The Probability Advantages of Two Linear Expressions in Symmetric Ciphers

Haina Zhang [*] Shaohui Wang [†] Xiaoyun Wang [‡]

Abstract

In this paper, we prove the probability advantages of two linear expressions which are summarized from the ABC stream cipher submitted to ECRPYT Estream Project. Two linear expressions with probability advantages reflect the linear correlations among Modular Addition equations. Corresponding to each linear expression and its advantage, a large amount of weak keys are derived under which all the ABC main keys can be retrieved successively. The first linear expression is a generic bit linear correlation between two Modular Addition equations. The second is a linear correlation of bit carries derived from three Modular Addition equations and the linear equation of LFSR in ABC.

It is remarked that the second is found by Wu and Preneel, and has been used to find 2^{96} weak keys. In the cryptanalysis of ABC, Wu and Preneel only utilized its estimated probability advantage which is concluded by experimental data, and they did not give its strict proof.

Modular Addition and XOR operations are widely used in designing symmetric ciphers. We believe that these types of linear expressions with probability advantages not only can be used to analyze some other symmetric ciphers, but also are important criteria in designing secure symmetric ciphers.

Key Words. Cryptanalysis, probability advantage, Modular Addition, ABC stream cipher.

[*] Shandong University, China. Email:hnzhang@math.sdu.edu.cn.

[†] Shandong University, China. Email:shwang@math.sdu.edu.cn.

[‡] Shandong University and Tsinghua University, China. Email:xywang@sdu.edu.cn.

1 Introduction

For a secure cryptographic algorithm, there should not exist any mathematic rule with substantial probability advantage. For example, among the ciphertexts and the transmitting data, there should be no statistical property or no algebraic expression whose probability is high enough to result in an attack faster than the brute force attack. The probability advantage means the probability bias between the real probability and $\frac{1}{2}$.

In the view of cryptanalysis history, cryptanalysis techniques based on the probability advantage are the most important not only in designing the provable security cryptosystems, but also in analyzing the security of symmetric ciphers.

Most provable security cryptosystems employ a similar proof model based on probability advantage. One type of earlier provable security cryptosystems is the random generator models proposed by Yao [12], Goldwasser, Micali[6] etc, and another type is zero-knowledge schemes [5]. These cryptosystems adopt a common probabilistic proof model based on the indistinguishable advantage, i.e. all the ciphertexts and the transmitting data only have an indistinguishable advantage for their statistical properties, otherwise, an adversary can find an available attack to solve the hard mathematic problem on which the cryptosystems are constructed. Here indistinguishability usually means polynomial time indistinguishability.

In symmetric cipher cryptanalysis, cryptanalysts try to find the probability advantages of some related statistical properties and algebraic expressions. For example, the correlation attack and the linear syndrome method are two important analysis methods on stream ciphers respectively developed by two groups, Meier and Staffelbach [10], Zeng and Huang[13]. The two methods are based on the probability advantage of some linear expressions extracted from LFSRs. The differential attack of block ciphers is formalized by Biham and Shamir [4], which fully utilizes the probability advantage of some special differential. Another important attack in block ciphers is linear cryptanalysis [9] which uses the probability advantages of some linear equations with plaintext bits, ciphertext bits and key bits.

In this paper, we prove the probability advantages of two linear expressions by analyzing a synchronous stream cipher ABC which is an ECRYPT candidate for stream cipher standard. ABC is designed by Anashin, Bogdanov, Kizhvatov and Kumar, and has two versions. The first version ABC v1 [1] had been withdrawn because of the attacks proposed in [3, 7, 8]. The second version ABC v2 [2] adopted a longer LFSR to avoid of the attack. There is no substantial attack until the weak key attack by Wu and Preneel in Feb 2006 [11].

In [11], Wu and Preneel found a linear expression which has a substantial probability advantage by considering the LFSR equation and the bit carries of three Modular Addition equations. In the cryptanalysis of ABC, Wu and Preneel utilized its estimated probability advantage which is concluded by experimental data, but they did not give its strict proof.

In this paper, we present the strict theoretical proof for the probability advantages of two linear expressions, one of which is presented in [11], another is a generic bit linear expression coming from two Modular Addition equations. Each linear expression can be used to find a large number of weak keys under which all the ABC main keys can be retrieved successively.

Modular Addition and XOR operations are popular in cryptography, especially in designing block ciphers, stream ciphers and hash functions. We believe that these linear expressions with probability advantages can not only be used to analyze some other symmetric ciphers, but also are important criteria in designing secure symmetric ciphers.

The paper is organized as follows: In section 2, a brief description of the ABC v2 is introduced. In section 3, a generic linear expression that has a probability advantage is shown. Section 4 gives the strict proof for the Wu-Preneel linear expression and expands it to the general constrained linear expression. Section 5 concludes the paper.

2 A Brief Description of ABC v2

ABC is a synchronous stream cipher optimized for software applications. It uses a 128-bit key and a 128$-$bit IV and claims a security of 2^{128}. Throughout this paper the symbols $\oplus$, $\gg$, $\ll$, $\ggg$, $+$ are respectively used for 32$-$bit XOR, right shift, left shift, right rotation and addition modulo 2^{32}.

ABC consists of three components:

1. Component A is a linear transformation of the 128$-$bit LFSR, and its characteristic polynomial is $f(x) = x(x^{127} + x^{63} + 1)$. The internal state is represented by $Z = (z_3, z_2, z_1, z_0)$, $z_i \in GF(2^{32})$, $0 \leq i < 4$. At each clock, the LFSR is updated as follows:

$$Temp \leftarrow z_2 \oplus (z_1 << 31) \oplus (z_0 >> 1)$$

$$z_0 \leftarrow z_1, \quad z_1 \leftarrow z_2, \quad z_2 \leftarrow z_3, \quad z_3 \leftarrow Temp.$$

2. Component B is the following single cycle T-function where d_0, d_1, $d_2 \in GF(2^{32})$ represent 32-bit key and IV dependent words.

   $$B(x) = ((x \oplus d_0) + d_1) \oplus d_2 \ (mod\ 2^{32})$$

   where $d_0 \equiv 0$, $d_2 \equiv 0$, $d_1 \equiv 1$ (mod 4).

3. Component C is a mapping from $GF(2^{32})$ to $GF(2^{32})$ which involves key dependent constant 32-bit words e, e_i $(0 \leq i < 32)$. $\delta_i(x)$ is the i-th bit selection function that determines the i-th bit of a 32-bit integer number, and the least significant bit is 0-bit.

   $$C(x) = ((e + \sum_{i=0}^{31} e_i \delta_i(x))\ mod\ 2^{32}) >>> 16$$

   $$e_{31} \equiv 2^{16} (mod\ 2^{17})$$

The key stream generation of ABC v2 involves the three components, the output key stream word is $y = z_0 + C(x), y \in GF(2^{32})$.

3 One Generic Linear Expression in Modular Addition Equations

In order to strengthen the avalanche of symmetric ciphers, cryptographers often adopt the mixture operations. In many ciphers, Modular Addition operation and XOR operation are often used alternatively. ABC is such a stream cipher that the two consist of the main operations. In order to retrieve the state of LFSR bit by bit, we need to find some XOR linear expressions related to LFSR which have probability bias. According to the equation $y = z_0 + C(x)$, it is necessary to employ some linear expressions on y and z_0, and especially, to search for the linear expression with a substantial advantage corresponding to special $C(x)$ which will results in weak keys. These types of linear expressions are extracted from the Modular Addition equations. So, we first discuss a generic linear expression without constraint conditions.

Definition 1. Denote y as n-bit integer(n>0), then $F(y, m)$ is defined as follows:

$$F(y, m) = \bigoplus_{i=0}^{m-1} \delta_i(y)$$

where $1 \leq m \leq n$.

Clearly, $F(y, m)$ is the XOR of m consecutive m bits of y.

Theorem 1. Denote y, c and x as n-bit (n>0) integers, suppose $y = c + x \ (mod\ 2^n)$. Denote $p_m = Pr(F(y, m+1) = F(c, m+1) \oplus F(x, m+1))$, $m = 0, 1, 2, \ldots$. Then

$$p_m = \frac{1}{2} + \frac{1}{2^{1+\lfloor \frac{m+1}{2} \rfloor}}.$$

Proof. Suppose $x_1 = x \ (mod\ 2^{m+1})$, $c_1 = c \ (mod\ 2^{m+1})$. We divide the pair (c_1, x_1) into 4 cases:

$SC(m)$ is the number of the pairs (c_1, x_1) that satisfy $F(y, m+1) = F(c_1, m+1) \oplus F(x_1, m+1)$ with the bit carry to $m+1$-th bit(take count from 0), i.e.

$$SC(m) = |\{(c_1, x_1) | F(y, m{+}1) = F(c_1, m{+}1) \oplus F(x_1, m{+}1), c_1{+}x_1 \geq 2^{m+1}\}|$$

Similarly,

$$SM(m) = |\{(c_1, x_1) | F(y, m{+}1) = F(c_1, m{+}1) \oplus F(x_1, m{+}1), c_1{+}x_1 < 2^{m+1}\}|$$

$$DC(m) = |\{(c_1, x_1) | F(y, m{+}1) \neq F(c_1, m{+}1) \oplus F(x_1, m{+}1), c_1{+}x_1 \geq 2^{m+1}\}|$$

$$DM(m) = |\{(c_1, x_1) | F(y, m{+}1) \neq F(c_1, m{+}1) \oplus F(x_1, m{+}1), c_1{+}x_1 < 2^{m+1}\}|$$

Clearly,

$$SC(m) + SM(m) + DC(m) + DM(m) = 2^{2(m+1)} \tag{1}$$

Each number has a recursion, and four recursions are dependent. We only show the recursion about $SM(m)$.

If the m-th bit of (x, c) is (0,1) or (1,0), then $SM(m) = 2SM(m-1)$. If the m-th bit of (x, c) is (0,0), then $SM(m) = SM(m-1) + DC(m-1)$. So,

$$SM(m) = 3SM(m-1) + DC(m-1) \tag{2}$$

By the same way, other 3 recursive equations hold.

$$\begin{aligned} DM(m) &= 3DM(m-1) + SC(m-1) & (3) \\ SC(m) &= SM(m-1) + 3DC(m-1) & (4) \\ DC(m) &= DM(m-1) + 3SC(m-1) & (5) \end{aligned}$$

From (2) and (4), we get

$$SM(m) + SC(m) = 4(SM(m-1) + DC(m-1)) \tag{6}$$

By (1) and (2) + (5), we get

$$SM(m) + DC(m) = 2(SM(m-1) + SC(m-1)) + 2^{2m} \tag{7}$$

From (6) and (7), we can obtain the recursive relation of $SM(m) + SC(m)$:

$$SM(m) + SC(m) = 4 \times 2^{2(m-1)} + 8(SM(m-2) + SC(m-2)) \tag{8}$$

It is easy to know $SM(0) + SC(0) = 4, SM(1) + SC(1) = 12$.

When $m = 2k$, $k = 0, 1, 2, \ldots$, $SM(2k) + SC(2k) = 2^{3k+1}(2^k + 1)$

When $m = 2k+1$, $k = 0, 1, 2, \ldots$, $SM(2k+1) + SC(2k+1) = 2^{3k+2}(2^{k+1} + 1)$

So

$$p_{2k} = \frac{2^{3k+1}(2^k+1)}{2^{2(2k+1)}} = \frac{1}{2} + \frac{1}{2^{k+1}},$$

$$p_{2k+1} = \frac{2^{3k+2}(2^{k+1}+1)}{2^{2(2k+2)}} = \frac{1}{2} + \frac{1}{2^{k+2}}.$$

i.e.

$$p_m = \frac{1}{2} + \frac{1}{2^{1+\lfloor\frac{m+1}{2}\rfloor}}.$$

End proof.

By Theorem 1, we can prove the following 3 corollaries.

Corollary 1. Suppose $q_m = Pr(\delta_m(y) = \delta_m(c) \oplus \delta_m(x))$, $m = 0, 1, 2, \ldots$, then

$$q_m = \frac{1}{2} + \frac{1}{2^{m+1}}.$$

Proof. It is obviously that $q_0 = p_0 = 1$. For $m > 0$, we denote A, B, C, D respectively as:

$$\begin{aligned} A &= F(x,m) \oplus \delta_m(x) \oplus F(c,m) \oplus \delta_m(c), \\ B &= F(y,m) \oplus \delta_m(y), \\ C &= F(x,m) \oplus F(c,m), \\ D &= F(y,m). \end{aligned}$$

If $A = B$, $C = D$ or $A \neq B$, $C \neq D$, then $\delta_m(y) = \delta_m(c) \oplus \delta_m(x)$. From Theorem 1, we know that,

$$Pr(A = B) = p_m,\ Pr(C = D) = p_{m-1},$$

$$Pr(A \neq B) = 1 - p_m,\ Pr(C \neq D) = 1 - p_{m-1}.$$

So,

$$q_m = p_m p_{m-1} + (1 - p_m)(1 - p_{m-1}) = \frac{1}{2} + \frac{1}{2^{m+1}}.$$

Corollary 2. Denote a_i, b_i, c_i as three random and independent n-bit (n>0) integers, $c_{i,n} = \delta_n(a_i + b_i)$, $1 \leq i \leq 3$. Then

$$Pr(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0) = \frac{1}{2} + 2^{-3n-1}. \tag{9}$$

Proof: From Corollary 1, it is obvious that

$$Pr(c_{i,n} = 0) = \frac{1}{2} + \frac{1}{2^{n+1}},\ 1 \leq i \leq 3.$$

Using the Piling-up Lemma:

$$Pr(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0) = \frac{1}{2} + \frac{1}{2} \cdot 2^3 \cdot (\frac{1}{2^{n+1}})^3 = \frac{1}{2} + 2^{-3n-1}.$$

Corollary 3. Suppose $y = c + x$, (y, c and $x are three k-bit (n > 0) integers$, $k = 1, 2, \ldots$, then

$$Pr(y < 2^k) = \frac{1}{2} + \frac{1}{2^{k+1}}.$$

The corollary implies that the sum of two k-bit positive integers that has no bit carry in position $k + 1$ causes a bias $\frac{1}{2^{k+1}}$.

Remark. Corollaries 1-3 can be proved by other methods, and corollary 2 is also mentioned in [11].

Applying Theorem 1 directly to ABC v2, we can get a similar linear expression with advantage. Provided that the output key stream sequence is $y_t = c_t + z_0^t$ at time t, the following probability advantage holds.

$$Pr(F(y_t, m) = F(c_t, m) \oplus F(z_0^t, m)) = \frac{1}{2} + \frac{1}{2^{1+\lfloor \frac{m+1}{2} \rfloor}}.$$

Utilizing the above probability advantages with $m = 1, 2, 3$ and some properties of $C(x)$, we can get at least 2^{95} weak keys for ABC v2 under which all the ABC v2 keys can be retrieved successively with computation complexity 2^{65} and data complexity 2^{54} key stream words. The details of the cryptanalysis on weak keys is omitted because of page limit.

4 Constrained Linear Expressions in Modular Addition Equation

Recently, Wu and Preneel found another constrained linear expression among three Modular Addition equations in which the constrained conditions are XOR equations. The linear expression can be used to find 2^{96} weak keys under which all other ABC v2 keys can be retrieved more efficiently. The linear expression in [11] has a probability advantage convergence when the bit position n ($1 \leq n \leq 32$) become higher, but the statistical advantage is estimated by the experimental data. In this section, we will give the strict mathematic proof. In addition, we also expand the linear expression to many other constrained linear expressions which may be useful in analyzing other symmetric ciphers.

4.1 Constrained Linear Expression by Wu and Preneel

In order to conveniently describe the constrained linear expressions and our proof, we formalize a conditional probability notation in [11].

Definition 2. Let $a_i, b_i, c_i (1 \leq i \leq 3)$ as three random n-bit (n>0) integers, the constraint A is $a_3 = a_1 \oplus a_2$. We define the probability that event B occurs under condition A as:

$$Pr_A(B) = Pr(B|A)$$

About $c_{i,n} = \delta_n(c_i)$, there is the following lemma in [11].

Lemma 1. Denote a, b as two random independent n-bit integers, $c_n = \delta_n(a + b)$, $a_{n-1} = \delta_{n-1}(a)$, $b_{n-1} = \delta_{n-1}(b)$. Then $c_n = (a_{n-1} \circ b_{n-1}) \oplus ((a_{n-1} \oplus b_{n-1}) \circ c_{n-1})$.

In the following, we will prove that, $Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0)$ has a recursion with variable n, and changes with different constraint conditions. Wu and Preneel estimated that the statistical probability is about 0.5714.

Theorem 2. Denote $a_i, b_i, c_i (1 \leq i \leq 3)$ as three random n-bit (n>0) integers. If $c_{i,n} = \delta_n(a_i + b_i)$, $a_1 \oplus a_2 = a_3$, then

$$Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0) = \frac{4}{7} + \frac{3}{7} \times \frac{1}{8^n} \tag{10}$$

Proof. From Lemma 1, we can obtain three expressions.

$$c_{1,n} = (a_{1,n-1} \circ b_{1,n-1}) \oplus ((a_{1,n-1} \oplus b_{1,n-1}) \circ c_{1,n-1}), \tag{11}$$
$$c_{2,n} = (a_{2,n-1} \circ b_{2,n-1}) \oplus ((a_{2,n-1} \oplus b_{2,n-1}) \circ c_{2,n-1}), \tag{12}$$
$$c_{3,n} = (a_{3,n-1} \circ b_{3,n-1}) \oplus ((a_{3,n-1} \oplus b_{3,n-1}) \circ c_{3,n-1}). \tag{13}$$

So,

$$\begin{aligned} c_{1,n} \oplus c_{2,n} \oplus c_{3,n} \quad = \quad & \Phi \oplus ((a_{1,n-1} \oplus b_{1,n-1}) \circ c_{1,n-1}) \\ & \oplus((a_{2,n-1} \oplus b_{2,n-1}) \circ c_{2,n-1}) \\ & \oplus(a_{3,n-1} \oplus b_{3,n-1}) \circ c_{3,n-1}). \end{aligned}$$

Here $\Phi = (a_{1,n} \circ b_{1,n}) \oplus (a_{2,n} \circ b_{2,n}) \oplus (a_{3,n} \circ b_{3,n})$.

Because $a_1 \oplus a_2 = a_3$, there are total 32 states for $(a_{1,n-1}, a_{2,n-1}, b_{1,n-1}, b_{2,n-1}, b_{3,n-1})$. Let $\alpha = (a_{1,n-1} \oplus b_{1,n-1},\ a_{2,n-1} \oplus b_{2,n-1},\ a_{3,n-1} \oplus b_{3,n-1})$. We divide 32 states into 3 cases according to the value of α, and each case has the same probability. Before we discuss the probabilities for 3 cases, we denote

$$\Delta_n = Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0)$$

$$\triangle_n(\alpha) = Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0 | \wedge \alpha).$$

1. If $\alpha = (0,0,0)$, then $b_{1,n-1} = a_{1,n-1}$, $b_{2,n-1} = a_{2,n-1}$, $b_{3,n-1} = a_{1,n-1} \oplus a_{2,n-1}$, $\Phi = 0$, and $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0$. Because $a_{1,n-1}$ and $a_{2,n-1}$ are independent, so
$$\triangle_n(\alpha) = \frac{2^2}{32} = \frac{1}{8}$$

2. If $\alpha = (1,0,0)$, then $b_{1,n-1} = a_{1,n-1} \oplus 1$, $b_{2,n-1} = a_{2,n-1}$, $b_{3,n-1} = a_{1,n-1} \oplus a_{2,n-1}$, $\Phi = a_{1,n-1}$, and $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = a_{1,n-1} \oplus c_{1,n-1}$.

   Note that $a_{1,n-1}$ and $a_{2,n-1}$ are independent, and $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0$ if and only if $c_{1,n-1} = a_{1,n-1}$, so
$$\triangle_n(\alpha) = \frac{1}{16}$$

Table 1: The Experimental Data of $\Delta_n - \frac{1}{2}$ in [3]

n	Probability advantage	Full space	Tested space
1	0.125	2^5	2^5
2	0.078125	2^{10}	2^{10}
3	0.072265625	2^{15}	2^{15}
4	0.071533203125	2^{20}	2^{20}
5	0.071441650390625	2^{25}	2^{25}
6	0.071430206298828125	2^{30}	2^{30}
7	0.071428775787353515625	2^{35}	2^{35}
8	0.071428596973419189453125	2^{40}	2^{40}
16	0.071424152	2^{80}	2^{32}
32	0.071434624	2^{160}	2^{32}

Similarly, we can easily prove that, if $\alpha = (0,0,1)$, $(0,1,0)$, $(0,1,1)$, $(1,0,1)$ or $(1,1,0)$, the same probability holds.

$$\triangle_n(\alpha) = \frac{2}{32} = \frac{1}{16}.$$

3. If $\alpha = (1,1,1)$, then $b_{1,n-1} = a_{1,n-1} \oplus 1$, $b_{2,n-1} = a_{2,n-1} \oplus 1$, $b_{3,n-1} = a_{1,n-1} \oplus a_{2,n-1} \oplus 1$, and $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = c_{1,n-1} \oplus c_{2,n-1} \oplus c_{3,n-1}$. So the following recursion holds.

$$\triangle_n(\alpha) = \frac{1}{8}\triangle_{n-1}$$

Summing up the above discussions, we obtain the recursion:

$$\triangle_n = \frac{1}{8} + \frac{1}{16} \times 6 + \frac{1}{8}\triangle_{n-1} = \frac{1}{2} + \frac{1}{8}\,\triangle_{n-1}\,.$$

From $\triangle_1 = \frac{5}{8}$, we get the equation:

$$\triangle_n = \frac{4}{7} + \frac{3}{7} \times \frac{1}{8^n}.$$

End Proof.

The correctness of the experimental probabilities in Table 1 can be verified by our probability formula.

Theorem 3. Provided that a_i, b_i, c_i are random n-bit (n≥2) integers, $c_{i,n} =$

$\delta_n(a_i + b_i)$, $h_{i,n} = \delta_{n-1}(a_i) \oplus \delta_{n-2}(a_i)$, $i = 1, 2, 3$, and $a_3 = a_1 \oplus a_2$. If $h_{i,n} = 0$, $i = 1, 2$ or 3, then

$$Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0|(h_{i,n} = 0)) = \frac{277}{448} + \frac{3}{7} \times \frac{1}{8^n}. \qquad (14)$$

Proof. Let $\Delta_n(h_{1,n}) = Pr_A(c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0|(h_{i,n} = 0))$. We only prove the following probability holds.

$$\Delta_n(h_{1,n}) = \frac{277}{448} + \frac{3}{7} \times \frac{1}{8^n}.$$

By iterating recursions (11), (12), and (13) twice, we get the following equation:

$$c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = \Psi \bigoplus_{i=1}^{3} ((a_{i,n-1} \oplus b_{i,n-1}) \circ (a_{i,n-2} \oplus b_{i,n-2}) \circ c_{i,n-2})$$

$$\Psi = \bigoplus_{i=1}^{3} (a_{i,n-1} \circ b_{i,n-1}) \oplus ((a_{i,n-1} \oplus b_{i,n-1}) \circ a_{i,n-2} \circ b_{i,n-2})$$

There are total 4×1024 states ($a_{1,n-1}$, $a_{1,n-2}$, $a_{2,n-1}$, $a_{2,n-2}$, $a_{3,n-1}$, $a_{3,n-2}$, $b_{1,n-1}$, $b_{1,n-2}$, $b_{2,n-1}$, $b_{2,n-2}$, $b_{3,n-1}$, $b_{3,n-2}$). By consideration of $a_3 = a_1 \oplus a_2$, and $h_{1,n} = 0$, there are 512 states left, and 9 independent variables.

Let $x_{i1} = a_{i,n-1} \oplus b_{i,n-1}$, $x_{i2} = a_{i,n-2} \oplus b_{i,n-2}$, $x_i = x_{i1} \circ x_{i2} = (a_{i,n-1} \oplus b_{i,n-1}) \circ (a_{i,n-2} \oplus b_{i,n-2})$, $i = 1, 2, 3$. We divide 512 states into 4 cases according to $\alpha = (x_1, x_2, x_3)$, and each case has the same probability.

1. If $\alpha = (1, 1, 1)$, then $a_{1,n-1} = b_{1,n-1} \oplus 1$, $a_{1,n-2} = b_{1,n-2} \oplus 1$, $a_{2,n-1} = b_{2,n-1} \oplus 1$, $a_{2,n-2} = b_{2,n-2} \oplus 1$, $a_{3,n-1} = b_{3,n-1} \oplus 1$, $a_{3,n-2} = b_{3,n-2} \oplus 1$. So, only 8 states left. For each case, we can easily show that $\Psi = 0$, and $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = c_{1,n-2} \oplus c_{2,n-2} \oplus c_{3,n-2}$. So

$$\Delta_n(h_{1,n} \wedge \alpha) = \frac{2^3}{512} \Delta_{n-2} = \frac{1}{64} \Delta_{n-2}$$

2. If $\alpha = (0, 0, 0)$, $\Delta_n(h_{1,n} \wedge \alpha) = \frac{168}{512}$.

   Denote $\beta = (x_{11}, x_{21}, x_{31})$, $\gamma = (x_{21}, x_{22}, x_{32})$. We divide all the states with $\alpha = (0, 0, 0)$ into 8 sets:

(a) If $\beta = (0,0,0)$, $\gamma = (*,*,*)$, $'*'$ represents a random bit, there are 6 independent variables left, so there are $2^6 = 64$ states.

(b) If $\beta = (0,0,1)$, $\gamma = (*,*,0)$, there are 32 states.

(c) If $\beta = (0,1,0)$, $\gamma = (*,0,*)$, there are 32 states.

(d) If $\beta = (1,0,0)$, $\gamma = (0,*,*)$, there are 32 states.

(e) If $\beta = (0,1,1)$, $\gamma = (*,0,0)$, there are 16 states.

(f) If $\beta = (1,1,0)$, $\gamma = (0,0,*)$, there are 16 states.

(g) If $\beta = (1,0,1)$, $\gamma = (0,*,0)$, there are 16 states.

(h) If $\beta = (1,1,1)$,$\gamma = (0,0,0)$, there are 8 sets.

Therefore, there are total $64 + 3 \times 32 + 3 \times 16 + 8 = 216$ states when $\alpha = (0,0,0)$, and only 168 states result in $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0$. So, $\Delta_n(h_{1,n} \wedge \alpha) = \frac{168}{512}$.

3. If $\alpha = (0,0,1)$, $(0,1,0)$, or $(1,0,0)$, $\Delta_n(h_{1,n} \wedge \alpha) = \frac{36}{512}$.

4. If $\alpha = (0,1,1)$, $(1,0,1)$ or $(1,1,0)$, $\Delta_n(h_{1,n} \wedge \alpha) = \frac{12}{512}$.

Summing up the above 4 cases, we get the recursion

$$\Delta_n(h_{1,n} \wedge \alpha) = \frac{168}{512} + \frac{36}{512} \times 3 + \frac{12}{512} \times 3 + \frac{1}{64}\Delta_{n-2} = \frac{277}{448} + \frac{3}{7} \cdot \frac{1}{8^n}.$$

For $h_{2,n}, h_{3,n}$, the same conclusion holds.
End proof.

Theorem 4. Provided that $a_i, b_i, c_i (1 \leq i \leq 3)$ as three random n-bit (n≥2) integers, $c_{i,n} = \delta_n(a_i + b_i)$, $a_1 \oplus a_2 = a_3$, $h_{i,n} = \delta_{n-1}(a_i) \oplus \delta_{n-2}(a_i)$. If $c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0$, then

$$Pr_A(h_{i,n} = 0 | c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0) = \frac{S_n}{2\Delta_n} = \frac{277 \times 2^{3n} + 3 \times 2^6}{2^{3n+9} + 3 \times 2^7} \qquad (15)$$

Here $1 \leq i \leq 3$.

Proof. The event H denotes $\{h_{i,n} = \delta_{n-1}(a_i) \oplus \delta_{n-2}(a_i) = 0\}$, event C is $\{c_{1,n} \oplus c_{2,n} \oplus c_{3,n} = 0\}$.

By the formula of conditional probability, we can get

$$Pr_A(CH) = Pr_A(C|H)Pr_A(H) = Pr_A(H|C)Pr_A(C).$$

Table 2: The Experimental Data of $Pr_A(H|C) - \frac{1}{2}$ in [3]

n	Probability advantage	Full space	Tested space
2	0.04054054054	2^8	2^8
3	0.04095563140	2^{13}	2^{13}
4	0.04100811619	2^{18}	2^{18}
5	0.04101468625	2^{23}	2^{23}
6	0.04101550765	2^{28}	2^{28}
7	0.04101561033	2^{33}	2^{33}
16	0.04103037	2^{78}	2^{32}

From Theorem 2, $Pr_A(C) = \Delta_n$, and from Theorem 3, $Pr_A(C|H) = \Delta h_{i,n}$. It is obvious that $Pr_A(H) = \frac{1}{2}$.

$$Pr_A(H|C) = \frac{\Delta_{h_{i,n}}}{2\Delta_n} = \frac{3 \times 2^6 + 277 \times 2^{3n}}{3 \times 2^7 + 2^{3n+9}}(1 \leq i \leq 3, n \geq 2)$$

The experimental data for $Pr_A(H|C)$ by Wu and Preneel is listed in **Table 2**. It is easy to verify the correctness of the experimental data by our formula.

Utilizing the probability advantages in Theorem 2, 3, and 4, Wu and Preneel found another 2^{96} weak keys. A weak key is distinguished with 2^{13} key stream words and $2^{13.5}$ operations by the probability advantage:

$$Pr(y_{t,16} \oplus y_{t+63,16} \oplus y_{t+127,16} = 0) - \frac{1}{2} \approx 0.0714.$$

After identifying the weak key, Wu and Preneel employed the estimated probability in Theorem 3 and 4 to retrieve the internal state of LFSR by the Fast Correlation Attack. Then, the coefficients of Component B and C is easy to be achieved.

4.2 The Generic Constrained Linear Expressions

The expression can be expanded to the following generic linear expressions. Provided that there are s Modular Addition equations and t constraints:

$$c_{i,n} = \delta_n(a_i + b_i), i = 0, 1, 2, ..., s-1$$

$$F_j = F_j(a_1, a_2, ..., a_k, b_1, b_2, ..., b_k) = 0, j = 0, 1, 2, .., t-1$$

$F_j,\ j = 0, 1, .., t-1$ are boolean functions. Discussing the probabilities of these linear expressions will be important for analyzing many existing symmetric ciphers.

$$Pr((c_{1,n} \oplus \ldots \oplus c_{s,n} = 0) | (F_1 = 0) \wedge (F_2 = 0) \wedge (\ldots) \wedge (F_t = 0)).$$

For example, if $s = 3$, $t = 2$, and $F_0 = F(a_1, a_2, a_3, b_1, b_2, b_3) = a_1 \oplus a_2 \oplus a_3$, $F_1 = F(a_1, a_2, a_3, b_1, b_2, b_3) = b_1 \oplus b_2 \oplus b_3$, the probability for the linear expression (17) is:

$$\triangle_n = \frac{5}{8} + \frac{1}{8} \triangle_{n-1}$$

It is obvious that $\triangle_1 = \frac{5}{8}$, so,

$$Pr((c_{1,n} \oplus \ldots \oplus c_{s,n} = 0) | (F_1 = 0) \wedge (F_2 = 0)) = \frac{5}{7} - \frac{5}{7} \times \frac{1}{8^n}. \qquad (16)$$

Comparing the probabilities in (9), (10), (14), (15) and (16), the advantage of (9) in the random situation is not enough to analyze ABC v2, and the (10), (14) and (15) have the substantial advantage to search for the keys. Probability (16) can become larger with another more condition. So, we can discuss many other linear expressions with different constraint conditions have some surprising change in probability. This phenomena is alluring in cryptanalysis. We believe that these kinds of linear expressions are helpful to analyze some other symmetric ciphers.

5 Conclusion

In this paper, we discuss two linear expressions in Modular Addition Equations which have probability advantages, and the two linear expressions can be expanded to many other general linear expressions which may be applicable to analyze some other symmetric ciphers. Corresponding to different constraint conditions, these linear expressions may have surprising probability advantages which leak important information about the ciphers.

References

[1] V. Anashin, A. Bogdanov, I. Kizhvatov, S. Kumar, ABC: A new fast flexible stream cipher, ECRYPT Stream Cipher Project Report 2005/001, http://www.ecrypt.eu.org/stream, 2005.

[2] V. Anashin, A. Bogdanov, I. Kizhvatov, S. Kumar, ABC: A new fast flexible stream cipher, Version 2, http://crypto.rsuh.ru/papers/abc-spec-v2.pdf, 2005.

[3] C. Berbain, H. Gilbert, Cryptanalysis of ABC, eSTREAM, ECRYPT Stream Cipher Project, Report 2005/048, http://www.ecrypt.eu.org/stream, 2005.

[4] E. Biham, A. Shamir, Differetial cryptanalysis of DES-like cryptosystems, J. of Cryptology, 4(1):3-72, 1991.

[5] S. Goldwasser, S. Micali, C.Rackoff, The Knowledge Complexity of Interactive Proof-Systems, SIAM Journal on Comput., Vol. 18, No. 1, 1989.

[6] S. Goldwasser, S. Micali, Probabilistic Enryption and How to Play Mental Poker Keeping Secret All Partial Information, 14th ACM STOC, pp. 365-377, 1982.

[7] S. Khazaei, Divide and conquer attack on ABC stream cipher. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/052, http://www.ecrypt.eu.org/stream, 2005.

[8] S. Khazaei, M. Kiaei, Distinguishing attack on the-ABC v.1 and v.2. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/061, 2005.

[9] M. Mastsui, Linear cryptanalysis method for DES cipher, Advances in Cryptology-EUROCRYPT'93, Springer-Verlag, pp.12-16, 1994.

[10] W. Meier and O. Staffelbach, Fast Correlation Attacks on Stream Ciphers, Journal of Cryptology 1(3), pp. 159-176, 1989.

[11] H. J. Wu, B. Preneel, Cryptanalysis of ABC v2, http://www.ecrypt.eu.org/stream/abc.html, Feb of 2006.

[12] A. C. Yao, Theroy and Applications of Trapdoor Functions, In Proc. of the 23th Annu. IEEE Symp. on Foundations of Computer Science, pp. 80–91, 1982.

[13] K. Zeng, H. Huang, On the linear syndrome method in cryptanalysis, Advances in Cryptology-EUROCRYPT'88,Berlin: Springer-Verlag, pp.469-478, 1990.

[14] K. Zeng, C. H. Yang, T. R. N. Rao. An improved linear syndrome algorithm in cryptanalysis with application, Advances in Cryptology-EUROCRYPT'90, Springer-Verlag, 34-47, 1991.