The ISC Int'l Journal of Information Security
ISeCure

January 2012, Volume 4, Number 1 (pp. 15–24)
http://www.isecure-journal.org

On the Multi–Chi-square Tests and Their Data Complexity☆

Ali Vardasbi [1,*], Mahmoud Salmasizadeh [1], and Javad Mohajeri [1]
[1] *Electronics Research Center, Sharif University of Technology, Tehran, Iran*

ARTICLE INFO.

Article history:
Received: 24 December 2011
Revised: 15 March 2012
Accepted: 16 March 2012
Published Online: 3 July 2012

Keywords:
Multi–Chi-square Test, Distinguishing Attacks, Critical Degrees of Freedom, Trivium.

ABSTRACT

Chi-square tests are generally used for distinguishing purposes; however when they are combined to simultaneously test several independent variables, extra notation is required. In this study, the chi-square statistics in some previous works is revealed to be computed half of its real value. Therefore, the notion of Multi–Chi-square tests is formulated to avoid possible future confusions. In order to show the application of Multi–Chi-square tests, two new tests are introduced and applied to reduced round Trivium as a special case. These tests are modifications of the ANF monomial test, and when applied to Trivium with the same number of rounds, the data complexity of them is roughly 2^4 times smaller than that of former ANF monomial test.

In a Multi–Chi-square test the critical degrees of freedom is defined to be the minimum value of the degrees of freedom for which the test is successful at distinguishing the samples set from random. This study investigates the relation between this critical value and the chi-square statistic of a Multi–Chi-square test. In the sequel, by exploiting this relation, a method to approximate the data complexity of a distinguishing Multi–Chi-square test is introduced and shown to perform properly in the special case of reduced round Trivium.

1 Introduction

Pearson's chi-square test is a widely used statistical test which was exploited in distinguishing attacks on stream ciphers for detecting nonrandom properties [1–3]. The distribution of the keystream bits in recent stream ciphers is hardly distinguishable from a random binary sequence. This property leads the cryptanalysts to investigate other ways of distinguishing a stream cipher; an effort which led to the introduction of d-monomial tests [3].

In this study the notion of the Multi–Chi-square test, which is a combination of several chi-square tests, is introduced and exploited to improve the results of former monomial tests. Furthermore, in order to determine an approximation method for the data complexity of the distinguishing Multi–Chi-square test, a relation between the data complexity and the test statistics is presented.

1.1 Previous Works

Testing the monomials of a Boolean function was introduced as the name of d-monomial test by Filiol [3]. However, the stream ciphers tested in Filiol's work did not have IV, so Filiol considered the key/keystream relations as Boolean functions. Saarinen improved Filiol's work and modified it by using the IV bits instead of the key bits [2].

☆ This article is an extended/revised version of an ISCISC'11 paper.

* Corresponding author.
Email addresses: vardasbi@alum.sharif.edu (A. Vardasbi), salmasi@sharif.edu (M. Salmasizadeh), mohajer@sharif.edu (J. Mohajeri).

ISSN: 2008-2045

After Saarinen, Englund *et al.* improved the d-monomial tests by introducing three types of monomial test [1]. They consider a subset of IV bits as the input variables to Boolean functions, leaving all the other IV bits and the key bits fixed. Consequently, considering the first keystream bit as the Boolean function's output, they are able to perform a chosen IV attack by querying the stream cipher for different IVs, computing the ANF representation of the considered Boolean function and testing its monomial distribution using chi-square test.

1.2 Our Contribution

In this study it is shown that the chi-square test in a number of previous monomial tests lacks a factor of two. The missing factor two in previous tests is caused by considering the test as a single chi-square test and not dividing the output space exhaustively. In order to avoid future confusion the introduced test is denoted by Multi–Chi-square test; since it is the combination of multiple chi-square tests.

Furthermore, we will improve Englund et al's monomial tests on Trivium in two aspects. Firstly, in case of monomial distribution test we will show that considering the missing factor of two will reduce nearly four bits in analyzing Trivium with the same round. Secondly, in case of maximum degree monomial test we will consider monomials with one degree less than the maximum degree in addition to the maximum degree monomial. Although our results are for reduced round Trivium, the same scenario can be applied to other reduced round stream ciphers which were studied in [1].

Moreover, the notion of critical degrees of freedom in a Multi–Chi-square test is defined and its relation to the chi-square statistic of the test is formulated. This relation is subsequently utilized to approximate the data complexity for which the Multi–Chi-square test is supposed to distinguish the samples set from random.

1.3 Outline

The next section describes some preliminaries and the notations needed for the successive sections. In section 3 we introduce and formalize the notion of Multi–Chi-square test and explain the relation between degrees of freedom and the statistics of these tests. Furthermore, the definition of critical degrees of freedom alongside its usage in the approximation of data complexity is appeared in this section. Section 4 is devoted to the description of the tests and their advantages to their predecessors. Section 5 contains the experimental results of performing the introduced tests on Trivium. Finally, we will conclude this study in section 6.

2 Preliminaries and Notations

In this section, the chi-square distribution, Pearson's chi-square test and Boolean functions are briefly explained.

2.1 Chi-square Distribution and Pearson's Test

Assume that $\{z_i\}_{i=1}^{k}$ are k independent standard normal random variables. Let $x = \sum_{i=1}^{k} z_i^2$, then the new random variable x is said to have a chi-square distribution with k degrees of freedom. It is obvious from the definition of chi-square distribution that chi-square random variables are additive; if x_1 and x_2 are two chi-square random variables with k_1 and k_2 degrees of freedom, respectively, then $x_1 + x_2$ is also a chi-square random variable with $k_1 + k_2$ degrees of freedom.

In an attempt for obtaining a criterion to reasonably suppose a correlated system of variables to have a random distribution, Pearson was able to introduce one of the currently most widely used statistical tests; i.e. chi-square test [4]. Originally, the chi-square test is used for testing the output samples of an experiment E, to see whether they fulfill a certain condition, expressed as the null hypothesis, or not. Suppose that the output space of E is partitioned into M subspaces $(A_1, \ldots, A_M)$, for which the expected probability (regarding the null hypothesis) of a sample to occur in A_i is p_i for $i = 1, \ldots, M$. In a test, where from a total number of N samples, there are O_i samples in A_i, Pearson proved that

$$\chi^2 = \sum_{i=1}^{M} (O_i - Np_i)^2/(NP_i) \qquad (1)$$

will have a chi-square distribution with $M - 1$ degrees of freedom, if the null hypothesis holds. Therefore, for χ^2_α such that $Pr(x > \chi^2_\alpha) < \alpha$, where x is a chi-square random variable, if $\chi^2 > \chi^2_\alpha$, it can be said that, with a confidence level of α, the null hypothesis does not hold.

2.2 Boolean Functions

An n-variable Boolean function is a mapping from F_2^n to F_2. The set of all n-variable Boolean functions is shown in this paper by BF_n. There are several ways to uniquely represent a Boolean function. However, in cryptography, the most widely used ones are truth table (TT) representation and Algebraic Normal Form (ANF). The ANF of a Boolean function $f : F_2^n \to F_2$ is defined as:

$$f(x_1, \ldots, x_n) = \sum_{u \in F_2^n} a_u x^u; \quad a_u \in F_2, \qquad (2)$$

where $u = (u_1, \ldots, u_n)$ and $x^u = \prod_{i=1}^{n} x_i^{u_i}$. The term x^u is called a monomial and for $weight(u) = d$ it is referred by a d-monomial. The ANF coefficients a_u can be computed from the truth table from:

$$a_u = \bigoplus_{u' \leq u} f(u'), \tag{3}$$

where $u' \leq u$ means $u'_i \leq u_i$, for all $1 \leq i \leq n$.

It is worth clarifying two different cases about the Boolean functions. When dealing with a specific n-variable Boolean function, if n input bits are independent and uniformly distributed variables, the maximum information of the function would be n bits. But, this should not be mistaken with the case where n-variable Boolean functions are considered as variables. Since a Boolean function is characterized with its 2^n output bits, a random n-variable Boolean function, selected uniformly from the set of all 2^{2^n} n-variable Boolean functions, has 2^n bits of information.

3 The Notion of Multi–Chi-square Test and Critical Degrees of Freedom

The goal of Pearson's chi-square test is to see whether or not a set of an experiment's output, categorized in M exclusively and exhaustively selected states $A_1, \ldots, A_M$, can be reasonably supposed to be arisen from random sampling [4].

In digital world's cryptography, we are usually dealing with bits. Particularly, in the case of stream ciphers, it is desirable to test a sequence of bits to see whether they are distinguishable from a random sequence of bits; i.e., each bit becomes one, independent of all the other bits, with a probability not far from 1/2.

Each bit can be interpreted as a two state random variable. Thus, to test the indistinguishability of a bit, one can use Pearson's chi-square test with two states. Note that by "bit" we do not necessarily mean one output bit. It can be the 'XOR' sum or any other combination of many output bits.

Sometimes, it is desired to test the indistinguishability of multiple independent bits. In that case, according to the additive property of chi-square distribution (see Section 2.1), one can combine the tests on multiple bits to obtain a new test. The chi-square statistics and the degrees of freedom of the new test will be equal to the sum of its constituent chi-square tests statistics and degrees of freedom, respectively. In the following subsection, the Multi–Chi-square test for distinguishing a real system generated sequence of bits from a random one is described in more details.

3.1 Distinguishing Multiple Bits by Multi–Chi-square Test

Assume that the set $\{x_1, \ldots, x_n\}$ is the samples of an experiment derived from a population with a balanced binomial distribution. The values of $x_1, \ldots, x_n$ can be interpreted as the values of a bit b obtained from n different experiments. We expect to have $P(x_i = 0) = P(x_i = 1) = 1/2$ for $i = 1, \ldots, n$. The output space of the experiment is, therefore, partitioned into two subspaces A_0 and A_1. To test the indistinguishability of b from a random bit, using the Pearson's chi-square test, the null hypothesis assumes $P(b = 0) = P(b = 1) = 1/2$.

Then, the chi-square statistic is calculated from:

$$\chi^2 = \frac{(O_0 - E_0)^2}{E_0} + \frac{(O_1 - E_1)^2}{E_1}, \tag{4}$$

where E_i and O_i are the expected and observed frequencies of members of A_i, respectively. By definition $E_0 = E_1 = \frac{1}{2}n$, and $O_0 + O_1 = n$. Therefore, the chi-square statistic becomes:

$$\chi^2 = \frac{(O_1 - \frac{1}{2}n)^2}{(\frac{1}{4}n)}. \tag{5}$$

Due to the null hypothesis, χ^2 is expected to be a random variable with chi-square distribution with one degree of freedom.

Now suppose that there are several bits for which we are interested in distinguishing from random. Suppose that these bits are represented by $(b_1, \ldots, b_v)$ and the value of each bit is queried n times. By O_i (the first index of $O_{1,i}$ is dropped for simplicity) we mean that the number of times b_i was equal to one, during n queries.

Using chi-square test for each bit, the chi-square statistic for the test on bit b_i will be shown by

$$\chi_i^2 = \frac{(O_i - \frac{1}{2}n)^2}{(\frac{1}{4}n)}. \tag{6}$$

If the null hypothesis is true for all the bits, each χ_i^2 has the distribution of a chi-square with one degree of freedom. Therefore, by the discussion in Section 2,

$$\chi^2 = \sum_{i=1}^{v} \chi_i^2 \tag{7}$$

has a chi-square distribution with v degrees of freedom. Suppose that the cumulative distribution function of a chi-square distribution with v degrees of freedom is $C_{\chi_v^2}(x)$, w the critical value of a chi-square statistic with v degrees of freedom, due to a level of confidence α, is shown by $\chi^2_{v,\alpha}$; i.e.

$$C_{\chi_v^2}(\chi^2_{v,\alpha}) = 1 - \alpha. \tag{8}$$

Whenever $\chi^2 > \chi^2_{v,\alpha}$, it means that the probability of χ^2 to have a chi-square distribution with v degrees of freedom is at most α. Since α is the probability of false alarm, it is usually assigned negligible values such as 2^{-10}.

This kind of test was exploited in previous works, e.g., in [1], for distinguishing purposes. In this section, however, we showed that the chi-square statistics should be computed by:

$$\chi^2_{\text{new}} = \sum_{i=1}^{v} \frac{(O_i - \frac{n}{2})^2}{\frac{n}{4}}, \tag{9}$$

while in the former versions of this test, the chi-square statistic was computed by:

$$\chi^2_{\text{old}} = \sum_{i=1}^{v} \frac{(O_i - \frac{n}{2})^2}{\frac{n}{2}}. \tag{10}$$

More discriminately, we explained why the computation of the chi-square statistics should be done via χ^2_{new} formula rather than $\chi^2_{\text{old}} = \chi^2_{\text{new}}/2$. In other words, a factor of two should be multiplied to the previous chi-square statistics.

The reason that in previous works the chi-square statistic was computed by Equation 10 instead of Equation 9 is that in [1], for example, the experiment of querying v bits $(b_1, \ldots, b_v)$ for n times was considered as a single experiment with the output space categorized by $A = A_1, \ldots, A_v$, where A_i is the event of $b_i = 1$. Remember from Section 2 that in a chi-square test, for a single experiment, the categorization of the output space should be done exclusively and exhaustively. In the setting of [1], however, the categorization is done exclusively, but not exhaustively. Note that, the event $b_i = 0$ is not considered for neither of b_is in the A. In the setting of our Multi–Chi-square test, there are n different experiments, each of which is categorized by $\{A_0, A_1\}$; giving a total of $2n$ different categories. By this modification, the categorization of the output space would be both exclusively and exhaustively.

3.2 The Relation Between Chi-square Statistic and Degrees of Freedom in A Chi-square Test

In a Multi–Chi-square test, we are dealing with multiple bits. Each component chi-square test is performed on one bit, hence having one degree of freedom, thus, the degrees of freedom for the overall Multi–Chi-square test will be equal to the total number of bits involving in the test. Therefore, performing the test on more bits means a higher degree of freedom. In this sense, one can speak of the relation between the degrees of freedom (number of tested bits) and the chi-square statistics in a Multi–Chi-square test. The following lemma states such a relation when the degrees of freedom is free to be selected very large.

Lemma 1. *For $v \in N$, if a chi-square cumulative distribution function with v degrees of freedom is shown by $C_{\chi^2_v}(x)$, then*

$$\lim_{v\to\infty} C_{\chi^2_v}(kv) = \begin{cases} 0, & k < 1; \\ 1/2, & k = 1; \\ 1, & k > 1. \end{cases} \tag{11}$$

Proof. During this proof, x is used to represent a chi-square random variable with v degrees of freedom. The following is a well known statement (see e.g. [5]):

$$v \to \infty \Rightarrow \frac{(x - v)}{\sqrt{2v}} \to^{Dist} N(0, 1), \tag{12}$$

where $\to^{Dist}$ means convergence in distribution. From this, the case of $k = 1$ in relation Equation 11 would be obvious. Here we only prove the $k > 1$ case (the case $k < 1$ can be proved just the same).

If the mean and variance of x is shown by μ and σ^2, respectively. Since x has a chi-square distribution, $\mu = v$ and $\sigma^2 = 2v$. All we have to show is:

$$\frac{|kv - \mu|}{\sigma} \to \infty. \tag{13}$$

Since $C_{\chi^2_v}(kv) = Pr(x < kv)$, the proof is complete. However, the relation Equation 13 can be easily shown:

$$\frac{|kv - \mu|}{\sigma} = \frac{(k-1)v}{\sqrt{2v}} = \sqrt{\frac{(k-1)^2 v}{2}} \to \infty, \tag{14}$$

where the convergence follows from $k \neq 1$ and $v \to \infty$.

■

This lemma gives a somehow good perspective for the results of Multi–Chi-square tests. The practical restatement of the above lemma is as follows. In running a Multi–Chi-square test for several times, when the statistics of the test, kv, becomes greater than the degrees of freedom, v, for most of the times, one can expect the test to distinguish the tested samples from random, rejecting the null hypothesis, if the degrees of freedom is selected sufficiently large.

According to Lemma 1, for a large v and $\chi^2 = kv > v$, we have:

$$C_{\chi^2_v}(x^2) \to 1. \tag{15}$$

Therefore, there exists a χ^2_0 slightly smaller than χ^2 which satisfies:

$$C_{\chi^2_v}(\chi^2_0) > 1 - \alpha. \tag{16}$$

Therefore from $\chi^2 > \chi^2_0$ one can say, with a level of confidence α, that χ^2 is not a sample of a chi-square

random variable with v degrees of freedom, hence rejecting the null hypothesis.

We can deduce from this lemma that when a chi-square test is performed on an indistinguishable sample set, its statistics is expected to fall near the degrees of freedom. Otherwise, increasing the degrees of freedom, which is equal to the number of tested bits, would result in a distinguisher. We will use this fact in section 5 to show the inconsistency of the results of [1] with the chi-square distribution properties discussed in this section.

3.3 Critical Value for Degrees of Freedom in a Multi–Chi-square Test

As was mentioned in Lemma 1, when $k > 1$, we have:

$$\lim_{v\to\infty} C_{\chi^2_v}(kv) = 1. \tag{17}$$

This equation means:

$$\forall \alpha, k > 1, \exists v_c : \forall v (v > v_c \Rightarrow C_{\chi^2_v}(kv) > 1-\alpha). \tag{18}$$

We call v_c the critical value for the degrees of freedom. In order to find the v_c corresponding to a specific α and k, one should solve

$$C_{\chi^2_v}(kv_c) = 1 - \alpha, \tag{19}$$

which leads to

$$\frac{\int_0^{kv_c} t^{v_c/2-1}.e^{-t/2}dt}{\int_0^{\infty} t^{v_c/2-1}.e^{-t/2}dt} = 1 - \alpha. \tag{20}$$

However, one may think of estimating v_c without the need to solve this hard relation. In the following, we will try to obtain an approximate formula for computing v_c as a function of k and α.

It is well known that for a large enough v, $C_{\chi^2_v}(Y)$ can be approximated by $\Phi((Y-v)/\sqrt{2v})$, where $\Phi(.)$ represents the CDF of a standard normal random variable. Using the notation of Q-function, which is defined as:

$$Q(x) = 1 - \Phi(x), \tag{21}$$

one can approximate Equation 19 to obtain the following.

$$\begin{aligned} &\Phi((Y-v)/\sqrt{2v}) \mid_{v=v_c \& Y=kv_c} = 1-\alpha \\ &\Rightarrow Q(\frac{kv_c - v_c}{\sqrt{(2v_c)}}) = Q(\frac{(k-1)}{\sqrt{2}}\sqrt{v_c}) = \alpha \\ &\Rightarrow \frac{(k-1)}{\sqrt{2}}\sqrt{v_c} = Q^{-1}(\alpha) \end{aligned} \tag{22}$$

Since α is the level of confidence and it is usually assigned a value at the beginning of any process, $Q^{-1}(\alpha)$ can be computed, numerically, once at the beginning of the process. Let

$$A \triangleq Q^{-1}(\alpha). \tag{23}$$

The approximate formula for the critical degrees of freedom would be

$$v_c \approx \frac{2A^2}{(k-1)^2}. \tag{24}$$

One application of approximating v_c is in performing Multi–Chi-square tests where the number of tested bits is equal to the degrees of freedom. Hence, v_c can be interpreted as the minimum number of bits required for distinguishing the output of a PRBG. This application is discussed in the next section.

3.4 Approximation of the Data Complexity

Assume a Multi–Chi-square test with v degrees of freedom (i.e., v number of bits) is performed a large number of times. According to Lemma 1, if the computed chi-square statistics is greater than v in almost all the tests, one can expect the tested bits to be distinguishable. Consequently, taking the average of all the chi-square statistics Q^2_{average} could be useful in estimating the critical degrees of freedom v_c (i.e., the minimum number of bits required to distinguish the output of PRBG). In this section, it is shown that v_c is approximated by:

$$v_c \approx \frac{2A^2.v^2}{(Q^2_{\text{average}} - v)^2}. \tag{25}$$

The generated bits in each run of PRBG are expected to be independent. Therefore, applying the Multi–Chi-square test on v bits for t times and adding all chi-square statistics is equivalent to applying the test on $v.t$ bits. If the chi-square statistic of each test on v bits is near Q^2_{average} then the chi-square statistic of the test on vt bits is approximately $t.Q^2_{\text{average}}$. Therefore, the additive property of the chi-square statistics causes the ratio of the chi-square statistics to the degrees of freedom to be constant, i.e.,

$$\frac{t.\chi^2_{\text{average}}}{tv} = \frac{\chi^2_{\text{average}}}{v}. \tag{26}$$

In Lemma 1 this ratio was represented by k, since the chi-square statistics was shown as kv.

To sum up, the minimum number of bits required to distinguish the output of a PRBG can be approximated by:

$$v_c \approx \frac{2A^2}{(k-1)^2}, \tag{27}$$

in which replacing k with $(\chi^2_{\text{average}})/v$ will result in Equation 25.

An interesting observation in Equation 27 is its

similarity to the data complexity formula in linear attacks. The fact that $O(\frac{1}{\epsilon^2})$ bits are required to exploit a linear relation with a bias equal to ϵ is somehow similar to $v_c = O\left(\frac{1}{(k-1)^2}\right)$. It is worth noting that in this setting $k = 1$ is the case where the tested bits are indistinguishable, corresponding to the case of $\epsilon = 0$ in linear attacks.

4 Testing ANF Representation Monomials in Stream Ciphers

A random n-variable Boolean function can be thought of as a random variable which takes as its values members of BF_n with a uniform distribution. This interpretation of random Boolean function can be shown to be the basic definition for random Boolean functions, since all the other definitions of randomness can be extracted from it.

The transformation that maps the TT vector to the ANF vector is bijective, i.e., each TT vector corresponds to a unique ANF vector. Therefore, the ANF vector of a random n-variable Boolean function has all the properties of its TT vector. This leads researchers to test the ANF vector of a Boolean function in case its TT vector does not reveal any distinguishing properties. Using the ANF vector of a Boolean function has been exploited in other works, such as [1, 2, 6, 7]. In this section we will introduce two statistical tests on stream ciphers, or any other cryptography primitive. These tests are the modifications of tests in [1]. We will show in section 5 that they are more efficient and more reality consistent than their predecessors.

4.1 ANF Coefficients Distribution Test, a Modified Version

The most straight forward way of testing the ANF vector of a Boolean function is to test whether or not all its ANF coefficients are equiprobable, i.e., the probability of a co-efficient being '1' is equal to its probability of being '0'. In case of stream ciphers, Englund et al [1] proposed a way for testing. However, there are two things which were left unnoticed in their work. First, as we discussed in Section 3, the chi-square statistics computed in their work lacks a '2' factor. We will show in Section 5, in case of Trivium, the result of this missing factor is at least a division of $O(2^4)$ in the memory and computation complexities of the test, although it may seem less effective.

Secondly, Englund et al considered two extreme cases for testing the ANF coefficients: (a) testing all the coefficients and (b) testing the coefficients of the monomial with all the inputs in it (maximum degree monomial). When these tests are performed on n-variable Boolean functions, case (a) needs a memory of 2^n bits, while case (b) needs only one memory bit. They noticed that, in an ANF representation, the coefficients of higher degree monomials are more vulnerable to being distinguishable than lower degree monomials. However, testing only the maximum degree monomial is a waste of data. We suggest a medium case in which the monomials with one degree less than the maximum are tested. We denote the maximum degree monomial test in [1] by "MD test" and the test in this paper by "MD++ test".

Subsequently, we mention two tests in this section: the first one is the modification of ANF monomial distribution test in [1], with only considering the missing factor. The second test is the MD++ test, which considers the missing factor and tests more than one monomial. We will compare these tests in Section 5, for a special case of Trivium, and show the best result is due to MD++ test.

In the following, the size of IV is represented by m. For a stream cipher, since the IV bits are public, one can obtain a Boolean function which is defined by the mapping from a subset of IV bits to one keystream bit. For concreteness, assume that by "a Boolean function in a stream cipher" we mean the following. For $n < m$, for which computations of order 2^n*are* are feasible, choose n bits from the IV bits and name them as $(x_1, \ldots, x_n)$. Name the remaining IV bits as $(x_{n+1}, \ldots, x_m)$. Since n is chosen such that $O(2^n)$ is practical, one can compute the first keystream bit (z_0), 2^n times, for 2^n different values of $(x_1, \ldots, x_n)$. Note that here the key bits, as well as $(x_{n+1}, \ldots, x_m)$, are fixed during the generation of the first keystream bit. After 2^n computations of z_0, a binary vector of length 2^n will be obtained. This 2^n-bit vector is the TT representation of the Boolean function which maps $(x_1, \ldots, x_n)$ to z_0. We denote this TT vector by Z_0^n.

For every choice of key bits and $(x_{n+1}, \ldots, x_m)$ the above process will give us a Boolean function. Then, the distribution of the ANF coefficients of these Boolean functions are tested using Multi–Chi-square test. Description of the tests is shown in Figure 1. Since the basics of the tests are the same, we write them in one pseudo-code. For the "modified version of ANF distribution test" let $v = 2^n$ and for the "MD++ test" let $v = n + 1$.

It is worth noting that in previous tests, the terms $\frac{P - b_i - P/2^2}{(P/2)}$ were not included in the summation, hence making the chi-square statistics half its real value.

Modified version of ANF distribution test and MD++ test:

$(b_1, \ldots, b_v) \leftarrow (0, \ldots, 0)$

For P different values of $(x_{n+l} \ldots, x_m)$ while the key bits remain fixed

Construct Z_0^n;

Compute ANF coefficients from Z_0^n;

Save the coefficients corresponding to monomials:

`/* in case of ANF distribution test */` with all degrees; or

`/* in case of MD++ test */` with degrees $n-1$ and n;

and put them in $(a_v \ldots, a_v)$;

$(b_1 \ldots, b_v) \leftarrow (b_l, \ldots, b_v) + (a_1 \ldots, a_v)$;

Compute the chi-square statistic: $\chi^2 = \sum_{i=1}^{v} \left(\frac{(b_i - P/2)^2}{P/2} + \frac{(P - b_i - P/2)^2}{P/2} \right) = 2 \cdot \sum_{i=1}^{v} \frac{(b_i - P/2)^2}{P/2}$;

If $\chi^2 > \chi^2_{v,\alpha}$

return "**cipher**";

Else

return "**random**";

Figure 1. Pseudo-code for ANF monomial distribution test and MD++ test

5 Results on Trivium

In this section we present the results of our proposed tests on Trivium.

5.1 Trivium

Trivium is one of eSTREAM candidates in Profile 2 (hardware) that was designed in 2005 by C. De Cannire and B. Preneel [8]. Due to its desired properties including simplicity, speed, and high security, Trivium became a part of the portfolio for Profile 2 in eSTREAM project.

Trivium uses three nonlinear feedback shift registers (NFSR) with different lengths and a total length of 288 state bits. Key and IV are 80 bits vectors and are placed in the first 80 bits of the first and the second register, respectively. The initialization phase consists of $4288 = 1152$ rounds in which no outputs are generated. The keystream generating phase starts after 1152 initialization rounds.

Despite the simple structure that led cryptanalysts to try to attack it, the full version Trivium resisted to all of those attacks. There were some key recovery attacks on Trivium with 576 initialization rounds in negligible time [9], and 672 initialization rounds with complexity 255 [6]. The best key recovery attack on Trivium, so far, is due to the cube attacks [7] that could find the full key of a 735 round Trivium with 230 bit operations.

In case of distinguishing attacks, on the other hand, we can mention the d-monomial test by Englund et al [1] that detected non randomness in a 736 round Trivium with time complexity of about 238. A better work was done, as the name of cube testers, by Aumasson et al [10], which could detect non random properties of Trivium with 790 rounds in 230 time complexity.

5.1.1 A Note on Trivium Initialization

This section contains a discussion of Trivium initialization phase, which is useful in chosen IV attacks on Trivium.

The only nonlinear parts of Trivium are three AND gates at which two adjacent state bits from each register are multiplied during each round. Suppose we want to choose n IV bits as variables, $\{x_1, \ldots, x_n\}$, and fix the remaining IV bits at some constant value. Since the first 80 bits of the second register is loaded with IV bits (Figure 2), it is easy to see that if no two adjacent IV bits are chosen as variables, after approximately 80 rounds, no state bit will be a nonlinear combination of the variables $\{x_1, \ldots, x_n\}$. Therefore, after a specific number of rounds, the state bits in this scenario are a function of $\{x_1, \ldots, x_n\}$ with lower nonlinearity, compared to the case when the variables are chosen from consecutive IV bits.

Furthermore, since the second register's feedback has the term $(s_{162} + s_{177} + s_{175} \cdot s_{176})$, if the distance

$(s_1, s_2, \ldots, s_{93}) \leftarrow (K_1, K_2, \ldots, K_{80}, 0, \ldots, 0)$

$(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (IV_1, IV_2, \ldots, IV_{80}, 0, \ldots, 0)$

$(s_{178}, s_{179}, \ldots, s_{288}) \leftarrow (0, 0, \ldots, 0, 1, 1, 1)$

For $i = 1$ to $4 \cdot 288$ do

$t_1 \leftarrow s_{66} + s_{93}$

$t_2 \leftarrow s_{162} + s_{177}$

$t_3 \leftarrow s_{243} + s_{288}$

$t_1 \leftarrow t_1 + s_{91} \cdot s_{92} + s_{171}$

$t_2 \leftarrow t_2 + s_{175} \cdot s_{176} + s_{264}$

$t_3 \leftarrow t_3 + s_{286} \cdot s_{287} + s_{69}$

$(s_1, s_2, \ldots, s_{93}) \leftarrow (t_3, s_1, \ldots, s_{92})$

$(s_{94}, s_{95}, \ldots, s_{177}) \leftarrow (t_1, s_{94}, \ldots, s_{176})$

$(s_{178}, s_{279}, \ldots, s_{288}) \leftarrow (t_2, s_{178}, \ldots, s_{287})$

Figure 2. Pseudo-code for full round Trivium initialization phase

of any two chosen IV bits' indexes is greater than two, and not a multiple of 15, 14 or 13 , after approximately 80 rounds each state bit will be a function of at most one variable from $\{x_1, \ldots, x_n\}$.

Based on the above discussions, in a chosen IV attack on Trivium, if the IV bits, to stand for the attack's variables, are chosen according to this section's observations, we expect that the results would not be worse than the case when the variables are chosen from consecutive IV bits. Therefore, in choosing the IV bits in our simulations on Trivium, we take these observations into account.

In the following subsections we show the results of performing the tests introduced in section 4 on Trivium. A comparison is also performed between our tests and their predecessors. In all the tests the level of confidence is considered to be $\alpha = 2^{-10}$.

5.2 Modified ANF Monomial and MD++ Tests on Trivium

The results of our tests in various rounds of Trivium are listed in Table 1 for ANF monomial test, and Table 2 for both MD and MD++ tests. The notations are the same as Section 4.

It is worth to note that maximum degree monomial test was also performed in [1] on reduced round Trivium. However, we just mention the results that are a little better than [1]. The reason may lie in the selection of IV bits on which the test is performed. As can be seen, the best results among these four tests are due to MD++ test, especially when the test is performed on higher rounds of Trivium.

Table 1. Results of ANF Monomial Test on Trivium

	ANF monomial test in [1]		**Modified ANF monomial test**	
rounds	P	n	P	n
672	2^8	18	2^7	14
704	2^6	23	2^7	19
736	–	–	2^8	25

Table 2. Results of MD And MD++ Tests on Trivium

	MD test		**MD++ test**	
rounds	P	n	P	n
672	2^5	14	2^5	14
704	2^5	19	2^5	18
736	2^6	26	2^6	25

Table 3. Inconsistency Of χ^2_{old} With What Is Expected

Degrees of freedom	χ^2_{new}	χ^2_{old}	n	**rounds**
4096	4084	2041	12	672
65536	65502	32751	16	704
1048576	1048404	524202	20	736

In addition to our theoretical discussion in Section 3, the missing factor two in the ANF monomial test of [1] can be observed through experimental results. Remember the discussion in Section 3 which came after Lemma 1. The chi-square statistics of a test, when performed on indistinguishable samples set, is expected to be near its degrees of freedom. In case of ANF monomial test in [1], the first row of the table, for example, means the 640 rounds Trivium is indistinguishable for $n < 13$. Thus, for smaller n values from the ones suggested in Table 2, the test on Trivium is expected to have a statistics near 2^n, the degrees of freedom. However, Table 3 shows the test's statistic for the previous version of ANF monomial test [1], which is near 2^{n-1}, half of the degrees of freedom.

In the Table 3, χ^2_{old} is the average of several chi-square statistics without considering the missing factor of two. Clearly, χ^2_{new} is the double of χ^2_{old}, and as can be seen, χ^2_{new} is almost as the degrees of freedom, as expected.

Table 4. Comparison between approximation and practical results

	Approximation method	Modified ANF monomial test
rounds	$\log_2 v_c$	n
672	14.51	14
704	19.65	19
736	26.11	25

5.3 Verification of the Approximation Formula for Data Complexity

In order to show the effectiveness of our introduced method in approximating data complexity, we perform this method on reduced round Trivium. The test for which the approximation method is intended to be verified in this section is the modified ANF monomial test.

The modified ANF monomial test and its results on several versions of reduced round Trivium were presented in previous sections. Table 1 contains the practical values of n for which the test can successfully distinguish the 672, 704, and 736 rounds of Trivium. As it was discussed in Section 4.1, the degrees of freedom in an ANF monomial test is equal to 2^n. Table 4 depicts the comparison between the practical results of Table 1 and the values which are resulted from the approximation formula. In Table 4, the value for the approximated critical degrees of freedom is represented by v_c.

As can be seen in Table 4, there is a slight difference between the approximated and the practical values for the critical degrees of freedom. One reason for such a difference is the fact that our method of approximation is a general method in which the structure of the under test algorithm is not taken into account. However, in case of Trivium, the more the IV bits are involved in the test, the better the nonrandom properties are detectable. Since the approximation is based on the observations from lower values of n where some of the weaknesses of Trivium are not revealed yet, the value for the critical degrees of freedom is approximated greater than its real value.

6 Conclusion

It was shown in this study how the chi-square test on multiple bits should be performed to obtain a more consistent result with chi-square distribution properties. Furthermore, using the Multi–Chi-square test decreases the data and computation complexities. In case of Trivium, we showed this reduction to be approximately four bits of data and a division of $O(2^4)$ computations.

Two tests were introduced which could distinguish reduced round Trivium (as a special case) with more rounds and less complexities compared to previous tests. Thus, considering the input/output relations in a system as Boolean functions, and testing their ANF monomials, is a useful cryptanalysis and there may be lots of work in the future which are based on this approach.

Additionally, the notion of critical degrees of freedom in a Multi–Chi-square test was defined. This definition was exploited as a means to derive an approximation formula for the data complexity in a distinguishing Multi–Chi-square test. In the cases where it is not feasible to practically determine the required bits for which a Multi–Chi-square test is successful in distinguishing, the approach introduced in this study can be exploited to approximate the minimum required bits for a successful test.

References

[1] Håkan Englund, Thomas Johansson, and Meltem Sönmez Turan. A Framework for Chosen IV Statistical Analysis of Stream Ciphers. In *Proceedings of the cryptology 8th international conference on Progress in cryptology*, INDOCRYPT'07, pages 268–281, Berlin, Heidelberg, 2007. Springer-Verlag.

[2] Markku juhani O. Saarinen. Chosen-IV Statistical Attacks on Estream Stream Ciphers. In *eSTREAM, ECRYPT Stream Cipher Project, Report 2006/013*, pages 5–19, 2006.

[3] Eric Filiol. A New Statistical Testing for Symmetric Ciphers and Hash Functions. In *Proceedings of the 4th International Conference on Information and Communications Security*, ICICS '02, pages 342–353, London, UK, UK, 2002. Springer-Verlag.

[4] K. Pearson. On The Criterion that a Given System of Deviations from the Probable in the Case of a Correlated System of Variables is Such that it Can be Reasonably Supposed to Have Arisen from Random Sampling. *Philosophical Magazine Series*, 5(50):157–175, 1900.

[5] M.G. Kendall, A. Stuart, J.K. Ord, and S.F. Arnold. *Kendall's Advanced Theory of Statistics: Classical inference and relationship*. Number v. 2 in Kendall's library of statistics. Arnold, 1999.

[6] Simon Fischer, Shahram Khazaei, and Willi Meier. Chosen IV Statistical Analysis for Key Recovery Attacks on Stream Ciphers. In *Proceedings of the Cryptology in Africa 1st inter-*

national conference on Progress in cryptology, AFRICACRYPT'08, pages 236–245, Berlin, Heidelberg, 2008. Springer-Verlag.

[7] Itai Dinur and Adi Shamir. Cube Attacks on Tweakable Black Box Polynomials. In *Proceedings of the 28th Annual International Conference on Advances in Cryptology: the Theory and Applications of Cryptographic Techniques*, EUROCRYPT '09, pages 278–299, Berlin, Heidelberg, 2009. Springer-Verlag.

[8] Christophe De Cannière. Trivium: A Stream Cipher Construction Inspired by Block Cipher Design Principles. In *Proceedings of the 9th international conference on Information Security*, ISC'06, pages 171–186, Berlin, Heidelberg, 2006. Springer-Verlag.

[9] M. Vielhaber. Breaking ONE.FIVIUM by AIDA an Algebraic IV Differential Attack, 2007. IACR ePrint Archive, Report 2007/413. AVailable from http://eprint.iacr.org/2007/413.pdf.

[10] Jean-Philippe Aumasson, Itai Dinur, Willi Meier, and Adi Shamir. Cube Testers and Key Recovery Attacks on Reduced-Round MD6 and Trivium. In *Fast Software Encryption*, pages 1–22, Berlin, Heidelberg, 2009. Springer-Verlag.

Ali Vardasbi received both B.Sc. and M.Sc. degrees from Sharif University of Technology in 2008 and 2011, respectively. Since then he is a research assistant at the Electronics Research Center of Sharif University of Technology. His research interests include cryptography, design and analysis of algorithms and signal processing.

Mahmoud Salmasizadeh received the B.Sc. and M.Sc. degrees in Electrical Engineering from Sharif University of Technology in Iran, in 1972 and 1989, respectively. He received the Ph.D. degree in Information Technology from Queensland University of Technology in Australia, in 1997. Currently he is an associate professor in Electronics Research Center and adjunct associate professor in Electrical Engineering Department at Sharif University of Technology, Tehran, Iran. His research interests include information-theoretic secrecy, design and cryptanalysis of cryptographic algorithms and protocols and e-commerce security. He is a founding member of Iranian Society of Cryptology.

Javad Mohajeri received his B.Sc. degree from the Isfahan University in 1986, and his M.Sc. degree from Sharif University of Technology in 1989, both in mathematics. Since 1990 he has been a faculty member at Electronics Research Center of Sharif University of Technology. His research interests include cryptography and data security. He is the author/co-author of over 60 research articles in refereed journals/conferences.