The Key-Dependent Attack on Block Ciphers

Xiaorui Sun * Xuejia Lai †

Abstract

In this paper, we formalize an attack scheme using the key-dependent property, called key-dependent attack. In this attack, the intermediate value, whose distribution is key-dependent, is considered. The attack determines whether a key is right by conducting statistical hypothesis test of the intermediate value. The time and data complexity of the key-dependent attack is also discussed.

We also apply key-dependent attack on reduced-round IDEA. This attack is based on the key-dependent distribution of certain items in Biryukov-Demirci Equation. The attack on 5.5-round variant of IDEA requires 2^{21} chosen plaintexts and $2^{112.1}$ encryptions. The attack on 6-round variant requires 2^{49} chosen plaintexts and $2^{112.1}$ encryptions. Compared with the previous attacks, the key-dependent attacks on 5.5-round and 6-round IDEA have the lowest time and data complexity, respectively.

Key words: Block Cipher, Key-Dependent Attack, IDEA

*Shanghai Jiao Tong University, China. E-mail: sunsirius@sjtu.edu.cn.
†Shanghai Jiao Tong University, China. E-mail: lai-xj@cs.sjtu.edu.cn.

1 Introduction

In current cryptanalysis on block ciphers, widespread attacks use special probability distributions of certain intermediate values. These probability distributions are considered as invariant under different keys used. For example, differential cryptanalysis [7] makes use of the probability of the intermediate differential with high probability. Its value is assumed not to vary remarkably with different keys. Linear cryptanalysis [23] is based on the bias of the linear approximation, which is also generally constant for different keys.

Instead of concentrating on the probability distribution which is invariant for different keys, Ben-Aroya and Biham first proposed the key-dependent property in [2]. Key-dependent property means that the probability distribution of intermediate value varies for different keys. In [2], an attack on Lucifer using key-dependent differential was presented. Knudsen and Rijmen also used similar idea to attack DFC in [20].

In this paper, we consider the key-dependent property. The distribution of intermediate value which is key-dependent is called *key-dependent distribution*. Assume that there are some randomly chosen encryptions. For the intermediate values calculated from these encryptions with the actual key, they should conform the key-dependent distribution. On the other hand, if we use a wrong key to calculate the intermediate values, they are assumed to conform random distribution. Basing on key-dependent distribution, we formalize a scheme of discovering the actual key by performing statistical hypothesis test[17] on possible keys, and we call this scheme *key-dependent attack*. For a given key, the null hypothesis of the test is that the intermediate value conforms the key-dependent distribution determined by the key. The samples of the test are the intermediate values calculated from a few encryptions. If the test is passed, the given key is concluded to be the actual key, otherwise it is discarded. For the keys that share the same key-dependent distribution and the same intermediate value calculation, the corresponding hypothesis tests can be merged to reduce the time needed. By this criterion, the key space is divided into several *key-dependent subsets*.

Due to the scheme of the key-dependent attack, the time complexity of the attack is determined by the time for distinguishing between random distribution and key-dependent distribution. The time needed relies on the entropy of the key-dependent distribution: the closer the key-dependent distribution is to the uniform distribution, the more encryptions are needed. For each key-dependent subset, the number of encryptions and the criteria of rejecting hypothesis can be chosen so that the attack on this subset is optimized. The expected time of the attack on each subset is also obtained.

Total expected time complexity can be calculated from the expected time on each key-dependent subset. Different orders of the key-dependent subsets attacked have different expected time complexities. The order with minimal expected time complexity is presented. The total expected time complexity is also minimized in this way if the actual key is supposed to be chosen uniformly from the whole key space.

This paper also presents a key-dependent attack on block cipher IDEA. The block cipher IDEA (International Data Encryption Algorithm) was proposed in [21, 22]. The cryptanalysis of IDEA was discussed in [1, 3, 4, 5, 6, 8, 9, 11, 12, 13, 14, 15, 16, 18, 19, 24, 25], and no attack on full version IDEA is faster than exhaustive search so far. We investigate the Biryukov-Demirci Equation, which is widely used in recent attacks on IDEA[1, 5, 6, 13, 16, 18]. We find that particular items of Biryukov-Demirci Equation satisfy key-dependent distribution under some specific constraints. This makes it possible to perform the key-dependent attack on IDEA. Biryukov-Demirci Equation is used to recover the intermediate values from encryptions.

Our key-dependent attack on 5.5-round variant of IDEA requires 2^{21} chosen plaintexts and has a time complexity of $2^{112.1}$ encryptions. Our key-dependent attack on the 6-round variant of IDEA requires 2^{49} chosen plaintexts and has a time complexity of $2^{112.1}$ encryptions. These attacks use both fewer chosen plaintexts and less time than all the previous corresponding attacks. We summarize our attacks and pre-

vious attacks in Table 1, where the data complexity is measured in the number of plaintexts and the time complexity is measured in the number of encryptions needed in the attack.

Rounds	Attack type	Data	Time	Ref.
4.5	Impossible Differential	2^{64} CP	2^{112}	[3]
4.5	Linear	16 CP	2^{103}	[5]
5*	Meet-in-the-Middle	2^{24} CP	2^{126}	[13]
5*	Meet-in-the-Middle	$2^{24.6}$ CP	2^{124}	[1]
5	Linear	$2^{18.5}$ KP	2^{103}	[6]
5	Linear	2^{19} KP	2^{103}	[5]
5	Linear	16 KP	2^{114}	[6]
5.5	Higher-Order Differential-Linear	2^{32} CP	$2^{126.85}$	[6]
6	Higher-Order Differential-Linear	$2^{64} - 2^{52}$ KP	$2^{126.8}$	[6]
5.5	Key-Dependent	2^{21} CP	$2^{112.1}$	Section 5.1
6	Key-Dependent	2^{49} CP	$2^{112.1}$	Section 5.2

CP - Chosen Plaintext, KP - Known Plaintext

* Attack on IDEA starting from the first round

Table 1: Selected Results of attacks on IDEA

The paper is organized as follows: In Section 2 we give a general view of the key-dependent attack. In Section 4 we show that the probability distribution of some items of the Biryukov-Demirci Equation is a key-dependent distribution. In Section 5 we present two key-dependent attacks on reduced-round IDEA. Section 6 concludes this paper.

2 Key-Dependent Attack

In [2], Ben-Aroya and Biham first proposed the key-dependent property and implemented a key-dependent differential attack on Lucifer. Knudsen and Rijmen also used similar idea to attack DFC in [20].

In this section, we formalize a scheme of identifying the actual key using the following key-dependent property(with high success probability).

Definition 2.1. *For a block cipher, if the probability distribution of an intermediate value varies for different keys under some specific constraints, then this probability distribution is defined as* key-dependent distribution.

Consider some randomly chosen encryptions satisfying the specific constraints. If one uses the actual key to calculate the intermediate value, it should conform key-dependent distribution. If one uses a wrong key to calculate the intermediate value, it is assumed to be randomly distributed. With such a property, determining whether a given key is right can be done by distinguishing which distribution the intermediate value conforms, key-dependent distribution or random distribution.

We propose an attack scheme, called *key-dependent attack*, using key-dependent distribution. The attack uses statistical hypothesis test, whose idea is also used in differential and linear attack [17], to distinguish between key-dependent distribution and random distribution. For a key, the null hypothesis of the test is that the intermediate value conforms the key-dependent distribution determined by the key. Then the attack uses some samples to determine whether the hypothesis is right. The samples of the statistical hypothesis test are

the intermediate values obtained from the encryptions satisfying the specific constraints. If the key passes the hypothesis test, the attack concludes that the key is right, otherwise the key is judged to be wrong.

For the keys that share the same key-dependent distribution and the same intermediate value calculation, the corresponding hypothesis tests can be merged. Hence the whole key space is divided into several key-dependent subsets. (Similar idea is proposed in [2].)

Definition 2.2. *A* key-dependent subset *is a tuple (P, U), where P is a fixed key-dependent distribution of intermediate value, and U is a set of keys that share the same key-dependent distribution P and the same intermediate value calculation.*

Definition 2.3. *The* key fraction *(f) of a key-dependent subset is the ratio between the size of U and the size of the whole key space.*

The key-dependent attack determines which key-dependent subset the actual key is in by conducting hypothesis tests on each key-dependent subset. Such process on a key-dependent subset (P, U), called *individual attack*, can be described as the following four phases:

1. **Parameter Determining Phase** Determine the size of the samples and the criteria of rejecting the hypothesis that the intermediate values conform P.
2. **Data Collecting Phase** Randomly choose some encryptions according to the specific constraints. [1]
3. **Judgement Phase** Calculate the intermediate values from the collected encryptions. If the results satisfy the criteria of rejection, then discard this key-dependent subset, otherwise enter the next phase.
4. **Remaining Key Bits Search Phase** Exhaustively search remaining key bits to find the whole key. If the exhaustive search does not find the whole actual key, then start another individual attack on the next key-dependent subset.

The time complexity of the key-dependent attack is determined by the time complexity of each individual attack and the order of performing these individual attacks.

For a key-dependent subset (P, U), the time needed for individual attacks relies on the entropy of P: the closer P is to the random distribution, the more difficult the attack is—to ensure the same probability of making the right judgement, the attack needs more encryptions. This indicates that individual attacks for different keys have different time complexities. The time complexity of each individual attack is determined by corresponding key-dependent distribution P. For each key-dependent subset, the number of encryptions and the criteria of rejecting hypothesis are then chosen to minimize the time complexity of this individual attack.

To minimize the time complexity of an individual attack, the attack should consider the probability of committing two types of errors: Type I error and Type II error. Type I error occurs when the hypothesis is rejected for a key-dependent subset while in fact the actual key is in U, and the attack will fail to find the actual key in this case. The probability of Type I error is also defined as significant level, denoted as α. Type II error occurs when the test is passed while in fact it is not right, and in this case the attack will come into the remaining key bits search phase, but will not find the actual key. The probability of Type II error is denoted as β. With a fixed size of samples (denoted as N) and the significance level α, the criteria of rejecting the hypothesis is determined, and the probability of Type II error β is also fixed. For a fixed size of

[1] Though each individual attack chooses encryptions randomly, one encryption can be used for many individual attacks thus to reduces the total data complexity.

samples, it is impossible to reduce both α and β simultaneously. In order to reduce both α and β, the attack has to use a larger size of samples, but time and data complexity will increase. Hence, an individual attack needs to balance between the size of samples, and the probability of making wrong judgement.

For a key-dependent subset (P, U), if the actual key is not in this subset, the expected time complexity (measured by the number of encryptions) of the individual attack on this subset is

$$W = N + \beta|U| \tag{1}$$

If the actual key is in this subset, the expected time of the individual attack on this subset is

$$R = N + (1 - \alpha)\frac{|U|}{2} \tag{2}$$

Since the time complexity is dominated by attacking on wrong key-dependent subsets (there is only one key-dependent subset containing the actual key), the attack only needs to minimize the time complexity of the individual attack for each wrong key-dependent subset to minimize the total time complexity. Although α does not appear in Equation (1), α affects the success probability of the attack, so α should also be considered. We set one upper bound of α to ensure that the success probability is above a fixed value, and then choose such size of samples that Equation (1) is minimized, in order to minimize the time complexity of individual attacks.

In addition, it is entirely possible that some key-dependent distributions is so close to random distribution that the expected time for performing hypothesis tests is longer than directly searching the subsets. For these key-dependent subsets, the attack exhaustively searches the subset directly instead of using statistical hypothesis test method.

On the other hand, the time complexity of the key-dependent attack is also affected by the order of performing individual attacks on different key-dependent subsets. Because the expected time complexities of individual attacks are different, different sequences of performing individual attacks result in different total expected time complexity. Assume that a key-dependent attack performs individual attacks on n key-dependent subsets in the order of $(P_1, U_1), \ldots, (P_n, U_n)$. Let R_i denote the expected time for (P_i, U_i) if the actual key is in U_i, and W_i denote the expected time if the actual key is not in U_i. We have following result:

Theorem 2.4. *The expected time for the whole key-dependent attack is minimal if the following condition is satisfied*

$$\frac{f_1}{W_1} \geq \frac{f_2}{W_2} \geq \cdots \geq \frac{f_n}{W_n} \tag{3}$$

Proof. The expected time of the attack in the order of $(P_1, U_1), \ldots, (P_n, U_n)$ is

$$\begin{aligned}\Phi =& f_1[R_1 + \alpha(W_2 + W_3 + \cdots + W_n)] + f_2[W_1 + R_2 + \alpha(W_3 + \cdots + W_n))] \\ &+ f_3[W_1 + W_2 + R_3 + \alpha(W_4 + \ldots W_n)] + \cdots + f_n(W_1 + W_2 + \ldots W_{n-1} + R_n) \\ =& \sum_{i=1}^{n} f_i R_i + \sum_{i=1}^{n}(f_i \sum_{j=1}^{i-1} W_j) + \alpha \sum_{i=1}^{n}(f_i \sum_{j=i+1}^{n} W_j)\end{aligned} \tag{4}$$

If the attack is performed in the order of $(P_{s_1}, U_{s_1}), (P_{s_2}, U_{s_2}), \ldots, (P_{s_n}, U_{s_n})$, where $s_1, s_2, \ldots, s_n$ is a permutation of $1, 2, \ldots, n$. The expected time is

$$\Phi' = \sum_{i=1}^{n} f_{s_i} R_{s_i} + \sum_{i=1}^{n}(f_{s_i} \sum_{j=1}^{i-1} W_{s_j}) + \alpha \sum_{i=1}^{n}(f_{s_i} \sum_{j=i+1}^{n} W_{s_j}) \tag{5}$$

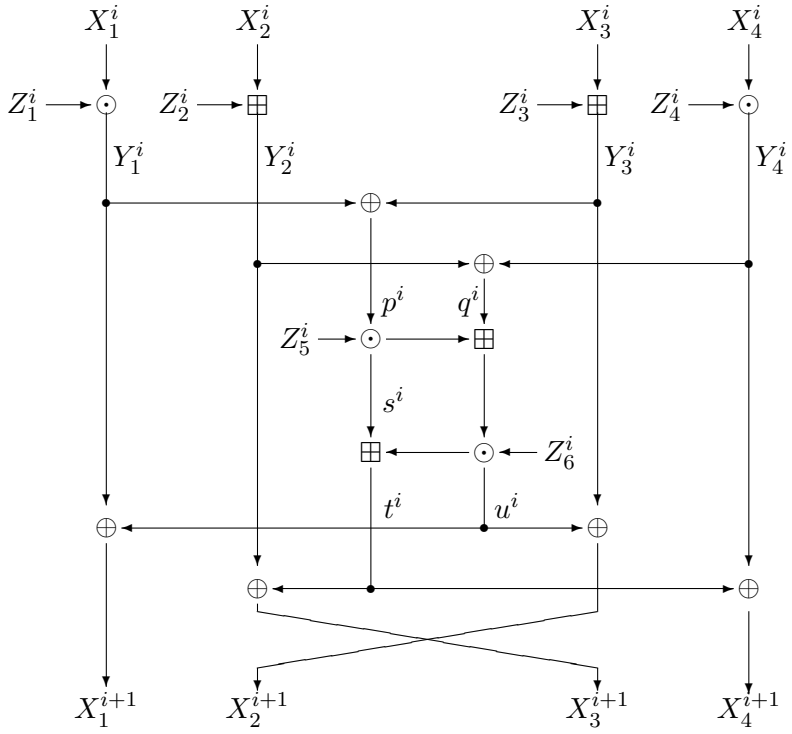

Figure 1: Round i of IDEA

$f_iW_j + \alpha f_jW_i$ occurs in Φ if and only if $j < i$ and occurs in Φ' if and only if $j' < i'$ where $s_{i'} = i$ and $s_{j'} = j$. Hence

$$\Phi - \Phi' = \sum_{j<i \text{ and } j'>i'} (f_iW_j + \alpha f_jW_i - f_jW_i - \alpha f_iW_j) \tag{6}$$

Since $f_iW_j - f_jW_i \leq 0$ for $j < i$, $\Phi - \Phi' \leq 0$ for any permutation $s_1, s_2, \ldots s_n$.

□

In the following sections of this paper, we present a concrete key-dependent attack on the block cipher IDEA.

3 IDEA Block Cipher

In this section, we give a brief introduction of IDEA and notations used later in this paper.

IDEA block cipher encrypts a 64-bit plaintext with a 128-bit key by an 8.5-round encryption. The fifty-two 16-bit subkeys are generated from the 128-bit key Z by key-schedule algorithm. The subkeys are generated in the order Z_1^1,Z_2^1,...,Z_6^1,Z_1^2,...,Z_6^8, Z_1^9,...,Z_4^9. The key Z is partitioned into eight 16-bit words which are used as the first eight subkeys. The key Z is then cyclically shifted to the left by 25 bits, and then

Round	Z_1^i	Z_2^i	Z_3^i	Z_4^i	Z_5^i	Z_6^i
1	0-15	16-31	32-47	48-63	64-79	80-95
2	96-111	112-127	25-40	41-56	57-72	73-88
3	89-104	105-120	121-8	9-24	50-65	66-81
4	82-97	98-113	114-1	2-17	18-33	34-49
5	75-90	91-106	107-122	123-10	11-26	27-42
6	43-58	59-74	100-115	116-3	4-19	20-35
7	36-51	52-67	68-83	84-99	125-12	13-28
8	29-44	45-60	61-76	77-92	93-108	109-124
9	22-37	38-53	54-69	70-85		

Table 2: The Key-Schedule of IDEA

generate the following eight subkeys. This process is repeated until all the subkeys are obtained. In Table 3, the correspondence between the subkeys and the key Z is directly given.

The block cipher partitions the 64-bit plaintext into four 16-bit words and uses three different group operations on pairs of 16-bit words: exclusive OR, denoted by $\oplus$; modular addition 2^{16}, denoted by $\boxplus$ and modular multiplication $2^{16}+1$(0 is treated as 2^{16}), denoted by $\odot$.

As Figure 1, each round of IDEA contains three layers: KA layer, MA layer and Permutation layer. We denote the 64-bit input of round i by $X^i = (X_1^i, X_2^i, X_3^i, X_4^i)$. In the KA layer, the first and the fourth words are modular multiplied with Z_1^i and Z_4^i respectively. The second and the third words are modular added with Z_2^i and Z_3^i respectively. The output of the KA layer is denoted by $Y^i = (Y_1^i, Y_2^i, Y_3^i, Y_4^i)$.

In the MA layer, two intermediate values $p^i = Y_1^i \oplus Y_3^i$ and $q^i = Y_2^i \oplus Y_4^i$ are computed first. These two values are processed to give u^i and t^i,

$$u^i = (p^i \odot Z_5^i) \boxplus t^i$$

$$t^i = ((p^i \odot Z_5^i) \boxplus q^i) \odot Z_6^i$$

We denote s^i the intermediate value $p^i \oplus Z_5^i$ for convenience. The output of the MA layer is then permutated to give the output of this round $(Y_1^i \oplus u^i, Y_3^i \oplus u^i, Y_2^i \oplus t^i, Y_4^i \oplus t^i)$, which is also the input of round $i+1$, denoted by $(X_1^{i+1}, X_2^{i+1}, X_3^{i+1}, X_4^{i+1})$. The complete diffusion, which means every bit of $(X_1^{i+1}, X_2^{i+1}, X_3^{i+1}, X_4^{i+1})$ is affected by every bit of $(Y_1^i, Y_2^i, Y_3^i, Y_4^i)$, is obtained in the MA layer.

In this paper, we will use $P = (P_1, P_2, P_3, P_4)$ and $P' = (P_1', P_2', P_3', P_4')$ to denote a pair of plaintexts, where P_i and P_i' are 16-bit words. $C = (C_1, C_2, C_3, C_4)$ and $C' = (C_1', C_2', C_3', C_4')$ are their ciphertexts respectively. We also use the symbol ' to distinguish the intermediate values corresponding to P' from to P. For example, s^i is obtained from plaintext P and P' will generate s'^i. The notation Δ will denote the XOR difference, for instance, Δs^i is equal to $s^i \oplus s'^i$.

4 Key-Dependent Distribution on IDEA

In this section, we propose a key-dependent distribution on the block cipher IDEA. The description of IDEA is omitted here. Notions used is same to [6].

The Biryukov-Demirci relation was first proposed by Biryukov [16] and Demirci [13]. Many papers have discussed attacking on IDEA using this relation, such as [1, 5, 6, 13, 16, 18]. The relation can be written in following form:

$$\begin{aligned} LSB(C_2 \oplus C_3) =& LSB(P_2 \oplus P_3 \oplus Z_2^1 \oplus Z_3^1 \oplus s^1 \oplus Z_2^2 \oplus Z_3^2 \oplus s^2 \\ & \oplus Z_2^3 \oplus Z_3^3 \oplus s^3 \oplus Z_2^4 \oplus Z_3^4 \oplus s^4 \oplus Z_2^5 \oplus Z_3^5 \oplus s^5 \\ & \oplus Z_2^6 \oplus Z_3^6 \oplus s^6 \oplus Z_2^7 \oplus Z_3^7 \oplus s^7 \oplus Z_2^8 \oplus Z_3^8 \oplus s^8 \\ & \oplus Z_2^9 \oplus Z_3^9) \end{aligned} \tag{7}$$

It is shown in [5] that, for two pairs of plaintext and ciphertext (P, C) and (P', C'), XOR their corresponding Biryukov-Demirci relation, we will obtain from Equation (7)

$$\begin{aligned} LSB(C_2 \oplus C_3 \oplus C_2' \oplus C_3') =& LSB(P_2 \oplus P_3 \oplus P_2' \oplus P_3' \oplus \Delta s^1 \oplus \Delta s^2 \\ & \oplus \Delta s^3 \oplus \Delta s^4 \oplus \Delta s^5 \oplus \Delta s^6 \oplus \Delta s^7 \oplus \Delta s^8) \end{aligned} \tag{8}$$

We call Equation (8) *Biryukov-Demirci Equation*.

The following theorem shows that the probability distribution of $LSB(\Delta s^i)$ in Biryukov-Demirci Equation is a key-dependent distribution.

Theorem 4.1. *Consider round i of IDEA. If one pair of intermediate values (p^i, p'^i) satisfies $\Delta p^i = 8000_x$, then the probability of $LSB(\Delta s^i) = LSB(8000_x \odot Z_5^i)$ is*

$$Prob(LSB(\Delta s^i) = LSB(8000_x \odot Z_5^i)) = \frac{\#W}{2^{15}} \tag{9}$$

where W is the set of all such 16-bit words w that $1 \le w \le 8000_x$ and that

$$(w * Z_5^i) + (8000_x * Z_5^i) < 2^{16} + 1 \tag{10}$$

*where * is defined as*

$$a * b = \begin{cases} a \odot b & \text{if } a \odot b \neq 0 \\ 2^{16} & \text{if } a \odot b = 0 \end{cases}$$

Proof. Consider every intermediate pair (p^i, p'^i) which satisfies $\Delta p^i = 8000_x$, excluding $(0, 8000_x)$. We have $p'^i = p^i + 8000_x$ or $p^i = p'^i + 8000_x$. Without losing generality, assume $p'^i = p^i + 8000_x$, where $1 \le p^i < 8000_x$ and $8000_x < p'^i < 2^{16}$.

If we consider only the least significant bit, $LSB(s^i) = LSB(p^i * Z_5^i)$. The following equations also hold

$$\begin{aligned} LSB(s'^i) =& LSB(p'^i \odot Z_5^i) \\ =& LSB(p'^i * Z_5^i) \\ =& LSB((p^i + 8000_x) * Z_5^i) \\ =& LSB(((p^i * Z_5^i) + (8000_x * Z_5^i)) \pmod{2^{16}+1}) \end{aligned} \tag{11}$$

In the special case when (p^i, p'^i) is $(0, 8000_x)$, let $p^i = 8000_x$, and $p'^i = 0$. The Equations (11) also holds, because $p'^i = 0$ is actually treated as 2^{16} for inputs of $\odot$ and $*$.

If $(p^i * Z_5^i) + (8000_x * Z_5^i)$ is smaller than $2^{16}+1$, then $LSB(s'^i) = LSB(s^i) \oplus LSB(8000_x * Z_5^i)$ holds because of the equivalence of XOR and modular addition for the least significant bit. Moreover, $LSB(\Delta s^i) = LSB(8000_x * Z_5^i)$ is satisfied, which means $LSB(\Delta s^i) = LSB(8000_x \odot Z_5^i)$

Otherwise, $LSB(s'^i)$ is equal to $LSB(s^i) \oplus LSB(8000_x * Z_5^i) \oplus 1$ because of the carry. So $LSB(\Delta s^i)$ equals to $LSB(8000_x \odot Z_5^i) \oplus 1$.

Therefore, we may conclude that $LSB(\Delta s^i) = LSB(8000_x \odot Z_5^i)$ if and only if the pair (p^i, p'^i) satisfies $(w * Z_5^i) + (8000_x * Z_5^i) < 2^{16}+1$, where w is either p^i or p'^i, whichever between 1 and 8000_x. And there are at most 2^{15} such w, hence Equation (11) holds. This completes the proof.

□

Remark 4.2. *Figure 4.2 plots the relation between subkey Z_5^i and the probability of $LSB(\Delta s^i) = 1$. As shown in Figure 4.2, for most Z_5^i, the probability of $LSB(\Delta s^i) = 1$ is different from uniform distribution. Hence, it is possible to perform key-dependent attack on IDEA using this key-dependent distribution.*

There are four cases for the probability of $LSB(\Delta s^i) = 1$ as Z_5^i grows from 0 to $2^{16}-1$, which can be generally approximated as following:

$$Prob(LSB(\Delta s^i) = 1) \approx \begin{cases} \frac{Z_5^i}{2^{17}} & \textit{last two bits of } Z_5^i = 0 \\ 0.5 - \frac{Z_5^i}{2^{17}} & \textit{last two bits of } Z_5^i = 1 \\ 1.0 - \frac{Z_5^i}{2^{17}} & \textit{last two bits of } Z_5^i = 2 \\ 0.5 + \frac{Z_5^i}{2^{17}} & \textit{last two bits of } Z_5^i = 3 \end{cases} \tag{12}$$

From Equation (12), following approximation also holds for most Z_5^i.

$$min\{Prob(LSB(\Delta s^i) = 0), Prob(LSB(\Delta s^i) = 1)\} \approx \begin{cases} \frac{Z_5^i}{2^{17}}, & LSB(Z_5^i) = 0 \\ 0.5 - \frac{Z_5^i}{2^{17}}, & LSB(Z_5^i) = 1 \end{cases} \tag{13}$$

This fact indicates that we can approximate left hand side of Equation (13) by fixing several most significant bits and the least significant bit.

In following sections, we will show that we only need to distinguish this approximate probability distribution from uniform distribution. Calculation shows that, for only 219 out of all 2^{16} possible Z_5^i, the difference between this approximation and accurate provability is larger than 0.01. Hence, for most Z_5^i, this approximation is close to the accurate value. For Z_5^i that can be not approximated in this way, we use other methods to deal with this situation.

5 Key-Dependent Attack on IDEA

In this section, we will present two key-dependent attacks on reduced-round IDEA. In Section 5.1, we will give a basic attack on the 5.5-round variant of IDEA and then extend it to 6-round variant in Section 5.2.

5.1 Attack on 5.5-Round Variant of IDEA

We first present one key-dependent attack on the 5.5-round variant of IDEA. The attack starts from the third round and ends before the MA layer of the eighth round. The main idea of this attack is to perform key-dependent attack based on the key-dependent distribution of Δs^4 described in Theorem 4.1.

Consider the 5.5-round variant of IDEA starting from the third round, the Biryukov-Demirci Equation can be rewritten as

$$LSB(\Delta s^4) = LSB(P_2 \oplus P_3 \oplus P_2' \oplus P_3' \oplus C_2 \oplus C_3 \oplus C_2' \oplus C_3' \oplus \Delta s^3 \oplus \Delta s^5 \oplus \Delta s^6 \oplus \Delta s^7) \quad (14)$$

Where P and P' are equivalent to X^3 and X'^3, C and C' are equivalent to Y^8 and Y'^8 by the variant of IDEA.

We first construct a pair of plaintexts satisfying the specific constraint $\Delta p^4 = 8000x$. The construction is based on the following lemma.

Lemma 5.1. *For any α, if two 16-bit words x and x' have the same least 15 significant bits, then*

- *$x \oplus \alpha$ and $x' \oplus \alpha$ have the same least 15 significant bits,*
- *$x \boxplus \alpha$ and $x' \boxplus \alpha$ have the same least 15 significant bits.*

Based on Lemma 5.1, the following proposition can be obtained.

Proposition 5.2. *If a pair of intermediate values Y^3 and Y'^3 satisfy the following conditions:*

a. $\Delta Y_1^3 = \Delta Y_3^3 = 0$

b. $\Delta Y_2^3 = 8000_x$

c. $Y_2^3 \oplus Y_4^3 = Y_2'^3 \oplus Y_4'^3$

then $\Delta s^3 = 0$ and the probability of $LSB(\Delta s^4) = 0$ can be determined by Equation(9).

Proof. From Condition a, $\Delta Y_1^1 = \Delta Y_3^1 = 0$, p^1 is equal to p'^1. Then $\Delta s^1 = 0$ is quite straightforward. From Condition b, q^1 is equal to q'^1. If p^i and q^i are fixed, u^i and t^i are also fixed with respect to any Z_5^i and Z_6^i. It indicates that $X_1^2 = Y_1^1 \oplus u^1 = X_1'^2$. Note that Y_1^2 and $Y_1'^2$ are the results of modular-multiplying X_1^2 and $X_1'^2$ with the same Z_1^2, hence Y_1^2 is equal to $Y_1'^2$.
On the other hand, $\Delta Y_2^1 = 8000_x$ means that the least significant 15 bits of Y_2^1 are equal to those of $Y_2'^1$ and the most significant bit of Y_2^1 and that of $Y_2'^1$ are different. Because u^1 is fixed, by Lemma 5.1, the least significant 15 bits of X_3^2 are equal to those of $X_3'^2$. Then ΔX_3^2 is equal to 8000_x and $\Delta Y_3^2 = 8000_x$ is obtained by modular addition with the same Z_3^2. From $\Delta Y_1^2 = 0$ and $\Delta Y_3^2 = 8000_x$, Δp^2 is 8000_x. By Theorem 4.1, the conclusion is obtained. □

In our attack, we use the plaintext pairs satisfying Proposition 5.2. We obtain Condition (a) by letting $\Delta P_1 = \Delta P_3 = 0$. By Lemma 4.1, P_2 and P_2' are fixed to have the same least significant 15 bits, and hence $\Delta Y_2^1 = 8000_x$. In order to fulfill Condition (c), we have to guess and then according to this guess, to choose P_4 and P_4' which satisfy $\Delta Y_4^3 = 8000_x$.

By Proposition 5.2, Δs^3 is equal to zero. In order to get the right hand side of Equation (14), we still need to get Δs^5, Δs^6, Δs^7. We need to guess Z_5^5, Z_1^6, Z_2^6, Z_5^6, Z_6^6, Z_1^7, Z_2^7, Z_3^7, Z_4^7 Z_5^7, Z_6^7, Z_1^8, Z_2^8, Z_3^8, Z_4^8. As shown in [6], one can partially decrypt one pair of encryptions using these 15 subkeys to calculate the values of Δs^5, Δs^6, Δs^7. These 15 subkeys only take key bits 125-99 and also cover the subkey Z_4^3. Hence, for one guessed 103 key bits, we can calculate the value of Δs^4 from a special pair of encryptions.

We also note that these 103 bits also cover the key Z_5^4, which determine the key-dependent distribution on Δs^4 according to Theorem 4.1. Therefore, we can perform the key-dependent attack on 5.5-round variant

of IDEA. As described in Section 2, the key space can be divided into 2^{103} key-dependent subsets by the 103 key bits, each contains 2^{25} keys.

For a key-dependent subset (P, U), let p denote the probability of $LSB(\Delta s^4) = 0$. For simplicity, in the following analysis, we assume that $p \leq 0.5$, the case when $p > 0.5$ is similar. Assume the size of the samples is n pairs of encryptions that satisfy the specific constraint on this key-dependent subset, and t of them satisfy $LSB(\Delta s^4) = 0$. The criteria for not rejecting the hypothesis is that t is smaller or equal to a fixed value k. The probability of Type I error is

$$\alpha = \sum_{i=k+1}^{n} \binom{n}{i} p^i (1-p)^{n-i}$$

Type II error is

$$\beta = \sum_{i=0}^{k} \binom{n}{i} 0.5^n$$

If (P, U) is a wrong key-dependent subset, the expected time complexity of checking this subset is

$$W = 2n + 2^{25}\beta \tag{15}$$

As shown in Section 2, the attack sets α smaller than or equal to 0.01 to ensure that the probability of the false rejection will not exceed 0.01. Under this precondition, the attack chooses n and β so that $\alpha < 0.01$ and minimizes Equation (15) to minimize the time complexity on the key-dependent subset (P, U). By Section 2, we minimize the total expected time complexity with this method. Because this choice is related only to the key Z_5^4, so we only need to get n and k for 2^{16} different values.

Since all key-dependent subsets have the same key fraction, the order of performing individual attacks with minimal expected time complexity becomes the ascending order of W for all key-dependent subsets due to Theorem 2.4.

Our experiment shows that the total expected time complexity of our attack is $2^{112.1}$ with 99% success probability. The number of pairs needed in one test is about 2^{19} in the worst case. The attack uses a set of 2^{21} plaintexts, which can provide 2^{20} plaintext pairs satisfying the structure in Proposition 1 for each key-dependent subset.

The attack is summarized as follows:

1. For every possible Z_5^4, calculate the corresponding number of plaintext pairs needed n and the criteria of not rejecting the hypothesis k.
2. Suppose S is an empty set. Randomly enumerate a 16-bit word s, insert s and $s \oplus 8000_x$ into the set S. Repeat this enumeration until set S contains 2^5 different words. Ask for the encryption of all the plaintexts of the form (A, B, C, D), where A and C are fixed to two arbitrary constants, B takes all the values in S and D takes all the 16-bit possible values.
3. Enumerate the key-dependent sets in ascending order of W:
   (a) Randomly choose a set of plaintext pairs with cardinality n from the known encryptions. The plaintext pairs must satisfy the requirements of Proposition 5.2.
   (b) Partially decrypt all the selected encryption pairs and count the occurrence of $LSB(\Delta s_4) = 0$.
   (c) Test the hypothesis. If the hypothesis is not rejected, perform exhaustive search for the remaining 25 key bits.

5.2 Attack on 6-Round Variant of IDEA

We now extend the 5.5-round attack to an attack on the 6-round variant of IDEA starting before the MA layer of the second round. The data complexity of the attack is 2^{49} and the time complexity is $2^{112.1}$.

As shown in [6], Z_5^2 and Z_6^2 are included in the 103 key bits in the 5.5-round attack. Hence, we can add this half round to the 5.5-round attack without enlarging the time complexity.

It is more difficult to construct right structures satisfying Proposition 5.2. Consider a pair of intermediate values X^3 and X'^3 before the third round, which satisfy Proposition 1. If we partially decrypt X^3 and X'^3 using any possible Z_5^2 and Z_6^2, the only fact we know is that all the results have the same XOR of the first and third words. The attack hence selects all the plaintexts P where the least 15 significant bits of $P_1 \oplus P_3$ are fixed to an arbitrary 15-bit constant. The total number of selected plaintexts is 2^{49}. It is possible to provide 2^{48} plaintext pairs in the test for any Z_5^2, Z_6^2 and Z_4^3. This number is sufficient in any situation.

5.3 Attack on 5-Round IDEA

We now apply the similar technique to the 5-round IDEA starting from the first round. Biryukov-Demirci Equation is reduced to

$$\begin{aligned} LSB(\Delta s^2) = LSB(&P_2 \oplus P_3 \oplus P_2' \oplus P_3' \oplus C_2 \\ &\oplus C_3 \oplus C_2' \oplus C_3' \oplus \Delta s^1 \oplus \Delta s^3 \oplus \Delta s^4 \oplus \Delta s^5) \end{aligned} \tag{16}$$

We choose the plaintext pairs to satisfy Proposition 5.2 before the first round by guessing Z_4^1, and then Δs^1 is equal to 0 as shown in Section 5.1. In order to determine the right hand side of Equation (16), we need to know Z_5^3, Z_1^4, Z_2^4, Z_5^4, Z_6^4, Z_1^5, Z_2^5, Z_3^5, Z_4^5, Z_5^5, Z_6^5. These 12 subkeys take the bits 75-65 from key Z. These 119 bits only cover the most significant nine bits of Z_5^2, which determines the probability distribution of $LSB(\Delta s^2)$. It is not necessary to guess the complete subkey Z_5^2. The attack continues to guess the least significant bit of Z_5^2(72-nd bit of Z), and estimates the probability of $LSB(\Delta s^2) = 0$ by Remark 4.2 instead. Hence, the attack guesses 120 key bits, and performs the individual attack on each guess. For the key bits of which the probability can not be approximated by Remark 4.2 as shown in Section 4, the attack exhaustively searches the remaining key bits. For other guesses, the attack uses statistical hypothesis test method to determine the remaining eight key bits.

In this attack, it is possible that the expected time of individual attacks may be larger than exhaustively search directly for some guessed key bits, which means

$$2n + \beta \cdot 2^8 \geq 2^8 \tag{17}$$

Under this condition, the attack also uses exhaustive key search to determine the remaining eight key bits to make sure the time needed not exceed exhaustive search.

This attack also choose $\alpha \leq 0.01$ to ensure that the attack successes with 99% probability. In this case, the expected time is $2^{125.5}$ encryptions.

Our experiment shows that the attack needs at most 75 pairs of encryptions for one test. We ask for 2^{17} encryptions which can provide 2^{16} pairs of encryptions, which is sufficient for the test. This data complexity(2^{17}) is the least out of all the known attacks on the 5-round IDEA starting from the first round.

In the second attack, we try to obtain the plaintext pairs satisfying the Proposition 5.2 before the second round. In order to determine $LSB(\Delta s^3)$, we need to know the least significant bits of Δs^1, Δs^2, Δs^4 and Δs^5. Hence, the subkeys we need to know are Z_1^1, Z_2^1, Z_3^1, Z_4^1, Z_5^1, Z_6^1, Z_4^2, Z_5^3. Z_5^4, Z_1^5, Z_2^5, Z_5^5 and

Z_6^5. These 13 subkeys only cover 107 bits of key Z(0-106). For every guessed 107 key bits, we use similar technique as before. The expected time complexity is $2^{115.3}$, which is the least time complexity out of all the known attacks on the 5-round IDEA starting from the first round.

Because it is not possible to predict the plaintext pair which produces the intermediate values satisfying Theorem 4.1 before the second round, the encryptions of all the 2^{64} plaintexts are required.

6 Conclusion

In this paper, we formalized a scheme of identifying the actual key using the key-dependent distribution, called key-dependent attack. How to minimize the time complexity of the key-dependent attack was also discussed. With the key-dependent attack, we could improve known cryptanalysis results and obtain more powerful attacks. We presented two key-dependent attacks on IDEA. Our attack on 5.5-round and 6-round variant of IDEA has the least time and data complexities compared with other attacks.

We only implemented a tentative exploration of the key-dependent distribution. How to make full use of the key-dependent distribution, especially how to use the key-dependent distribution to improve existing attacks, is worth further studying.

The attack on IDEA makes use of the relation between XOR, modular addition and modular multiplication. We believe that the operation XOR and modular multiplication have more properties that can be explored further [10]. Similar relations among other operations are also valuable to research. The way of making full use of the Biryukov-Demirci Equation to improve attacks on IDEA is also interesting.

References

[1] Eyüp Serdar Ayaz and Ali Aydin Selçuk. Improved DST Cryptanalysis of IDEA. In Eli Biham and Amr M. Youssef, editors, *Selected Areas in Cryptography*, volume 4356 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2006.

[2] Ishai Ben-Aroya and Eli Biham. Differential Cryptanalysis of Lucifer. *J. Cryptology*, 9(1):21–34, 1996.

[3] Eli Biham, Alex Biryukov, and Adi Shamir. Miss in the Middle Attacks on IDEA and Khufu. In Lars R. Knudsen, editor, *FSE*, volume 1636 of *Lecture Notes in Computer Science*, pages 124–138. Springer, 1999.

[4] Eli Biham, Orr Dunkelman, and Nathan Keller. Related-Key Boomerang and Rectangle Attacks. In Ronald Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 507–525. Springer, 2005.

[5] Eli Biham, Orr Dunkelman, and Nathan Keller. New Cryptanalytic Results on IDEA. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT*, volume 4284 of *Lecture Notes in Computer Science*, pages 412–427. Springer, 2006.

[6] Eli Biham, Orr Dunkelman, and Nathan Keller. A New Attack on 6-Round IDEA. In Alex Biryukov, editor, *FSE*, volume 4593 of *Lecture Notes in Computer Science*, pages 211–224. Springer, 2007.

[7] Eli Biham and Adi Shamir. Differential Cryptanalysis of DES-like Cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *CRYPTO*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 1990.

[8] Alex Biryukov, Jorge Nakahara Jr., Bart Preneel, and Joos Vandewalle. New Weak-Key Classes of IDEA. In Robert H. Deng, Sihan Qing, Feng Bao, and Jianying Zhou, editors, *ICICS*, volume 2513 of *Lecture Notes in Computer Science*, pages 315–326. Springer, 2002.

[9] Johan Borst, Lars R. Knudsen, and Vincent Rijmen. Two Attacks on Reduced IDEA. In Walter Fumy, editor, *EUROCRYPT*, volume 1233 of *Lecture Notes in Computer Science*, pages 1–13. Springer, 1997.

[10] Scott Contini, Ronald L. Rivest, Matthew J. B. Robshaw, and Yiqun Lisa Yin. Improved Analysis of Some Simplified Variants of RC6. In Lars R. Knudsen, editor, *FSE*, volume 1636 of *Lecture Notes in Computer Science*, pages 1–15. Springer, 1999.

[11] Joan Daemen, René Govaerts, and Joos Vandewalle. Weak Keys for IDEA. In Douglas R. Stinson, editor, *CRYPTO*, volume 773 of *Lecture Notes in Computer Science*, pages 224–231. Springer, 1993.

[12] Hüseyin Demirci. Square-like Attacks on Reduced Rounds of IDEA. In Kaisa Nyberg and Howard M. Heys, editors, *Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 147–159. Springer, 2002.

[13] Hüseyin Demirci, Ali Aydin Selçuk, and Erkan Türe. A New Meet-in-the-Middle Attack on the IDEA Block Cipher. In Mitsuru Matsui and Robert J. Zuccherato, editors, *Selected Areas in Cryptography*, volume 3006 of *Lecture Notes in Computer Science*, pages 117–129. Springer, 2003.

[14] Philip Hawkes. Differential-Linear Weak Key Classes of IDEA. In Kaisa Nyberg, editor, *EUROCRYPT*, volume 1403 of *Lecture Notes in Computer Science*, pages 112–126. Springer, 1998.

[15] Philip Hawkes and Luke O'Connor. On Applying Linear Cryptanalysis to IDEA. In Kwangjo Kim and Tsutomu Matsumoto, editors, *ASIACRYPT*, volume 1163 of *Lecture Notes in Computer Science*, pages 105–115. Springer, 1996.

[16] Jorge Nakahara Jr., Bart Preneel, and Joos Vandewalle. The Biryukov-Demirci Attack on Reduced-Round Versions of IDEA and MESH Ciphers. In Huaxiong Wang, Josef Pieprzyk, and Vijay Varadharajan, editors, *ACISP*, volume 3108 of *Lecture Notes in Computer Science*, pages 98–109. Springer, 2004.

[17] Pascal Junod. On the Optimality of Linear, Differential, and Sequential Distinguishers. In Eli Biham, editor, *EUROCRYPT*, volume 2656 of *Lecture Notes in Computer Science*, pages 17–32. Springer, 2003.

[18] Pascal Junod. New Attacks Against Reduced-Round Versions of IDEA. In Henri Gilbert and Helena Handschuh, editors, *FSE*, volume 3557 of *Lecture Notes in Computer Science*, pages 384–397. Springer, 2005.

[19] John Kelsey, Bruce Schneier, and David Wagner. Key-Schedule Cryptoanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES. In Neal Koblitz, editor, *CRYPTO*, volume 1109 of *Lecture Notes in Computer Science*, pages 237–251. Springer, 1996.

[20] Lars R. Knudsen and Vincent Rijmen. On the Decorrelated Fast Cipher (DFC) and Its Theory. In Lars R. Knudsen, editor, *FSE*, volume 1636 of *Lecture Notes in Computer Science*, pages 81–94. Springer, 1999.

[21] Xuejia Lai. *On the design and security of block ciphers*. ETH Series in Information Processing, Konstanz: Harturg-Gorre Verlag, 1992.

[22] Xuejia Lai and James L. Massey. A Proposal for a New Block Encryption Standard. In Ivan Damgård, editor, *EUROCRYPT*, volume 473 of *Lecture Notes in Computer Science*, pages 389–404. Springer, 1990.

[23] Mitsuru Matsui. Linear Cryptoanalysis Method for DES Cipher. In G. Goos and J. Hartmanis, editors, *EUROCRYPT*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

[24] Willi Meier. On the Security of the IDEA Block Cipher. In G. Goos and J. Hartmanis, editors, *EUROCRYPT*, volume 765 of *Lecture Notes in Computer Science*, pages 371–385. Springer, 1993.

[25] Håvard Raddum. Cryptanalysis of IDEA-X/2. In Thomas Johansson, editor, *FSE*, volume 2887 of *Lecture Notes in Computer Science*, pages 1–8. Springer, 2003.

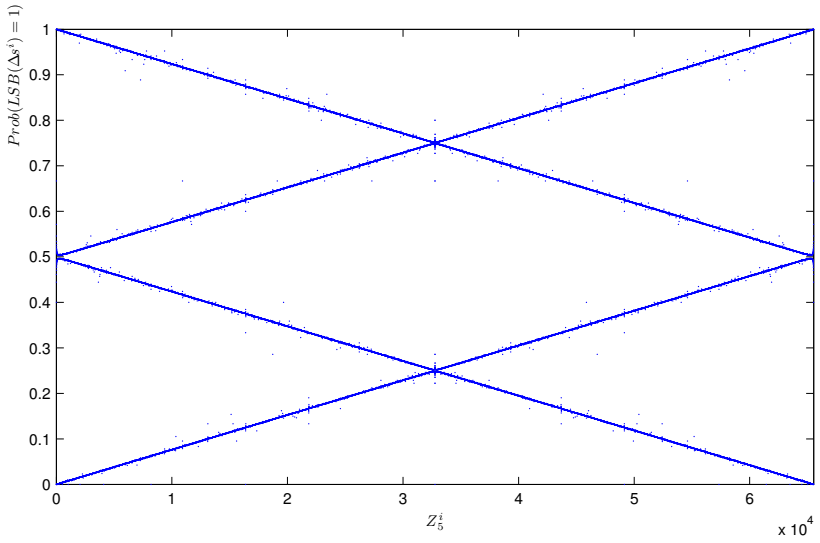

Figure 2: The key-dependent distribution of $Prob(LSB(\Delta s) = 1)$ on the value of Z_5^i