to-now known test detected. For that, the cryptanalyst generally first designs a new hypothesis testing based on a new test. Let us recall that in fact randomness is a theoretical indeed "philosophical" concept. Practically speaking, it can only be determined and defined relatively to the set of statistical tests used to evaluate it.

In this paper we present a new hypothesis testing based on a χ^2 distribution and called Statistical Möbius Analysis. More precisely, we define as working statistic X the number of monomials of degree exactly d in the *Algebraic Normal Form* (ANF) of all the Boolean functions modeling each of the output bits. The set of these d-monomials which are effectively represented in the ANF, are practically computed by means of the Möbius transform. A secure cryptosystem has a fixed distribution determined by general results on random Boolean functions. Then one-sided tests allow us to check if the constituent Boolean functions are truly random.

These tests have been implemented for a few recently proposed stream ciphers and block ciphers, as well as for the main hash functions. All are known to have passed the previously known statistical tests and thus are considered as having very good random properties. Our main results is that famous cryptosystems such AES, DES, Snow and Lili-128 did not pass our tests, wholly or partly. Other results as well as detailed data will be found in [5].

This paper is organized as follows. Section 2 presents the necessary preliminaries and gives the characterization of the Algebraic Normal Form (ANF) of random Boolean functions. In particular, we complete the results presented in [19], make them more practical and give new results on the total degree of a Boolean function. Section 3 presents the new test we designed whilst Section 4 gives detailed numerical results that have been obtained for a few stream ciphers (Lili-128, Snow, BGML and RC4), block ciphers (DES and AES) and hash functions (SHA-0, SHA-1, Ripe-MD, Ripemd160, Haval, MD4 and MD5).

2 Characterization of Boolean Functions and Results

In this section, we present a new statistical way of describing a Boolean function by use of its ANF. This latter can be uniquely computed by means of the Möbius transform. We deduce results on the balancedness and correlation properties with the help of the Walsh transform.

2.1 Structure of the Algebraic Normal Form

A Boolean function is a function f from $\mathbb{F}_2^n$ to $\mathbb{F}_2$. The number of such functions is 2^{2^n}. We define a random Boolean function as a function f whose values are independent, identically distributed (*i.i.d.*) random variables that is to say

$$\forall (x_1, \ldots, x_n) \in \mathbb{F}_2^n, \quad P[f(x_1, \ldots, x_n) = 0] = \frac{1}{2}. \tag{1}$$

In other words, every $f(x_1, \ldots, x_n)$ is a *Bernoulli* random variable of parameter $\frac{1}{2}$. The corresponding probabilistic law will be denoted $\mathcal{B}(p)$ whith $p = \frac{1}{2}$ in our present case[1].

The weight of a Boolean function over $\mathbb{F}_2^n$ is defined by $wt(f) = |\{x \in \mathbb{F}_2^n | f(x) = 1\}|$. Then a Boolean function will be said to be *balanced* if $wt(f) = 2^{n-1}$. Note that a random Boolean function, as defined above, may be not balanced. In fact we will give the general probability for such a function to be balanced.

The *Algebraic Normal Form* (ANF) of f is the multivariate polynomial given by $f(x_1, \ldots, x_n) = \bigoplus_{u \in \mathbb{F}_2^n} a_u x^u, \quad a_u \in \mathbb{F}_2$, where $u = (u_1, \ldots, u_n)$ and $x^u = \prod_{i=1}^n x_i^{u_i}$. The a_u are given by the Möbius transform [14] of f:

$$a_u = \bigoplus_{x \preceq u} f(x) \tag{2}$$

where $\preceq$ denotes the partial order on the Boolean lattice, that is to say that $\alpha \preceq \beta$ if and only if $\alpha_i \leq \beta_i$ for all $1 \leq i \leq n$. A monomial $a_u x^u$ of the ANF will then be said of degree k if $a_u = 1$ and if $wt(u) = k$ where $wt(.)$ denotes the Hamming weight. With these notations we now can state:

Proposition 1 *The Algebraic Normal Form (ANF) of a random Boolean function f from $\mathbb{F}_2^n$ to $\mathbb{F}_2$ has 2^{n-1} monomials in average. For every k such that $0 \leq k \leq n$, there are an average of $\frac{1}{2}\binom{n}{k}$ monomials of degree k.*

When $k = 0$ (resp. $k = n$), it is equivalent to assert that half of randomly chosen Boolean functions contains a_0 (resp $a_{(111\cdots11)}$) in their ANF.

Proof. A given monomial $x_{i_1} x_{i_2} \ldots x_{i_k}$ of degree k will be part of the ANF if and only if $a_u = 1$ where the support of u (that is to say the set of indices j such that $u_j = 1$ and denoted $supp(u)$) is $\{i_1, i_2, \ldots, i_k\}$. Now we have

$$a_u = f(\overline{0})) \oplus \bigoplus_{j=1}^{k} f(e_{i_j}) \oplus \left(\bigoplus_{l=1}^{k} \bigoplus_{j=1, j \neq l}^{k} f(e_{i_j} \oplus e_{i_l}) \right) \oplus \ldots \oplus f(\bigoplus_{j=1}^{k} e_{i_j}), \tag{3}$$

[1] Every n-tuple $(x_1, \ldots, x_n)$ is randomly and independently chosen, then $f(x_1, \ldots, x_n)$ too. It is equivalent to randomly choose f from the set of Boolean functions.

where $\overline{0} = (0, 0, \ldots, 0)$ and e_i is the n-uple whose only its i-th coordinate is non zero. The right side of Equation (3) has $\sum_{j=1}^{k} \binom{k}{j} = 2^k$ terms. We have $a_u = 1$ if an odd number of terms are all equal to 1. There are 2^{k-1} such odd configurations. Each of them, according to (1) has probability $\frac{1}{2^k}$ to be equal to 1 since we consider *i.i.d.* variables. Whence we have $P[a_u = 1] = 2^{k-1} \times \frac{1}{2^k} = \frac{1}{2}$. Thus the number of monomials of degree k in the ANF will be $P[a_u = 1] \times \binom{n}{k} = \frac{1}{2} \times \binom{n}{k}$. □

We can in fact generalize this results with the following theorem:

Theorem 1 *With the notation of Proposition 1, the number n_k of monomials of degree k has normal distribution with mean value $E[n_k] = \frac{1}{2}\binom{n}{k}$ and variance $V[n_k] = \frac{1}{4}\binom{n}{k}$.*

To be mathematically rigorous, we should consider the binomial distribution instead of the normal distribution. Moreover, we should write "X tends toward normal distribution" rather than "X has normal distribution". However, probability theory [4] entitle us such shortcuts as soon as the conditions of application for the Central Limit Theorem are fulfilled. It is the case in our work.

Proof. The proof is straightforward when considering that a_u, for all $u \in \mathbb{F}_2^n$ is a Bernouilli random variable with parameter $\frac{1}{2}$, where $E[a_u] = \frac{1}{2}$ and $V[a_u] = \frac{1}{4}$. Since $n_k = \sum_{wt(u)=k} a_u$, for large enough values of the number of u of weight k, the Central Limit Theorem gives the result (as soon as $n_k \geq 30$ [4]). □

This proposition allows to study the randomness properties of a Boolean function. Let us consider a function f used for the feedback of a shift register of length L. If f is constant (its ANF has only one monomial), the output will not be random at all. In the case of the linear feedback (the ANF of f is of degree 1 and has at most n monomials), the randomness properties are limited: the linearity properties are not suppressed, and combinatorial information is easy to get (for details see [11]). Moreover, it is very easy to reconstruct the feedback polynomial with only $2L$ output bits [16]. This is due to the fact that linear functions have very limited randomness properties.

In other words, if we consider $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$ such that (*e.g.*) $f(x) = f(y) = 1$, the less random the function is, the easier is the extraction of information on x and y.

Example 1 *Let us take $f(x_1, x_2) = x_1 \oplus x_2$. Any $x = (x_1, x_2)$ and $y = (y_1, y_2)$ with $x \neq y$ such that $f(x) = f(y) = 1$ will satisfy $x_1 \oplus y_1 = 1$. This comes from the fact that the values of the truth table are "structured" and not "randomly spread" into this table.*

Proposition 1 gives us the following criterion:

Corollary 1 *A Boolean function used for cryptographic applications and presenting the best trade-off in terms of its cryptographic properties must have a degree as high as possible.*

Proof. This directly comes from the fact that a n-variable random Boolean function in average has its term of degree n with probability $\frac{1}{2}$ and will contain $\frac{n}{2}$ terms of degree $n-1$. According to the upper bound of the degree [23] of a function presenting the best trade-off in terms of correlation immunity, balancedness, ..., we have for a t-correlation immune function: $\deg(f(x_1,\ldots,x_n)) \leq n-t-1$. Constraining the function with given properties lowers the algebraic degree. Combinatorial structures are introduced while randomness is lessened. In the search for the best possible trade-off, to keep good randomness properties by forbidding to get combinatorial information on the function inputs, the function should have the highest possible degree. $\square$

2.2 Characterization of the Walsh Coefficients

The *Walsh Hadamard transform* of a Boolean function f refers to the following transformation: $\forall u \in \mathbb{F}_2^n, \quad \widehat{\chi_f}(u) = \sum_{x\in\mathbb{F}_2^n}(-1)^{f(x)+<x,u>}$, where $<x,u>$ denotes the usual scalar product computed over $\mathbb{F}_2^n$. A well-known result allows to characterize the correlation immunity of f with the Walsh Hadamard transform:

Proposition 2 *[24] A Boolean function f is t-order correlation immune if and only if* $\forall u \in \mathbb{F}_2^n, \quad 1 \leq wt(u) \leq t \qquad \widehat{\chi_f}(u) = 0.$

Moreover f is balanced if and only if $\widehat{\chi_f}(0,0,\ldots,0) = 0$.

Proposition 3 *Let f be a random Boolean function over $\mathbb{F}_2^n$ with $n \geq 5$. For all $u \in \mathbb{F}_2^n$, $\widehat{\chi_f}(u)$ is a random variable which has Gaussian distribution with mean value 0 and variance 2^n.*

Proof. First we can write $\widehat{\chi_f}(u) = \sum_{x\in\mathbb{F}_2^n}(-1)^{f(x)+<x,u>} = (2^n-2)\cdot\sum_{x\in\mathbb{F}_2^n}(f(x) + <x,u>)$. Since x and $f(x)$ are independent, we can consider $<x,u> + f(x)$ as independent, identically distributed random variables for all x as well. Let us note $Y = \sum_{x\in\mathbb{F}_2^n}(f(x)+ <x,u>)$. For $n > 5$ (that is to say $2^n > 30$), due to the central limit theorem [4], Y has a Gaussian distribution $\mathcal{LG}(E,\sigma^2)$ with

$$\begin{aligned} E[Y] &= 2^n P[f(x)+ <x,u>= 1] = 2^{n-1} \\ (\sigma_Y)^2 &= 2^n P[f(x)+ <x,u>= 1]P[f(x)+ <x,u> \neq 1] = 2^{n-2}. \end{aligned}$$

Hence $\widehat{\chi_f}(u)$ has Gaussian distribution with mean value $E[\widehat{\chi_f}(u)] = 2^n(1 - 2P[f(x)+ <x,u>= 1]) = 0$ and variance $\sigma^2 = 4.2^n P[f(x)+ <x,u>= 1]P[f(x)+ <x,u> \neq 1] = 2^n$. $\square$

If Φ denotes the normal distribution function, $\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x}\exp\left(-\frac{t^2}{2}\right)dt$ and if $p_0 = \Phi(\frac{1}{2^{\frac{n}{2}-1}}) - \frac{1}{2}$, we then can state

Lemma 1

$$P[f \text{ } balanced \text{ }] = p_0.$$

Proof. For a balanced Boolean function, we have $\widehat{\chi_f}(0,\ldots,0) = 0$. By definition, $\widehat{\chi_f}(u)$, $\forall u \in \mathbb{F}_2^n$ is even. Then we have $P[\widehat{\chi_f}(u) = 0] = P[0 < \widehat{\chi_f}(u) < 2]$. The rest is straightforward to proove with Proposition 3. □

Remark.- This result is an accurate approximation of the "exact" probability for a function to be balanced given by $p = \frac{\binom{2^n}{2^{n-1}}}{2^{2^n}}$. Table 1 compares exact probability with that computed with Theorem 1 for $5 \leq n \leq 19$. Note that computing exact probability p is highly time consuming while computation time is negligible for p_0.

Table 1. Exact and approximate probabilities for a function to be balanced

n	p	p_0	n	p	p_0	n	p	p_0
5	0.1399	0.1381	10	0.02493	0.02491	15	0.004408	0.004407
6	0.09935	0.09870	11	0.01763	0.01762	16	0.003117	0.003116
7	0.07039	0.07015	12	0.01247	0.01246	17	0.002204	0.002203
8	0.04982	0.49738	13	0.008815	0.008814	18	0.001558	0.001558
9	0.03524	0.03521	14	0.006233	0.006233	19	0.001102	0.001101

3 The New Statistical Testing

We now present the different tests we built up to evaluate new statistical properties of symmetric cryptosystems and hash functions. Let us now consider such a cryptosystem and specify the context we choose. Let there be a secret key $K = (k_0, \ldots, k_{n-1})$. A stream cipher can be seen as follows: every output bits i generated from the secret key K can be expressed by a unique ANF (by means of the Möbius transform defined by Equation (2)).

In other words, the N-bits output sequence can be described by a family of N Boolean functions $(f_t(K))_{0 \leq t < N} = (f_0(K), \ldots, f_{N-1}(K))$ where $f_i(K)$ denotes the i-th bit produced by the system and modelled as a polynomial in variables k_i (ANF). Each output bit is a Boolean function $f_t : \mathbb{F}_2^n \mapsto \mathbb{F}_2$.

Similarly, let us represent a block cipher with n-bit key K working on m-bit blocks. In the same way, but with the different output functions being evaluated on the key space and the plaintext space $P = (p_0, \ldots, p_{m-1})$, for a block cipher C, we then have $C = (c_0, \ldots, c_{m-1}) = (f_0(K,P), \ldots, f_{m-1}(K,P))$. Each of the m ciphertext bits is a Boolean function $f_t : \mathbb{F}_2^{n+m} \mapsto \mathbb{F}_2$.

A hash function $H : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ will have its m-bit message digest of block $B = (b_0, \ldots, b_{n-1})$ represented by $(h_t(B))_{0 \leq t < m} = (h_0(B), \ldots, h_{m-1}(B))$. In the rest of this paper we will use indifferently the term *output bits* and *output Boolean Functions* (or output ANFs for short) to describe the quantities produced by the cryptosystem we consider. At last we will consider that the different output Boolean functions (or bits) are statistically independant. It is precisely the result stated by previous usual, known tests.

The complete output ANF cannot be computed since it contains in average 2^{n-1} monomials. It would require exponential memory and computing time complexity. For our tests we only focus on the monomials of degree at most 3 and need only to compute the 3-truncated ANF, that is to say the partial ANF whose coefficients are effectively computed up to degree 3. For a few cases, 5-truncated ANFs have been computed when necessary. From a practical point of view, we use Formula (3) to produce them. As a result, we observe in every ANF, $\hat{n}_d$ monomials of degree exactly d.

Let us now note H_0^d the statistical hypothesis that the number $\hat{n}_d$ of monomials of degree exactly d is distributed according to the Theorem 1. In other words, the cryptosystem passes our tests and thus exhibits no particuliar structural, statistical bias for the aspect we consider when satisfying this hypothesis.

We suppose the reader is familiar with basic probability and statistics theories (for a detailed presentation see [4] and [15, Chap 5.4]).

3.1 The Affine Constant Test

Our hypothesis is then denoted H_0^0. According to Theorem 1, the probability for the affine constant a_0 to be represented in each of the output ANFs is $p = \frac{1}{2}$. Equivalently, it means that the number of output Boolean functions having $a_0 = 1$ in their ANF has normal distribution $\mathcal{N}(\frac{N}{2}, \frac{\sqrt{N}}{2})$ where N is the total number of output ANFs.

If X_S, the number of times $a_0 = 1$, is the statistic we consider over the sample output S of N ANFs, we can now describe the following two-sided test, called the *Affine Constant Test*:

1. Compute X_S over S.
2. Let us fix a significance level α (*i.e.* probability of rejecting H_0^0 when it is true) and choose a threshold x_α so that for a statistic X of normal standard distribution we have $P[X > x_\alpha] = P[X < x_\alpha] = \frac{\alpha}{2}$.
3. If the value $\hat{X}_S = \frac{X_S - \frac{N}{2}}{\frac{\sqrt{N}}{2}} > x_\alpha$ or if $\hat{X}_S < -x_\alpha$ then H_0^0 is rejected (the system fails the test) otherwise H_0^0 is kept (the system passes the test).

3.2 The d-monomial Tests

We are now considering the monomials of degree exactly d in the output ANFs. Our testing is now denoted H_0^d.

With the notation of Theorem 1, the number of monomials of degree d in a Random Boolean Function ANF is a random variable which is $\mathcal{N}(\frac{1}{2}\binom{n}{d}, \frac{1}{2}\sqrt{\binom{n}{d}})$ distributed. We now consider two *goodness-of-fit*, one-sided tests between the expected frequencies (denoted n_d) and those (denoted $\hat{n}_d$) we observe for the considered cryptosystem.

The first test, T_1^d consider every different ANF and thus has a rather local scope by giving more weight to very weak output ANFs. The second one, T_2^d, groups the N output ANFs according to a few numbers of sets or classes. So to summarize, we will use the χ^2 distribution with ν degrees of freedom by considering the sum of the ν squared, independent random variables $\frac{(n_d^i - \hat{n}_d^i)}{\sqrt{n_d^i}}$ $(i \leq \nu)$ which have by definition standard normal distribution.

In T_1^d we have $\nu = N - 1$ (*i.e.* the number of output ANFs) while for T_2^d we choose $2 \leq \nu \leq 9$

1. Compute for each of the ν random variables n_d^i and $\hat{n}_d^i$ (n_d^i is given by applying Theorem 1).
2. Let us fix a significance level α and a threshold value x_α (computed directly from the cumulative density function of the χ^2 distribution) so that for a statistic X over a random sample we would have $P[X > x_\alpha] = \alpha$ (when X follows a χ^2 distribution with ν degrees of freedom).
3. Compute the statistics D^2 given by $D^2 = \sum_{i=1}^{\nu} \frac{(n_d^i - \hat{n}_d^i)^2}{n_d^i}$.
4. If $D^2 > x_\alpha$ then we must reject H_0^d (the system fails the test and thus presents a statistical bias) otherwise we keep H_0^d (the system does not present any significative bias).

Test T_2^d is intended to describe the considered cryptosystem from a global point of view. In particular it aims at verifying if local biases (detected with T_1^d) are still really significative at a more global level. Instead of dealing with the observed frequencies $\hat{n}_d^i$ of d-monomials for each of the N output ANFs we rather are interested with the number of output ANFs whose number $\hat{n}_d$ belongs to a given, predefined intervall $[a, b[$. The expected frequency for every class is computed from Theorem 1 by applying basic probability results.

3.3 The *d*-monomial Tests on a Given Output ANFs Subset

Essentially, we consider the tests of Section 3.1 and 3.2 but on particuliar subsets S of output ANFs. These test are intended to detect subsets of weak output ANFs. They are denoted $T_i^d|S$ where $i = 1, 2$. Accordingly to the probability and statistics theories, results for which a given cryptosystem exhibits weaknesses must be thoroughly examined and inspected. Complementary results on sampling theory must be taken into account to discriminate "normal but extremal results" (that is to say samples S for which $P[X > x_\alpha] = \alpha$ whilst having truly random distribution) from "truly non-random behaviour".

For all these tests and in all our experiments , we considered $\alpha = 0.05, 0.01$ and 0.001.

4 Simulation Results

4.1 Stream Ciphers

We will here mainly focus on two stream ciphers that have been proposed for the NESSIE Open Call for Cryptographic Primitives [18]: Lili-128 and Snow. Other stream ciphers have been tested or are currently under testing. Table 2 summarizes results for a few of them. We considered the first $N = 6016$ output bits in our experiments.

It is worth noticing that:

Table 2. Stream Ciphers: Tests Results (significance levels $\alpha = 0.001$)

	T_1^1	T_1^2	T_2^1	T_2^2		T_1^1	T_1^2	T_2^1	T_2^2
Lili-128	**fail**	**fail**	**fail**	**fail**	RC4 [20]	pass	pass	pass	pass
Snow	pass	pass	**fail**	**fail**	Bgml [18]	pass	pass	pass	pass

– All the tested stream ciphers pass the Affine Constant test except Lili-128.
– Lili-128 exhibits extremely strong biases. Table 3 presents the results for this stream cipher. These biases have been analyzed and exploited for an operational cryptanalysis in [6].
– Snow exhibits strong biases too but only when considering global statistical behavior. Unfortunately these biases allowed us to design a complete, operationnal cryptanalysis of Snow [6].
– We can give the following interesting observations based on the comparison of the tests convergence (that is to say the distance between the estimator and the threshold value; for details see [12]). The ciphers of Table 2 can be ranked according to their relative "random" quality. We observe that ($\succeq$ means "better than") Bgml $\succeq RC4 \succeq$ Snow $\succeq\succeq$ Lili-128.
– Note that the existence of "weak keys" in stream ciphers like Lili-128 (for example all zero secret key) can only very partly explain these bad statistical results (it only affects the Affine Constant test). Snow presents bad results too whilst it does not have any weak key.
– Second version of Snow and Lili-128 exhibit the same weaknesses.

Table 3. Lili128: Experimental results for tests T_1^d and T_2^d.

	T_1^1	T_1^2	T_2^1	T_2^2
D^2	39,344.03	400,839.93	667729.02	1,028,048.45
$\chi^2_{0.001}$	6349.15			

4.2 Block Ciphers

We mainly focus on the DES [7] and the AES [1]. Results for other block ciphers will be found on [5]. For block ciphers we considered both the encryption ANFs and the decryption ANFs. Since every output ANF involves both plaintext and key variables, tests T_2^d ($d = 1, 2$) have been replaced by tests T_1^d relatively to:

– the number n_1 of plaintext variables from one side and of key variables from the other side (denoted respectively $T_1^1|p$ and $T_1^1|k$).
– the number n_2 of 2-monomials respectively involving plaintext/plaintext variables, key/key variables and plaintext/key variables (tests denoted respectively $T_1^1|pp$, $T_1^1|kk$ and $T_1^1|pk$).

The DES.- Table 4 gives detailed experimental results of the estimator D^2 with 63 degrees of freedom. The critical values are $\chi^2 = 82.52$ ($\alpha = 0.05$), $\chi^2 = 92.01$ ($\alpha = 0.01$) and $\chi^2 = 103.44$ ($\alpha = 0.001$).

It is worth noticing that:

Table 4. DES: Values of Estimator D^2

	T_1^1	T_1^2	$T_1^1\|p$	$T_1^1\|k$	$T_1^1\|pp$	$T_1^1\|kk$	$T_1^1\|pk$
Encr. + IP	35.06	37.65	34.75	35.57	35.41	33.47	33.25
Decr. + IP	33.68	33.93	34.75	39.74	35.41	39.12	29.95

- DES passes the Affine Constant Test in all modes and all significance levels.
- The overall statistical quality is slightly different for encryption and for decryption (in particular the statitical results are slightly better for encryption when only the key is considered).
- DES fails the tests $T_1^1|S$ for many subsets S. For example, several 3-uples including output ANFs 0 and 22 do not pass the test. The overall results present a significant difference for the DES with or without IP. According to the results for the tests $T_1^1|S$ and $T_1^2|S$, the different modes of DES can be ranked in the following manner (($\succeq$ means "better than"):
  {DES Encr. - IP, DES Decr. + IP, DES Decr. - IP} $\succeq$ DES Encr. + IP.
  For these tests, the initial permutation IP improves the overall statistical quality for encryption only. Nevertheless IP is usually discarded by cryptology community when considering its cryptanalysis.

The AES.- We will focus on the algorithm working on 128-bit blocks and with 128-bit secret key. Table 5 gives detailed experimental results of the estimator D^2 with 127 degrees of freedom. The critical values for $\alpha = 0.05$ is $\chi^2 = 159.59$ It is worth noticing that:

- AES passes the Affine Constant Test in all modes and all significance levels.
- Overall statistical quality of AES (128, 128) is good. Partial results on tests T_1^5 and T_2^5 indicate that AES do not pass the test. Moreover AES (encryption and decryption) do not pass the tests $T_1^1|S$ and $T_1^2|S$ for many subsets S. As an example, 3-uples containing output ANFs 52 and 110 are weak subsets for encryption. These biases are currently exploited to greatly improve the cryptanalysis of AES.
- Encryption and decryption exhibits quite the same overall statistical properties.

Table 5. AES (128, 128): Values of Estimator D^2

	T_1^1	T_1^2	$T_1^1\|p$	$T_1^1\|k$	$T_1^1\|pp$	$T_1^1\|kk$	$T_1^1\|pk$
Encryption	59.61	71.32	57.84	61.51	64.47	72.34	62.39
Decryption	67.38	62.27	67.21	70.70	71.26	60.11	47.27

4.3 Hash Functions

We tested the following hash functions: SHA-0 [9], SHA-1 [10], Ripemd160 [3], MD4 [21], MD5 [22], Ripe-MD [2] and Haval [25] (for this latter we tested all the different versions). Extensively detailed numerical results (due to lack of space) are only available in [5]. Tests $T_1^1|S$ and $T_1^2|S$ are under way.

All the tested hash functions have passed the tests whatever may be the significance level. However we can once again give the following interesting observations based on the comparison of the tests convergence.

- The different hash functions can be ranked according to their relative "random" quality. For example when considering results of test T_1^1 (1-monomials), which is the most interesting, we have the following ordering ($\succeq$ means "better than"):
  - 160-bit Message Digest: SHA-1 $\succeq$ (5, 160)-haval $\succeq$ Ripemd160 $\succeq$ (4, 160)-haval $\succeq$ (3, 160)-haval $\succeq$ SHA-0.
  - 128-bit Message Digest: (5,128)-haval $\succeq$ Ripe-MD $\succeq$ MD5 $\succeq$ (4,128)-haval $\succeq$ (3,128)-haval $\succeq$ MD4.
- SHA-1 has indeed better statistical properties than SHA-0, especially when considering the degree 1. The inclusion of the 1-bit rotation in the block expansion from 16 to 80 words really improved the randomness properties of the hash function.
- For the Haval family, the random quality increases with the number of rounds.

Table 6 presents the results of the tests T_1^d and T_2^d for $d = 1, 2$ and for the 160-bit message digest hash functions (significance level $\alpha = 0.05$; let us recall that passing the tests for significance level α imply passing the test for $\alpha' < \alpha$ since $\chi^2_{\alpha'} > \chi^2_{\alpha}$).

Table 6. Experimental results for tests T_1^d and T_2^d ($d = 1, 2$, $\alpha = 0.05$).

Hash Functions	T_1^1		T_1^2		T_2^1		T_2^2	
	D^2	χ^2	D^2	χ^2	D^2	χ^2	D^2	χ^2
SHA-1	76.87	189.52	70.89	189.52	0.04	5.99	0.42	5.99
(5,160)-haval	76.34		79.76		0.17		2.02	
Ripemd160	77.51		66.72		5.24		2.66	
(4,160)-haval	83.52		74.18		1.77		3.51	
(3,160)-haval	83.79		64.28		1.05		5.50	
SHA-0	97.08		74.50		3.26		0.42	

5 Conclusion

This paper presents a new statistical testing of symmetric ciphers and hash functions. Where previous known tests did not exhibit particuliar bias, these new tests reveal structural, statistical biases for DES, AES, Snow and Lili-128. Other cryptosystems are currently tested and may present unsuspected biases.

These tests are still rather quantitative tests but nonetheless they allow to detect possible structural weaknesses in the output ANFs. Current research focuses on more qualitative test involving factorial experiments. It should provide necessary information to greatly improve previous cryptanalytic techniques.

Acknowledgement

Many thanks to Ralph Wernsdorf (Rohde & Schwarz SIT) who detected a few errors in the simulations and in the results. Thus he helped me to greatly improved them and consequently this paper.

References

1. *http://www.nist.gov/aes/*
2. A. Bosselaers, B. Preenel editors, *Intregrity Primitives for Secure Information Systems: Final Report of RACE Integrity Primitives Evaluation RIPE-RACE 1040*, LNCS 1007, Springer, 1995.
3. H. Dobbertin, A. Bosselaers, B. Preenel, RIPEMD-160: a Strengthened Version of RIPEMD. In. *D. Gollman ed., Fast Software Encryption, Third International Workshop*, LNCS 1039, Springer, 1996.
4. W. Feller, *An Introduction to Probability Theory*, Wiley, 1966.
5. *http://www-rocq.inria.fr/codes/Eric.Filiol/index.html*
6. E.Filiol, New Combinatorial Cryptanalysis Techniques, Private Report, 2002.
7. FIPS 46, *Data Encryption Standard*, Federal Information Processing Standards Publication 140-1, US Dept of Commerce/NIST, 1977.
8. FIPS 140-1, *Security Requirements for Cryptographic Modules*, Federal Information Processing Standards Publication 140-1, US Dept of Commerce/NIST, 1994.
9. FIPS 180, *Secure Hash Standard*, Federal Information Processing Standards Publication 180, US Dept of Commerce/NIST, 1993.
10. FIPS 180-1, *Secure Hash Standard*, Federal Information Processing Standards Publication 180-1, US Dept of Commerce/NIST, 1995.
11. S.W. Golomb, *Shift Register Sequences*, Aegean Park Press, 1982.
12. R.V. Hogg, E.A. Tanis, *Probability and Statistical Inference*, MacMillan, 1988.
13. D.E. Knuth *The Art of Computer Programming*, Vol. 2, Addison Wesley, 1981.
14. P. J. McCarthy. *Introduction to Arithmetical Functions.* Springer, 1986.
15. A.J. Menezes, P.C. Van Oorschot, S.A. Vanstone, *Handbook of Applied Cryptography.* CRC Press, 1997.
16. J.L. Massey, Shift-Register Synthesis and BCH Decoding, *IEEE Trans. on Inf. Th.*, Vol. IT-15, pp 122–127, 1969.
17. U. Maurer, A Universal Statistical Test for Random Bit Generators, *J. of Cryptology, 5* pp 89-105, 1992.
18. *http://www.cryptonessie.org*
19. D. Olejár, M. Stanek, On Cryptographic Properties of Random Boolean Functions, *Electronic Journal of Universal Computer Science, Vol. 4, Issue 8*, 1998.
20. B. Schneier, *Applied Cryptography*, Wilew et Sons, 2nd ed., 1996.
21. R.L. Rivest, The MD4 Message Digest Algorithm, *Advances in Cryptology - CRYPT0'90*, LNCS 537, Springer, 1991.
22. R.L. Rivest, The MD5 Message Digest Algorithm, Internet Request for Comment 1321, April 1992.
23. T. Siegenthaler, Correlation Immunity of Nonlinear Combining Functions for Cryptographic Applications, *IEEE Trans. on Inf. Th.*, Vol. IT 35, pp 776–780, 1984.
24. G. Xiao, J.L. Massey, A Spectral Characterization of Correlation Immune Functions, *IEEE Trans. on Inf. Th.*, Vol. IT-34, pp 569–571, 1988.
25. Y. Zheng, J. Pieprzyk, J. Seberry, HAVAL - A One-way Hashing Algorithm with Variable Length of Output, *Advances in Cryptology - AUSCRYPT'92*, LNCS 718, Springer, 1993.