A New Approach to χ^2 Cryptanalysis of Block Ciphers

Jorge Nakahara Jr.[1], Gautham Sekar[4,5,$\star$], Daniel Santana de Freitas[2],
Chang Chiann[3], Ramon Hugo de Souza[2], and Bart Preneel[4,5]

[1] EPFL, Lausanne, Switzerland
jorge.nakahara@epfl.ch
[2] Federal University of Santa Catarina, Brazil
{santana,ramonh}@inf.ufsc.br
[3] University of São Paulo, Brazil
chang@ime.usp.br
[4] Interdisciplinary Institute for BroadBand Technology (IBBT), Belgium
[5] Katholieke Universiteit Leuven, Belgium
{gautham.sekar,bart.preneel}@esat.kuleuven.be

Abstract. The main contribution of this paper[1] is a new approach to χ^2 analyses of block ciphers in which plaintexts are chosen in a manner similar to that in a square/saturation attack. The consequence is a faster detection of χ^2 correlation when compared to conventional χ^2 cryptanalysis. Using this technique we *(i)* improve the previously best-known χ^2 attacks on 2- and 4-round RC6, and *(ii)* mount the first attacks on the MRC6 and ERC6 block ciphers. The analyses of these fast primitives were also motivated by their low diffusion power and, in the case of MRC6 and ERC6, their large block sizes, that favour their use in the construction of compression functions. Our analyses indicate that up to 98 rounds of MRC6 and 44 rounds of ERC6 could be attacked.

Keywords: Block ciphers, χ^2, square and linear cryptanalysis.

1 Introduction

In this paper we present a new, generic approach to χ^2 cryptanalysis which combines conventional χ^2 and integral techniques. In this approach, the plaintexts are chosen like in a square/saturation attack, that is, part of the input is fixed and the remaining part is varied exhaustively. Further, the attack is adaptive in the sense that we keep on generating plaintexts until χ^2 correlation is detected. The advantage of this approach is that it allows faster detection of χ^2 correlations in block ciphers compared to previous approaches. One drawback is that it is not straightforward to turn the chosen-plaintext (CP) setting into a known-plaintext (KP) one.

$\star$ This author is supported by an FWO project.

[1] The work described in this paper has been supported, in part, by the European Commission through the ICT programme under contract ICT-2007-216676 ECRYPT II.

We apply this new approach to the block ciphers RC6, ERC and MRC6. RC6 [1] was designed by Rivest *et al.* for the AES Development Process [2]. RC6 was one of the five finalists in the AES competition and was also submitted to NESSIE and CRYPTREC projects. ERC6 [3] is a wide-block variant of RC6, designed by Ragab *et al.* in 2001. MRC6, proposed by El-Fishawy *et al.* in 2004 [4], is another wide-block variant of RC6.

In our attacks, the choice of the plaintext bits to be chosen and the ciphertext bits to be analysed is based on prior linear analysis, which provided the bit positions with highest expected non-uniform bias. Our attacks follow a similar methodology as the mod-n attacks against the block ciphers RC5P and M6 [5].

Our considerations and conclusions of the analyses in this paper are based on empirical data collected through several attack simulations. We have used χ^2 threshold values corresponding to 25% significance level (or 75% specificity). See Table 12 in the appendix. This choice was based on the following reasons:

1. Our aim is to show the effectiveness of our attacks on RC6, ERC6 and MRC6 when compared to conventional χ^2 cryptanalysis with randomly generated plaintexts. Hence, as long as the same significance level is used for the two types of tests, the value of the significance level is irrelevant.
2. Our attack simulations show that the number of chosen plaintexts required with a better (we considered 10%) significance level could be determined by the number corresponding to 25% level.
3. In the literature 25% seems to be an acceptable value [6].

This paper is organized as follows. Section 2 briefly describes χ^2 cryptanalysis and introduces our technique; Sect. 3 gives the specifications of the RC6, ERC6 and MRC6 ciphers; Sect. 4 provides the experimental results of our χ^2 attacks on the three ciphers. Also, in Sect. 4 comparisons are drawn between our attacks and previously applied methods. Section 5 concludes the paper.

2   The χ^2 Test and Our Generic Approach

The χ^2 statistical test has already been applied to a number of ciphers, such as the DES in [7], on SEAL [8], on M6, MX and RC5P [5], on RC5, RC6 and many simplified variants [9,10,11,12,13,14,1,15].

Consider an experiment E with k simple, mutually independent outcomes. Let $o_1, \ldots, o_k$ and $x_1, \ldots, x_k$ denote the observed and expected frequencies, respectively, of the k outcomes when E is performed N times. Therefore, $N = \sum_{i=1}^{k} o_i = \sum_{i=1}^{k} x_i$. For each outcome, there can be a difference between the observed and the expected frequencies. The idea behind a χ^2 test is to combine all these differences into one overall measure of the distance between the data and the expectations of the model. The χ^2 statistic with $k-1$ degrees of freedom is defined [16] as,

$$Q = \sum_{i=1}^{k} \frac{(o_i - x_i)^2}{x_i}, \tag{1}$$

where the sum is over $x_i \neq 0$. When the observed frequency is far from the expected one, the corresponding term $o_i - x_i$ in the sum is large; when they are close, $o_i - x_i$ is small. The quantity Q gives a measure of the distance between the observed and expected frequencies; large values of Q indicate that the observed frequencies are far from the expected ones. In a χ^2 goodness-of-fit test, one defines two hypotheses - the null hypothesis (denoted H_0) and the alternative hypothesis (H_1). The null hypothesis is the one that exists solely to be *falsified* by the sample. If the null hypothesis is rejected, the result is *positive*. When the test result tallies with the actual reality, the result is *true*. The false-negative rate of the test, that is, the fraction of *positive* instances that were falsely reported as *negative*, is denoted by β. The sensitivity (or power) of the test is the true-positive rate $(1 - \beta)$. The significance of the test is the false-positive rate (α) and the specificity of the test is the true-negative rate $(1 - \alpha)$. Let $\chi^2_{1-\alpha,k-1}$ denote the $(1 - \alpha)$-th lower quantile of a χ^2 distribution with $k - 1$ degrees of freedom. In a χ^2 test, H_0 is rejected (in other words, H_1 is accepted), if $Q > \chi^2_{1-\alpha,k-1}$ with 100α % error. We denote $\chi^2_{1-\alpha,k-1}$ simply as $\chi^2_{1-\alpha}$ when $k - 1$ is clear from the context.

In our approach, N is the number of plaintexts - the parameter to be determined. Let E' denote the experiment E repeated N times. To minimise error, we consider q randomly generated keys and E' is performed q times. We could estimate the mean and variance of the χ^2 values for the entire key space using the Student's t-distribution. But this requires that the population be normally distributed. This is nearly achieved when the number of degrees of freedom $(k - 1)$ is large since when $k \to \infty$, the χ^2 variate becomes a normal variate. Finally, using q, the q-sample mean and sample variance, a confidence interval (CI) is computed, using the t-curve, for the mean of the population. We use 90% confidence interval in our tests. In other words, the chance that the population mean falls below (or above) the interval is 5%. The lower end point of the interval ($minCI$) is taken for the population mean. This means that there is 95% chance that the actual population mean is above this value. In our experiments, we accept H_1 if $minCI$ is greater than $\chi^2_{1-\alpha,k-1}$. This automatically implies that the actual population mean is greater than $\chi^2_{1-\alpha,k-1}$ with 95% probability and thus, the error is small.

In this paper, we use the χ^2 test under the following settings (where XRC6 denotes RC6, MRC6 or ERC6 and $r > 0$):

H_0: a subset of bits output by r-round XRC6 is uniformly distributed,
H_1: a subset of bits output by r-round XRC6 is non-uniformly distributed.

Thus, (1) becomes

$$Q = \sum_{i=1}^{k} \frac{(n_i - N/k)^2}{N/k}. \tag{2}$$

A requirement in χ^2 tests is that $N \geq 5 \cdot k$, so that the computed χ^2 value is valid. In conventional χ^2 cryptanalysis, most of the plaintext bits are generated at random. However, plaintexts can be chosen in the following way to yield more

efficient attacks. Initially, a linear analysis (LC) is performed to determine which z least significant bits (lsb) of d words, in an n-bit block are linearly correlated to the same set of bits after a certain number r of rounds. This approach of using LC results prior to the χ^2 analysis has already been adopted in [13]. For RC6, $d = 2$, $z \leq 5$, $n = 128$ and r is multiple of 2, as indicated by (3) in Sect. 4.1. This set of $d \cdot z$ plaintext bits will be fixed (to an arbitrary value), while the remaining $n - d \cdot z$ plaintext bits are free to vary. These two sets of bits are disjoint. These plaintexts are encrypted across r rounds, and the χ^2 value is computed for the $d \cdot z$ ciphertext bit positions given by the linear relation. If the resulting χ^2 value supports acceptance of H_0, then we stop, record the number N of plaintexts encrypted so far, and proceed the same analysis y rounds farther (in this paper, $y = 2$). Otherwise, we consider the remaining $n - d \cdot z$ plaintext bits as a counter, increment it, and encrypt the corresponding plaintext for r rounds. The number of degrees of freedom is $k - 1 = 2^{d \cdot z} - 1$. We look for the minimum N for which H_1 is accepted. Each test is repeated q times; we use $q = 20$. The following pseudocode describes the overall procedure.

TEST $(H_0, H_1, N, r, q, \alpha)$
(1.)   for $(i = 1; i \leq q; i + +)$ {
(2.)   for $(j = 1; j \leq N; j + +)$ {
(3.)   fix the given set of $d \cdot z$ bits of plaintext P_j
(4.)   vary the remaining bits of P_j incrementally
(5.)   encrypt P_j through r rounds and obtain C_j
(6.)   let X be the concatenation of given $d \cdot z$ bits of C_j
(7.)   increment counter $T[X]$ by 1
(8.)   }
(9.)   let Q_i be the χ^2 value of $T[X]$'s
(10.)   }
(11.)   let m be the average over all Q_i, $1 \leq i \leq q$
(12.)   let σ be the standard deviation over all Q_i, $1 \leq i \leq q$
(13.)   let $minCI = m - 1.729 \cdot \sigma / \sqrt{q}$ (lower limit of a 90% CI)
(14.)   let $\chi^2_{1-\alpha,k-1}$ = value at $100(1 - \alpha)\%$ in the χ^2 cumulative distribution
            with $k - 1$ degrees of freedom
(15.)   if $(minCI > \chi^2_{1-\alpha,k-1})$
(16.)   choose H_1 and note the j corresponding to N
(17.)   else choose H_0

For our target ciphers, a further consequence of the new approach is a smaller demand for chosen plaintexts, due to weak diffusion. As already pointed out in [13], too small or too large rotation amounts lead to weak diffusion across multiple rounds of RC6. The same phenomenon can be observed in ERC6 and MRC6. This is an essential weakness exploited in our attacks since the linear relations (3), (4) and (5), which indicate the $d \cdot z$ bits in lines (3.) and (6.) of TEST(), rely on these assumptions. A smaller number of plaintexts implies a smaller encryption time, and thus, faster attacks. It shall be observed that the more bits are under analysis, the better the attack outcome. Nonetheless, the data (and

time) complexities increased quickly beyond our computational resources. Consequently, we used different value of z for the plaintext and ciphertext, unlike in TEST() where z is identical for plaintext and ciphertext (here, we have followed the approach of [12,14]).

Our attacks on RC6 and the approach used in [13] are different. We fix a number of bits to zeros and vary the remaining bits incrementally; whereas in the latter, the remaining bits are random. The result is that, with Knudsen and Meier's method, one can turn the CP setting into a KP one at the cost of a factor of $2^{d \cdot z}$ in the data and time complexities. Secondly, we used 90% confidence interval (CI) to minimise error, whereas [13] did not use CI.

3 The RC6, ERC6 and MRC6 Families of Block Ciphers

Initially, we provide some relevant notations: '$\oplus$' denotes bitwise exclusive-OR; '$\boxplus$' denotes addition modulo 2^w; '$*$' denotes multiplication modulo 2^w; $x \lll y$, where x and y are w-bit words, means that x is cyclically shifted to the left by the amount given by least significant $\log_2 w$ bits of y. The function $F : \mathbb{Z}_2^w \to \mathbb{Z}_2^w$ is given by $F(X) = (2 * X^2 \boxplus X) \lll \log_2 w$. Notice that F has only one operand, and is a bijective mapping. Thus, it behaves as a $w \times w$-bit nonlinear S-box.

3.1 RC6

The RC6 cipher follows a generalized Feistel Network structure, and stands for a family of ciphers formally denoted RC6-$w/r/b$, where w is the word size in bits, r is the number of rounds, and b is the key size in bytes. For the AES competition, $w = 32$, $r = 20$, and $b \in \{16, 24, 32\}$, and RC6 is a shorthand for these parameter choices. All internal cipher operations are over w-bit words, where $w \in \{8, 16, 32, 64\}$. Fig. 1 depicts the RC6 encryption algorithm. Each text block contains four w-bit words. For instance, A_i, B_i, C_i, D_i, denote the input words to the i-th round. The w-bit round keys are indexed $S[0], \dots, S[2r + 3]$. The key schedule algorithm generates the round keys from the b-byte user key. We do not exploit the key schedule algorithm in our analysis; therefore, we omit its description and refer the interested reader to [1]. Former security analyses of RC6 include differential and linear analyses [1], multiple linear relations [17], and χ^2 analyses [9,13,1,15].

3.2 MRC6

The MRC6 cipher follows a generalized Feistel Network structure and was proposed in [4], with main focus on (software) performance. No security analysis was presented. MRC6 is a parameterized family of ciphers formally denoted MRC6-$w/r/b$, with the same meaning as for the parameters of RC6. But, nominal values of these parameters were omitted in [4]; one can find the values $w = 32$, $b = 16$ and $r = 16$ when the software performance of MRC6 is compared with that of the AES and RC6 (on Pentium-III, with the se parameters, MRC6 encrypts at about

19.5 MB/sec making it nearly twice as fast as RC6). Otherwise, these parameters are unspecified. The fact that these parameters are unrelated helps adapt MRC6 as a compression function in hash modes [18] such as Miyaguchi-Preneel and Matyas-Meyer-Oseas, where the key and text inputs have different sizes. An MRC6 text block contains sixteen w-bit words, denoted $A_i, B_i, \ldots, P_i$ as inputs to the i-th round. Moreover, the w-bit round keys are indexed $S[0], \ldots, S[8r+7]$. Like in RC6, there are pre-whitening and post-whitening layers. Here again, we omit the description of the key schedule algorithm and refer the reader to [4]. Fig. 3 depicts the MRC6 encryption algorithm. In our experiments, we use MRC6 with $w = 32$ and $b = 16$.

3.3 ERC6

The ERC6 cipher follows a generalized Feistel Network structure, and was proposed in [3], as a parameterized family of ciphers formally denoted ERC6-$w/r/b$, with $w \in \{16, 32, 64\}$, $r \in \{0, 1, 2, \ldots, 255\}$, $b \in \{0, 1, 2, \ldots, 255\}$. These parameters appear to be loosely coupled. No attacks have been reported on any version of ERC6. On Pentium-III, with parameters $w = 32$, $b = 16$ and $r = 16$, ERC6 encrypts at about 17.3 MB/sec making it about 1.7 times faster than RC6. Each text block of ERC6 contains eight w-bit words, denoted A_i, B_i, C_i, D_i, E_i, F_i, G_i, H_i, as inputs to the i-th round. The w-bit round keys are indexed $S[0], \ldots, S[4r+7]$. Here again, there are pre-whitening and post-whitening layers. Fig. 2 depicts the ERC6 encryption algorithm. In our experiments, we use ERC6 with $w = 32$ and $b = 16$.

4   Experimental Observations

Our χ^2 attacks operate in an adaptive chosen-plaintext (CP) setting.

4.1 Reduced-Round RC6

For RC6, the χ^2 test is motivated by an ensemble of linear relations involving up to the five least significant bits of words A_i and C_i for every two rounds [19]. These linear relations can be represented by

$$A_i \cdot e_{t_1} \oplus C_i \cdot e_{t_2} = A_{i+2} \cdot e_{t_3} \oplus C_{i+2} \cdot e_{t_4}, \tag{3}$$

where A_i and C_i denote the first and third input words to the i-th round. Each bitmask, $e_j = 2^j$, $0 \le j < 5$, contains only a single bit equal to one, in the j-th least significant bit ($j = 0$ denotes the lsb). This is the lowest possible Hamming weight. Table 1 shows the result of the experiment on reduced-round RC6 using our method in the case of ten bits: $lsb_5(A_{2i})$ and $lsb_5(C_{2i})$.

We use $\chi^2_{95} = 1098$ (95% specificity) to facilitate comparison, since [13] also uses the same threshold. Moreover, for the same comparison purpose, we did not use confidence intervals this time. In Table 1, note that with 2^2 texts we already

Table 1. χ^2 attack simulations on RC6, $2^{10} - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	hypothesis
2	2	1071.2	H_0
2	3	1169.6	H_1
2	4	1398.4	H_1
2	5	1561.6	H_1
2	6	2009.6	H_1
4	16	1039.5	H_0
4	17	1066.3	H_0
4	18	1094.2	H_0
4	19	1151.6	H_1
4	20	1267.6	H_1
6	32	1030.6	H_0
6	33	1036.0	H_0
6	34	1020.6	H_0
6	35	1018.1	H_0
6	36	1028.4	H_0
6	37	1009.6	H_0

start to reach the same results of [13], whereas they needed 2^{13} texts to arrive at a χ^2 value of 1098. For four rounds, we noticed very close approximations for the same χ^2 values with 2^{18} texts, while [13] required 2^{29} texts to arrive at data with the same specificity.

The experimental results for 2-round RC6 show that our approach requires only 2^3 texts to reach the same χ^2 value that is obtained with 2^{14} texts using the approach in [13]. For 4-round RC6, these figures are 2^{19} texts using our technique against 2^{30} for [13]. For 6-round RC6, our method required more than 2^{37} texts to detect correlation. In this case (and for more rounds), it could not be concluded whether our approach was better than [13].

4.2   MRC6

For MRC6, our χ^2 attacks were motivated by the following 2-round iterative linear relation (using Type-I approximations [19])

$$A_i \cdot e_{t_1} \oplus C_i \cdot e_{t_2} \oplus E_i \cdot e_{t_3} \oplus G_i \cdot e_{t_4} \oplus I_i \cdot e_{t_5} \oplus K_i \cdot e_{t_6} \oplus M_i \cdot e_{t_7} \oplus O_i \cdot e_{t_8} =$$
$$A_{i+2} \cdot e_{t_9} \oplus C_{i+2} \cdot e_{t_{10}} \oplus E_{i+2} \cdot e_{t_{11}} \oplus G_{i+2} \cdot e_{t_{12}} \oplus$$
$$I_{i+2} \cdot e_{t_{13}} \oplus K_{i+2} \cdot e_{t_{14}} \oplus M_{i+2} \cdot e_{t_{15}} \oplus O_{i+2} \cdot e_{t_{16}} \tag{4}$$

where A_i, C_i, E_i, G_i, I_i, K_i, M_i and O_i are input words to the i-th round. In particular, the masks e_j with highest bias are such that $0 \le j < 5$, that is, the bits in the masks are restricted to the five least significant bit (lsb) positions. Our experiments distinguish r rounds of MRC6 from a random permutation, where r is even. We fix up the $8 \cdot \log_2 w$ least significant bits of words A_0, C_0, E_0, G_0, I_0, K_0, M_0 and O_0 (that is, including the pre-whitening), and analyse

Table 2. χ^2 attack simulations on MRC6, $2^8 - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	1	305.2	264.6	4	H_0
2	2	444.0	376.8	18	H_1
2	3	667.2	622.9	20	H_1
4	3	299.2	262.8	8	H_0
4	4	321.6	303.0	19	H_1
4	5	368.8	353.7	20	H_1
6	8	265.8	256.7	8	H_0
6	9	282.1	270.6	11	H_1
6	10	288.9	276.7	15	H_1
6	11	316.5	298.1	16	H_1
6	12	370.8	343.3	20	H_1
8	18	273.0	265.0	11	H_0
8	19	288.1	279.8	16	H_1
8	20	327.5	311.1	20	H_1
10	27	273.5	265.4	11	H_0
10	28	288.2	275.7	12	H_1
10	29	290.2	278.2	15	H_1
10	30	336.2	317.2	19	H_1
10	31	402.7	368.0	19	H_1
10	32	407.8	357.0	19	H_1
10	33	434.1	374.1	20	H_1

the combined $8 \cdot y$ least significant bits ($y \in \{1, 2\}$) of A_{2i}, C_{2i}, E_{2i}, G_{2i}, I_{2i}, K_{2i}, M_{2i} and O_{2i}, for $i > 0$, that is, after an even number of rounds.

Table 2 shows the result of the experiment in the case of the eight bits: $\mathrm{lsb}_1(A_{2i})$, $\mathrm{lsb}_1(C_{2i})$, $\mathrm{lsb}_1(E_{2i})$, $\mathrm{lsb}_1(G_{2i})$, $\mathrm{lsb}_1(I_{2i})$, $\mathrm{lsb}_1(K_{2i})$, $\mathrm{lsb}_1(M_{2i})$, $\mathrm{lsb}_1(O_{2i})$. We use $\chi^2_{75} = 269.85$. Starting from six rounds, the number of texts for which H_0 is rejected starts to increase by a factor of about 2^{10} every two rounds. Thus, for r rounds (r even and $r \geq 6$), the following is expected for N (numner of chosen plaintexts) in terms of r: $N = 2^9 \cdot 2^{10 \cdot (r-6)/2} = 2^{5r-21}$. In TEST(), we choose plaintexts such that the $\mathrm{lsb}_5(A_0)$, $\mathrm{lsb}_5(C_0)$, $\mathrm{lsb}_5(E_0)$, $\mathrm{lsb}_5(G_0)$, $\mathrm{lsb}_5(I_0)$, $\mathrm{lsb}_5(K_0)$, $\mathrm{lsb}_5(M_0)$, $\mathrm{lsb}_5(O_0)$ are set to zero, while the remaining bits are changed incrementally. This implies at most $2^{512-40} = 2^{472}$ plaintext blocks are available. Thus, we require $2^{5r-21} \leq 2^{472}$, or $5r \leq 493$, or $r \leq 98$. The data complexity is at most 2^{472} plaintext blocks. It means that MRC6 would require at least $r = 99$ rounds to counter this χ^2 attack.

Table 3 shows the result of the experiment in the case of 16 bits: $\mathrm{lsb}_2(A_{2i})$, $\mathrm{lsb}_2(C_{2i})$, $\mathrm{lsb}_2(E_{2i})$, $\mathrm{lsb}_2(G_{2i})$, $\mathrm{lsb}_2(I_{2i})$, $\mathrm{lsb}_2(K_{2i})$, $\mathrm{lsb}_2(M_{2i})$, and $\mathrm{lsb}_2(O_{2i})$ after an even number of rounds of MRC6. We use $\chi^2_{75} = 65779$. Starting from six rounds, the number of texts for which H_0 is rejected starts to increase by a factor of about 2^{10} every two rounds. Thus, for r rounds (r even and $r \geq 6$), the following is expected for the number of chosen plaintexts in terms of r: $N = 2^{12} \cdot 2^{10 \cdot (r-6)/2} = 2^{5r-18}$. The analysis is similar to the 8-bit case in the previous paragraph. Following the same rationale, at most $2^{512-40} = 2^{472}$ plaintext blocks

Table 3. χ^2 attack simulations on MRC6, $2^{16} - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	1	68810.8	63145.2	1	H_0
2	2	80277.6	72615.5	8	H_1
2	3	90923.2	84597.2	18	H_1
2	4	125731.2	116640.0	20	H_1
4	4	65520.0	65520.0	0	H_0
4	5	66732.8	65828.2	5	H_1
4	6	67622.4	66920.0	14	H_1
4	7	69913.6	68843.6	19	H_1
4	8	74035.2	72727.0	20	H_1
6	11	65804.8	65643.3	12	H_0
6	12	66108.8	65941.0	16	H_1
6	13	66850.4	66685.1	20	H_1
8	20	65804.6	65648.0	11	H_0
8	21	66090.0	65912.8	15	H_1
8	22	66862.9	66608.5	19	H_1
8	23	68275.1	67872.4	20	H_1
10	30	65637.4	65450.0	9	H_0
10	31	65916.7	65760.9	14	H_1
10	32	65961.5	65778.7	11	H_1
10	33	66128.2	65961.9	16	H_1
10	34	66521.0	66262.4	18	H_1
10	35	67043.6	66671.7	19	H_1

will be available. Thus, this analysis holds as long as $2^{5r-18} \leq 2^{472}$, or $5r \leq 490$, or $r \leq 98$. Again, the data complexity is at most 2^{472} plaintext blocks, and MRC6 requires at least 99 rounds to counter this χ^2 attack.

In order to compare the approach in Table 2 with an alternative approach used in [13], we provide Table 4.

Experimentally, we have observed that less chosen plaintexts are needed in the new approach than in the conventional approach of [13], at least for two, four and six rounds.

We point out that in Tables 2 and 3, the minimum value of N for which H_1 is accepted may be less than $5 \cdot k$ when the number of rounds is small. For example, the values of N for 2, 4 and 6 rounds in Table 2. This phenomenon is particular for a small number of rounds, and is due to the large block size and the slow diffusion in MRC6 (unlike the AES, in which diffusion is guaranteed by an MDS matrix, in MRC6 the diffusion depends on appropriate rotation amounts). Therefore, we also use these former values of N to estimate the minimum N for which H_1 is accepted, for higher numbers of rounds. For 8 or more rounds, the (minimum) values for N are greater than $5 \cdot k$.

4.3 ERC6

For ERC6, our χ^2 attacks were guided by the following 2-round iterative linear relation (using Type-I approximations [19])

Table 4. χ^2 attack simulations on MRC6 using the approach in [13] with $2^8 - 1$ degrees of freedom

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	21	268.0	262.1	12	H_0
2	22	276.8	271.4	11	H_1
2	23	293.3	288.9	18	H_1
2	24	326.4	320.1	20	H_1
4	29	256.9	250.6	8	H_0
4	30	251.1	245.8	4	H_0
4	31	257.5	252.1	8	H_0
4	32	258.7	254.0	5	H_0
4	33	255.6	251.1	4	H_0
6	25	265.5	258.4	9	H_0
6	26	258.7	253.2	8	H_0
6	27	253.0	247.8	6	H_0
6	28	251.3	246.2	4	H_0
6	29	254.9	248.6	6	H_0

Table 5. χ^2 attack simulations on ERC6, $2^4 - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	1	18.0	15.25	5	H_0
2	2	26.4	22.44	8	H_1
2	3	37.8	34.20	20	H_1
4	12	26.9	22.08	10	H_0
4	13	38.7	28.95	14	H_1
4	14	65.8	43.67	17	H_1
4	15	120.8	78.61	18	H_1
4	16	222.5	138.04	19	H_1
4	17	446.8	276.82	20	H_1
6	23	26.0	20.23	10	H_0
6	24	37.9	27.31	13	H_1
6	25	59.4	42.71	16	H_1
6	26	99.0	68.65	17	H_1
6	27	196.2	133.90	18	H_1
6	28	375.3	258.18	20	H_1

$$A_i \cdot e_{t_1} \oplus C_i \cdot e_{t_2} \oplus E_i \cdot e_{t_3} \oplus G_i \cdot e_{t_4} = A_{i+2} \cdot e_{t_5} \oplus C_{i+2} \cdot e_{t_6} \oplus E_{i+2} \cdot e_{t_7} \oplus G_{i+2} \cdot e_{t_8}, \quad (5)$$

where A_i, C_i, E_i and G_i, are input words to the i-th round. In particular, the masks e_j with highest bias are such that $0 \le j < \log_2 w$.

Table 5 shows the result of attack simulation in the case of 4 bits: $\mathrm{lsb}_1(A_{2i})$, $\mathrm{lsb}_1(C_{2i})$, $\mathrm{lsb}_1(E_{2i})$ and $\mathrm{lsb}_1(G_{2i})$ after an even number of rounds of ERC6. We use $\chi^2_{75} = 22.31$. Starting from four rounds, the number of texts for which H_0 is rejected starts to increase by a factor of about 2^{11} every two rounds. Thus, for r rounds (r even and $r \ge 4$), the following is expected for the number

Table 6. χ^2 attack simulations on ERC6 using the approach in [13] with $2^4 - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	15	17.1	14.1	8	H_0
2	16	26.8	22.9	16	H_1
2	17	35.2	31.3	20	H_1
4	32	14.833	12.03	6	H_0
4	33	16.633	13.87	6	H_0
4	34	14.549	12.56	5	H_0
4	35	15.257	13.20	4	H_0
4	36	13.143	11.25	2	H_0

Table 7. χ^2 attack simulations on ERC6, $2^8 - 1$ degrees of freedom and 20 tests

#Rounds	$\log_2 N$	average χ^2	minCI	#values> χ^2_{75}	hypothesis
2	2	309.6	272.0	7	H_0
2	3	356.8	316.6	15	H_1
2	4	520.0	444.0	20	H_1
4	11	284.5	270.3	9	H_0
4	12	310.0	291.0	14	H_1
4	13	366.8	334.6	17	H_1
4	14	468.0	405.2	18	H_1
4	15	711.6	579.7	20	H_1
6	23	290.5	283.5	14	H_0
6	24	325.6	311.9	18	H_1
6	25	404.5	379.9	19	H_1
6	26	549.3	491.7	20	H_1

of chosen plaintexts: $N = 2^{13} \cdot 2^{11 \cdot (r-4)/2} = 2^{5.5r-9}$. The algorithm TEST(.) chooses plaintexts such that the $\mathrm{lsb}_5(A_0)$, $\mathrm{lsb}_5(C_0)$, $\mathrm{lsb}_5(E_0)$, $\mathrm{lsb}_5(G_0)$ are set to zero. This implies at most $2^{256-20} = 2^{236}$ plaintext blocks are available. Thus, this analysis holds as long as $2^{5.5r-9} \leq 2^{236}$, or $5.5r \leq 245$, or $r \leq 44$. Since the attack effort is at most 2^{236} encryptions equivalent number of text blocks, it means that ERC6 would require at least 45 rounds to counter this χ^2 attack.

In order to compare the approach in Table 5 with the approach used in [13], we provide Table 6. Empirically, we have observed that significantly less chosen plaintexts are needed in the new approach than in the conventional approach of [13], at least for two and four rounds.

Table 7 shows the result of analysing the 8-bit value from the concatenation of $\mathrm{lsb}_2(A_{2i})$, $\mathrm{lsb}_2(C_{2i})$, $\mathrm{lsb}_2(E_{2i})$ and $\mathrm{lsb}_2(G_{2i})$ after an even number of rounds of ERC6. We use $\chi^2_{75} = 284.34$. Starting from four rounds, the number of texts for which H_0 is rejected starts to increase by a factor of about 2^{12} every two rounds. Thus, for r rounds (r even and $r \geq 4$), the following behaviour is expected for the number of chosen plaintexts: $N = 2^{12} \cdot 2^{12 \cdot (r-4)/2} = 2^{6r-12}$. Following a

similar reasoning as in the previous paragraph, this analysis holds as long as $2^{6r-12} \leq 2^{236}$, or $6r \leq 248$, or $r \leq 41$. Since the attack effort is at most 2^{236} encryption, ERC6 requires at least 42 rounds to counter this χ^2 attack.

5   Conclusions and Further Work

This paper presented a new approach to the χ^2 statistical test applied to RC6, ERC6 and MRC6 block ciphers. These attacks were preceeded by a linear crypt-analysis of these same ciphers, which provided promising bit positions to be analysed by the χ^2 tests. For 2-round and 4-round RC6, our method improves the data complexity of the previously best-known χ^2 attacks [13] by a factor of about 2^{11}. Tables 8, 9, 10 and 11 summarize our attacks on ERC6 and MRC6.

Overall, our attacks reduced the number of chosen plaintexts to detect χ^2 correlation when compared to conventional χ^2 attacks. Consequently, we could apply and check in practice our predictions on attacks up to 10-round MRC6

Table 8. Summary of χ^2 attacks analysing 8 bits output by MRC6

#Rounds	Time	Data	Memory	Comment
2	2^2	2^2 CP	2^2	Table 2
4	2^4	2^4 CP	2^4	Table 2
r	2^{5r-21}	2^{5r-21} CP	2^{5r-21}	$6 \leq r < 99$, r even

Table 9. Summary of χ^2 attacks analysing 16 bits output by MRC6

#Rounds	Time	Data	Memory	Comment
2	2^2	2^2 CP	2^2	Table 3
4	2^5	2^5 CP	2^5	Table 3
r	2^{5r-18}	2^{5r-18} CP	2^{5r-18}	$6 \leq r < 99$, r even

Table 10. Summary of χ^2 attacks analysing 4 bits output by ERC6

#Rounds	Time	Data	Memory	Comment
2	2^2	2^2 CP	2^2	Table 5
4	2^{13}	2^{13} CP	2^{13}	Table 5
r	$2^{5.5r-9}$	$2^{5.5r-9}$ CP	$2^{5.5r-9}$	$4 \leq r < 45$, r even

Table 11. Summary of χ^2 attacks analysing 8 bits output by ERC6

#Rounds	Time	Data	Memory	Comment
2	2^3	2^3 CP	2^3	Table 7
4	2^{11}	2^{11} CP	2^{11}	Table 7
r	2^{6r-12}	2^{6r-12} CP	2^{6r-12}	$4 \leq r < 42$, r even

and 6-round ERC6. The reduction in the data complexity of our attacks was influenced by the weak diffusion in the target ciphers.

In the analyses of M6, MX and RC5P in [5], the χ^2 tests were supported by evidence collected from mod-n analyses of these ciphers. The nonuniform distribution of residues modulo n of internal cipher components, for n a prime number, was corroborated by experimental χ^2 tests. Likewise, in this paper, our results were supported by linear relations.

The analyses presented in this paper considered sets of randomly chosen keys, that is, no particular (weak) keys were purposefully used. This implies that even better results could have been achieved with keys that caused null rotation in some rounds under analysis (as observed in [13]). This issue of weak keys for χ^2 attacks is left as a problem for future work. Analogously, we have focused on distinguishing attacks only. Key-recovery attacks are also left as further work.

References

1. Rivest, R.L., Robshaw, M.J.B., Sidney, R., Yin, Y.L.: The RC6 block cipher (1998),
   http://csrc.nist.gov/encryption/aes/
2. AES: The advanced encryption standard development process (1997),
   http://csrc.nist.gov/encryption/aes/
3. Ragab, A., Ismail, N., Allah, O.F.: Enhancements and implementation of RC6 block cipher for data security. In: IEEE TENCON (2001)
4. El-Fishawy, N., Danaf, T., Zaid, O.: A modification of RC6 block cipher algorithm for data security (MRC6). In: International Conference on Electrical, Electronic and Computer Engineering, pp. 222–226 (2004)
5. Kelsey, J., Schneier, B., Wagner, D.: Mod n cryptanalysis, with applications against RC5P and M6. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 139–155. Springer, Heidelberg (1999)
6. Knuth, D.: The Art of Computer Programming, Seminumerical Algorithms. Addison-Wesley, Reading (1997)
7. Vaudenay, S.: An experiment on DES statistical cryptanalysis. Technical report, Ecole Normale Supérieure, LIENS–95–29 (1995)
8. Handschuh, H., Gilbert, H.: χ^2 cryptanalysis of the SEAL encryption algorithm. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 1–12. Springer, Heidelberg (1997)
9. Baudron, O., Gilbert, H., Granboulan, L., Handschuh, H., Joux, A., Nguyen, P., Noilhan, F., Pointcheval, D., Pornin, T., Poupard, G., Stern, J., Vaudenay, S.: Report on the AES candidates (1999),
   http://csrc.nist.gov/encrypt/aes/round1/conf2/papers/baudron1.pdf
10. Gilbert, H., Handschuh, H., Joux, A., Vaudenay, S.: A statistical attack on RC6. In: Schneier, B. (ed.) FSE 2000. LNCS, vol. 1978, pp. 64–74. Springer, Heidelberg (2001)
11. Hinoue, T., Miyaji, A., Wada, T.: The security of RC6 against asymmetric chi-square test attack. IPSJ Journal 48(9), 1–10 (2007)
12. Isogai, N., Matsunaka, T., Miyaji, A.: Optimized χ^2-attack against RC6. In: Zhou, J., Yung, M., Han, Y. (eds.) ACNS 2003. LNCS, vol. 2846, pp. 16–32. Springer, Heidelberg (2003)

13. Knudsen, L., Meier, W.: Correlations in RC6 with a reduced number of rounds. In: Schneier, B. (ed.) FSE 2000. LNCS, vol. 1978, pp. 94–108. Springer, Heidelberg (2001)
14. Nonaka, M., Miyaji, A.: A note on the security of RC6 against correlation attack. In: The 2002 Symposium on Cryptography and Information Security, pp. 681–686 (2002)
15. Shimoyama, T., Takeuchi, K., Hayakawa, J.: Correlation attack to the block cipher RC5 and the simplified variants of RC6 (2001), http://csrc.nist.gov/encryption/aes/
16. Bain, L., Engelhardt, M.: Introduction to Probability and Mathematical Statistics. Duxbury Press (1987)
17. Shimoyama, T., Takenaka, M., Koshida, T.: Multiple linear cryptanalysis of a reduced-round RC6. In: The 2002 Symposium on Cryptography and Information Security, pp. 931–936 (2002)
18. Menezes, A., van Oorschot, P., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1997)
19. Contini, S., Rivest, R., Robshaw, M., Yin, Y.: The security of the RC6 block cipher, v. 1.0 (1998)

A   Tables and Figures

**Table 12.** χ^2 threshold values, specificities and degrees of freedom

χ^2	Degrees of Freedom $(k-1)$					
	2^4-1	2^8-1	$2^{12}-1$	$2^{16}-1$	$2^{20}-1$	$2^{24}-1$
0.60	15.73	260.09	4117.30	65626.10	1048941.26	16778682
0.70	17.32	266.34	4141.97	65724.37	1049333.93	16780252
0.75	18.24	269.85	4155.67	65778.82	1049551.40	16781122
Specificity 0.80	19.31	273.79	4170.96	65839.50	1049793.60	16782090
$(1-\alpha)$ 0.85	20.60	278.43	4188.84	65910.27	1050075.96	16783219
0.90	22.31	284.34	4211.40	65999.39	1050431.31	16784639
0.95	24.99	293.25	4244.99	66131.63	1050958.14	16786744
0.99	30.58	310.46	4308.47	66380.16	1051946.85	16790690
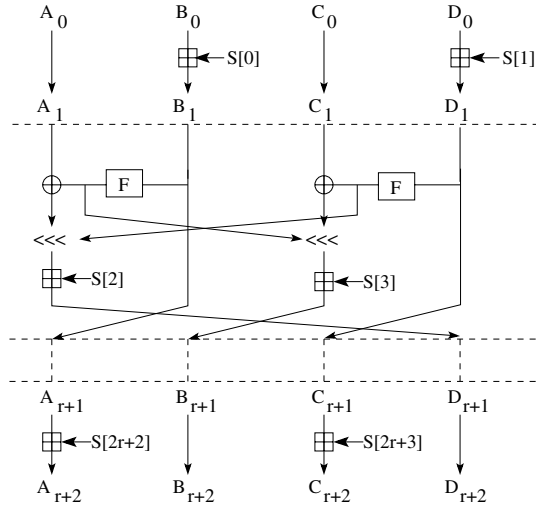
Fig. 1. Computational graph of the RC6 block cipher for encryption, showing pre-whitening, one full round and post-whitening
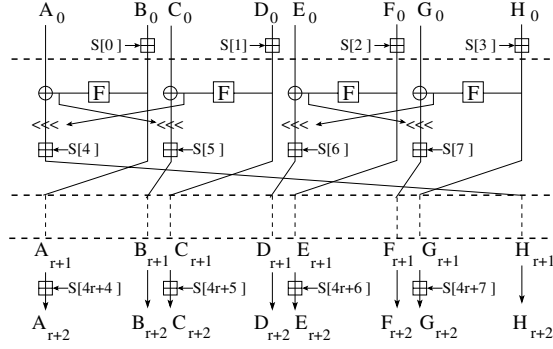
Fig. 2. Computational graph of the ERC6 block cipher for encryption, showing pre-whitening, one full round and post-whitening
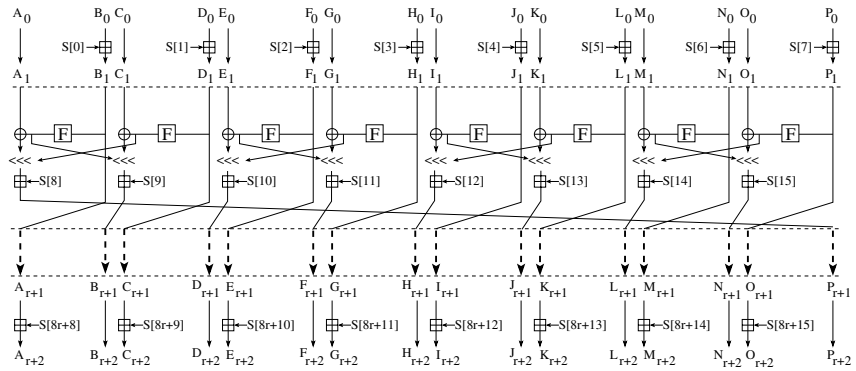
Fig. 3. Computational graph of the MRC6 block cipher for encryption, showing pre-whitening, one full round and post-whitening