New Observations on Impossible Differential Cryptanalysis of Reduced-Round Camellia*

Ya Liu[1], Leibo Li[2,3]**, Dawu Gu[1], Xiaoyun Wang[2,3,4], Zhiqiang Liu[1], Jiazhe Chen[2,3], Wei Li[5,6]

[1] Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai 200240, China
{liuya0611,dwgu,ilu_zq}@sjtu.edu.cn
[2] Key Laboratory of Cryptologic Technology and Information Security,
Ministry of Education, Shandong University, Jinan 250100, China
[3] School of Mathematics, Shandong University, Jinan 250100, China
{lileibo, jiazhechen}@mail.sdu.edu.cn
[4] Institute for Advanced Study, Tsinghua University, Beijing 100084, China
xiaoyunwang@mail.tsinghua.edu.cn
[5] School of Computer Science and Technology, Donghua University, Shanghai 201620, China
[6] Shanghai Key Laboratory of Integrate Administration Technologies
for Information Security, Shanghai 200240, China
liwei.cs.cn@gmail.com

Abstract. Camellia is one of the widely used block ciphers, which has been selected as an international standard by ISO/IEC. In this paper, by studying the properties of the key-dependent transformations FL/FL^{-1}, we improve the previous results on impossible differential cryptanalysis of reduced-round Camellia and gain some new observations. First, we introduce some new 7-round impossible differentials of Camellia for weak keys. These weak keys that work for the impossible differential take 3/4 of the whole key space, therefore, we further get rid of the weak-key assumption and leverage the attacks on reduced-round Camellia to all keys by utilizing a method that is called the multiplied method. Second, we build a set of differentials which contains at least one 8-round impossible differential of Camellia with two FL/FL^{-1} layers. Following this new result, we show that the key-dependent transformations inserted in Camellia cannot resist impossible differential cryptanalysis effectively. Based on these 8-round impossible differential, we present a new cryptanalytic strategy to mount impossible differential attacks on reduced-round Camellia.

Key words: Block Cipher, Camellia, Impossible Differential Cryptanalysis

1 Introduction

The block cipher Camellia was proposed by NTT and Mitsubishi in 2000 [1]. It was selected as an e-government recommended cipher by CRYPTREC in 2002 [4] and the NESSIE block cipher portfolio in 2003 [19]. In 2005, it was adopted as the international standard by ISO/IEC [6]. Camellia is a 128-bit block cipher. It supports variable key sizes and the number of the rounds depends on the key size, i.e., 18 rounds for a 128-bit key size and 24 rounds for 192/256-bit key sizes. For simplicity, they can be usually denoted as Camellia-128, Camellia-192 and Camellia-256, respectively. Camellia adopts the basic Feistel structure with some key-dependent functions FL/FL^{-1} inserted every six rounds, where these key-dependent transformations must be linear and reversible for any fixed key. The goals for such a design are to provide non-regularity across rounds and to thwart further unknown attacks.

* The authors Ya Liu, Dawu Gu, Zhiqiang Liu and Wei Li are supported by the National Natural Science Foundation of China (No. 61073150 and No. 61003278), the Opening Project of Shanghai Key Laboratory of Integrate Administration Technologies for Information Security and the Fundamental Research Funds for the Central Universities. The authors Leibo Li, Xiaoyun Wang and Jiazhe Chen are supported by the National Natural Science Foundation of China (Grant No. 61133013 and No. 60931160442), and Tsinghua University Initiative Scientific Research Program (2009THZ01002).

** Corresponding author.

Up to now, many cryptanalytic methods were used to evaluate the security of reduced-round Camellia such as linear cryptanalysis, differential cryptanalysis, higher order differential attack, truncated differential attack, collision attack, square attack and impossible differential attack. Among them, most attacks focused on the security of simplified versions of Camellia, which did not take the FL/FL^{-1} and whitening layers into account [9–11, 15–18, 20–23], and only a few involved in the study of the original Camellia. For instance, Hatano *et al.* gave an higher order differential attack on the last 11 rounds of Camellia-256 [5], Chen *et al.* constructed a 6-round impossible differential with FL/FL^{-1} layer to attack 10-round Camellia-192 and 11-round Camellia-256 [3], Liu *et al.* attacked 11-round Camellia-192 and 12-round Camellia-256 by constructing a 7-round impossible differential [14]. Li *et al.* presented impossible differential attacks on 10-round Camellia-192 and 11-round Camellia-256 with a 7-round impossible differential including two FL/FL^{-1} layers [12].

Impossible differential cryptanalysis was independently introduced by Biham [2] and Knudsen [7], which is one of the most popular cryptanalytic tool. In order to mount an attack, the adversary tries to seek for an input difference that can never result in an output difference. The differential which connects the input and output difference is impossible and called an impossible differential. When the adversary wants to launch an impossible differential attack on a block cipher, she adds rounds before and/or after the impossible differential, and collect enough pairs with required plaintext and ciphertext differences. Then she concludes that the guessed subkey bits in added rounds must be wrong, if there is a pair meets the input and output values of the impossible differential under these subkey bits. In this way, she discards as many wrong keys as possible and exhaustively searches the rest of the keys.

In this paper, we reevaluate the security of reduced-round Camellia with FL/FL^{-1} and whitening layers against impossible differential cryptanalysis from two aspects. On the one hand, we first construct some new 7-round impossible differentials of Camellia for weak keys, which work for 75% of the keys. Based on them, we mount an impossible differential attack on Camellia in the weak-key setting. Then we further propose a multiplied method to extend our attacks for the whole key space. The basic idea is that if the correct key belongs to the set of weak keys, then it will never satisfy the impossible differential. While if the correct key is not a weak key, we get 2-bit conditions about the key. Specifically, for the whole key space, we present an attack on 10-round Camellia-128 with about $2^{113.8}$ chosen plaintexts and 2^{120} 10-round encryptions, 11-round Camellia-192 with about $2^{114.64}$ chosen plaintexts and 2^{184} 11-round encryptions as well as 12-round Camellia-256 with about $2^{116.17}$ chosen plaintexts or chosen ciphertexts and 2^{240} 12-round encryptions, respectively. Meanwhile, we can also extend the attacks to 12-round Camellia-192 and 14-round Camellia-256 with two FL/FL^{-1} layers. On the other hand, by studying some properties of key-dependent functions FL/FL^{-1}, we build a set of differentials which contains at least one 8-round impossible differential of Camellia with two FL/FL^{-1} layers. The length of these impossible differentials with two FL/FL^{-1} layers is the same as the length of the longest known impossible differential of Camellia without FL/FL^{-1} layers given by Wu and Zhang [23]. Consequently, we show that the key-dependent transformations inserted in Camellia cannot resist impossible differential cryptanalysis effectively. On the basis of this differential set, we propose a new cryptanalytic strategy to attack 11-round Camellia-128 with 2^{122} chosen plaintexts and 2^{122} 11-round encryptions, 12-round Camellia-192 with 2^{123} chosen plaintexts and $2^{187.2}$ 12-round encryptions as well as 13-round Camellia-256 with 2^{123} chosen plaintexts and $2^{251.1}$ 13-round encryptions (not from the first round but with the whitening layers), respectively. In table 1, we summarize our results along with the former known ones on reduced-round Camellia.

The remainder of this paper is organized as follows. Section 2 gives some notations and a brief introduction of Camellia. Section 3 first presents 7-round impossible differentials of Camellia

Table 1. Summary of the attacks on Reduced-Round Camellia

Key Size	Rounds	Attack Type	Data	Time(Enc)	Memory (Bytes)	Source
Camellia-128	9†	Square	2^{48}CP	2^{122}	2^{53}	[10]
	10†	Impossible DC	2^{118}CP	2^{118}	2^{93}	[17]
	10†	Impossible DC	$2^{118.5}$CP	$2^{123.5}$	2^{127}	[12]
	10(Weak Key)	Impossible DC	$2^{111.8}$CP	$2^{111.8}$	$2^{84.8}$	Section 3.2
	10	Impossible DC	$2^{113.8}$CP	2^{120}	$2^{84.8}$	Section 3.2
	11	Impossible DC	2^{122}CP	2^{122}	2^{102}	Section 4.4
Camellia-192	10	Impossible DC	2^{121}CP	$2^{175.3}$	$2^{155.2}$	[3]
	10	Impossible DC	$2^{118.7}$CP	$2^{130.4}$	2^{135}	[12]
	11†	Impossible DC	2^{118}CP	$2^{163.1}$	2^{141}	[17]
	11(Weak Key)	Impossible DC	$2^{112.64}$CP	$2^{146.54}$	$2^{141.64}$	Section 3.3
	11	Impossible DC	$2^{114.64}$CP	2^{184}	$2^{141.64}$	Section 3.3
	12	Impossible DC	2^{123}CP	$2^{187.2}$	2^{160}	Section 4.3
	12†	Impossible DC	$2^{120.1}$CP	2^{184}	$2^{124.1}$	Section 3.5
Camellia-256	last 11 rounds	High Order DC	2^{93}CP	$2^{255.6}$	2^{98}	[5]
	11	Impossible DC	2^{121}CP	$2^{206.8}$	2^{166}	[3]
	11	Impossible DC	$2^{119.6}$CP	$2^{194.5}$	2^{135}	[12]
	12(Weak Key)	Impossible DC	$2^{121.12}$CP	$2^{202.55}$	$2^{142.12}$	Section 3.4
	12	Impossible DC	$2^{116.17}$CP/CC	2^{240}	$2^{150.17}$	Section 3.4
	13	Impossible DC	2^{123}CP	$2^{251.1}$	2^{208}	Section 4.2
	14†	Impossible DC	2^{120}CC	$2^{250.5}$	2^{125}	Section 3.5

DC: Differential Cryptanalysis; CP/CC: Chosen Plaintexts/Chosen Ciphertexts;
Enc: Encryptions; †: The attack doesn't include the whitening layers.

for weak keys. Based on them, impossible differential attacks on 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256 are elaborated. Section 4 first constructs a set of differentials which contains at least one 8-round impossible differential of Camellia with two FL/FL^{-1} layers, and then proposes impossible differential attacks on 11-round Camellia-128, 12-round Camellia-192 and 13-round Camellia-256, respectively. Section 5 summarizes this paper.

2 Preliminaries

2.1 Some Notations

- P, C: the plaintext and the ciphertext;
- L_{i-1}, R_{i-1}: the left half and the right half of the i-th round input;
- $\Delta L_{i-1}, \Delta R_{i-1}$: the left half and the right half of the input difference in the i-th round;
- $X \mid Y$: the concatenation of X and Y;
- $kw_1|kw_2, kw_3|kw_4$: the pre-whitening key and the post-whitening key;
- k_i: the subkey used in the i-th round;
- $kl_i (1 \leq i \leq 6)$: 64-bit keys used in the FL/FL^{-1} layers;
- $S_r, \Delta S_r$: the output and the output difference of the S-boxes in the r-th round;
- $X \lll j$: left rotation of X by j bits;
- $X_{L(\frac{n}{2})}, X_{R(\frac{n}{2})}$: the left half and the right half of a n-bit word X;
- $X_i, X_{\{i,j\}}, X_{\{i\sim j\}}$: the i-th byte, the i-th and j-th bytes and the i-th to the j-th bytes of X;
- $X^i, X^{(i,j)}, X^{(i\sim j)}$: the i-th bit, the i-th and j-th bits and the i-th to j-th bits of X;
- $\oplus, \cap, \cup$: bitwise exclusive-OR (XOR), AND, and OR operations, respectively;
- $0_{(i)}, 1_{(i)}$: consecutive i bits are zero or one.

2.2 Overview of Camellia

Camellia [1] is a 128-bit block cipher. Two keyed functions FL/FL^{-1} are inserted every 6 rounds. Camellia uses variable key sizes and the number of rounds depends on the key size, i.e.,

18 rounds for a 128-bit key size and 24 rounds for 192/256-bit key sizes. The round function of Camellia uses a SPN structure. Among it, the linear transformation P and its inverse function P^{-1} are defined as follows.

$$P : (\{0,1\}^8)^8 \rightarrow (\{0,1\}^8)^8, y_1 \mid y_2 \mid y_3 \mid y_4 \mid y_5 \mid y_6 \mid y_7 \mid y_8 \rightarrow z_1 \mid z_2 \mid z_3 \mid z_4 \mid z_5 \mid z_6 \mid z_7 \mid z_8;$$

$$\begin{aligned}
z_1 &= y_1 \oplus y_3 \oplus y_4 \oplus y_6 \oplus y_7 \oplus y_8; & y_1 &= z_2 \oplus z_3 \oplus z_4 \oplus z_6 \oplus z_7 \oplus z_8;\\
z_2 &= y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_7 \oplus y_8; & y_2 &= z_1 \oplus z_3 \oplus z_4 \oplus z_5 \oplus z_7 \oplus z_8;\\
z_3 &= y_1 \oplus y_2 \oplus y_3 \oplus y_5 \oplus y_6 \oplus y_8; & y_3 &= z_1 \oplus z_2 \oplus z_4 \oplus z_5 \oplus z_6 \oplus z_8;\\
z_4 &= y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7; & y_4 &= z_1 \oplus z_2 \oplus z_3 \oplus z_5 \oplus z_6 \oplus z_7;\\
z_5 &= y_1 \oplus y_2 \oplus y_6 \oplus y_7 \oplus y_8; & y_5 &= z_1 \oplus z_2 \oplus z_5 \oplus z_7 \oplus z_8;\\
z_6 &= y_2 \oplus y_3 \oplus y_5 \oplus y_7 \oplus y_8; & y_6 &= z_2 \oplus z_3 \oplus z_5 \oplus z_6 \oplus z_8;\\
z_7 &= y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_8; & y_7 &= z_3 \oplus z_4 \oplus z_5 \oplus z_6 \oplus z_7;\\
z_8 &= y_1 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7; & y_8 &= z_1 \oplus z_4 \oplus z_6 \oplus z_7 \oplus z_8;
\end{aligned}$$

The key-dependent function $FL : \{0,1\}^{64} \times \{0,1\}^{64}$ maps $(X_L \mid X_R, kl_L \mid kl_R) \mapsto Y_L \mid Y_R$, where $Y_R = ((X_L \cap kl_L) \lll 1) \oplus X_R, Y_L = (Y_R \cup kl_R) \oplus X_L$.

Key Schedule of Camellia The key schedule algorithm of Camellia applies a 6-round Feistel structure to generate two 128-bit intermediate variables K_A and K_B. These two variables K_A and K_B can be calculated by two 128-bit variables K_L and K_R defined by the main key K. For Camellia-128, the 128-bit key K is used as K_L and K_R is 0. For Camellia-192, the left 128-bit of the key K is used as K_L, and the concatenation of the right 64-bit of the key K and the complement of the right 64-bit of the key K is used as K_R. For Camellia-256, the main key K is separated into two 128-bit variables K_L and K_R, i.e., $K = K_L \mid K_R$.

3 7-Round Impossible Differentials of Camellia for Weak Keys and Their Applications [1]

In this section, we first construct some 7-round impossible differentials of Camellia in weak-key setting. Based on them, we present impossible differential attacks on 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256 which start from the first round. In addition, we can also extend the attack to 12-round Camellia-192 and 14-round Camellia-256 with two FL/FL^{-1} layers.

3.1 7-Round Impossible Differentials of Camellia for Weak Keys

This section introduces 7-round impossible differentials of Camellia in weak-key setting, which is based on the following propositions.

Lemma 1 ([8]). *Let X, X', K be l-bit values, and $\Delta X = X \oplus X'$, then the differential properties of AND and OR operations are:*
$(X \cap K) \oplus (X' \cap K) = (X \oplus X') \cap K = \Delta X \cap K$,
$(X \cup K) \oplus (X' \cup K) = (X \oplus K \oplus (X \cap K)) \oplus (X' \oplus K \oplus (X' \cap K)) = \Delta X \oplus (\Delta X \cap K)$.

Lemma 2 ([3]). *Let ΔX and ΔY be the input and output differences of FL. Then*

$$\begin{aligned}
\Delta Y_R &= ((\Delta X_L \cap kl_L) \lll 1) \oplus \Delta X_R, & \Delta Y_L &= \Delta X_L \oplus \Delta Y_R \oplus (\Delta Y_R \cap kl_R);\\
\Delta X_L &= \Delta Y_L \oplus \Delta Y_R \oplus (\Delta Y_R \cap kl_R), & \Delta X_R &= ((\Delta X_L \cap kl_L) \lll 1) \oplus \Delta Y_R.
\end{aligned}$$

[1] By Leibo Li, Xiaoyun Wang and Jiazhe Chen. See [13] for more details.

Proposition 1. *If the output difference of FL function is $\Delta Y = (0|0|0|0|d|0|0|0)$, where $d \neq 0$ and $d^{(1)} = 0$, then the input difference of FL function should satisfy $\Delta X_{\{2,3,4,6,7,8\}} = 0$.*

Proposition 2. *If the output difference of FL^{-1} function is $\Delta X = (0|e|e|e|0|e|e|e)$, and the subkeys of FL^{-1} function satisfy that $KL_L^{(9)}$ is 0 or $KL_R^{(8)}$ is 1, then the first byte of input difference ΔY should be zero, where e is a non-zero byte.*

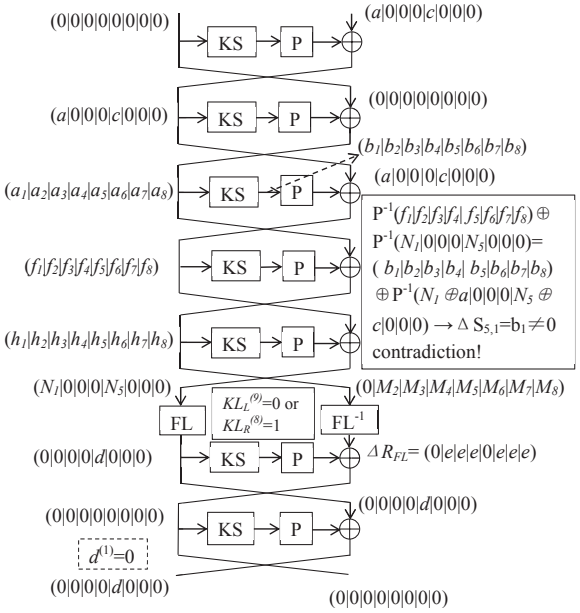

Fig. 1. A 7-Round Impossible Differential for Weak Keys

Proposition 3. *Given a 7-round Camellia encryption and a FL/FL^{-1} layer inserted between the fifth and sixth round. If the input difference of the first round is $(0|0|0|0|0|0|0|0, a|0|0|0|c|0|0|0)$, and the subkeys of FL^{-1} function satisfy $KL_L^{(9)} = 0$ or $KL_R^{(8)} = 1$, then the output difference $(0|0|0|0|d|0|0|0, 0|0|0|0|0|0|0|0)$ with $d^{(1)} = 0$ is impossible, where a and d are non-zero bytes, c is an arbitrary value (see Fig. 1).*

Proof. First, we analyze the forward direction. It is trivial that $(\Delta L_1, \Delta R_1) = (a|0|0|0|c|0|0|0, 0|0|0|0|0|0|0|0)$, then it propagates to $(\Delta L_2, \Delta R_2) = (a_1|a_2|a_3|a_4|a_5|a_6|a_7|a_8, a|0|0|0|c|0|0|0)$ after the second round, where a_1 and a_5 are non-zero values, a_i $(i = 2, 3, 4, 6, 7, 8)$ are unknown values. Getting through the key addition and substitution layers of the third round, the output difference of S-box layer in the third round is $\Delta S_3 = (b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8)$, where b_1 and b_5 are non-zero values. Then we have $(\Delta L_3, \Delta R_3) = (f_1|f_2|f_3|f_4|f_5|f_6|f_7|f_8, a_1|a_2|a_3|a_4|a_5|a_6|a_7|a_8)$, and $(\Delta L_4, \Delta R_4) = (h_1|h_2|h_3|h_4|h_5|h_6|h_7|h_8, f_1|f_2|f_3|f_4\ |f_5|f_6|f_7|f_8)$, where f_i, h_i are unknown values.

Second, we consider the backward direction. The output difference of the seventh round is $(\Delta L_7, \Delta R_7) = (0|0|0|0|d|0|0|0, 0|0|0|0|0|0|0|0)$, then the output difference of the sixth round is $(\Delta L_6, \Delta R_6) = (0|0|0|0|0|0|0|0, 0|0|0|0|d|0|0|0)$, and the output difference of FL/FL^{-1} layer is $(0|0|0|0|d|0|0|0, 0|e|e|e|0|e|e|e)$. According to the condition $d^{(1)} = 0$ and Proposition 1, we obtain that the input difference of FL function is $(N_1|0|0|0|N_5|0|0|0)$. Since $KL_L^{(9)} = 0$ or $KL_R^{(8)} = 1$,

in the light of Proposition 2, the input difference of FL^{-1} function is $(0|M_2|M_3|M_4|M_5|M_6|M_7|M_8)$, which means $\Delta L_{4,1} = h_1 = 0$. Where N_1, N_5 and M_i $(i = 2, ..., 8)$ are unknown bytes.

Finally, we focus on the fifth round. The output difference of S-layer in the fifth round is

$$\begin{aligned}\Delta S_5 &= P^{-1}(f_1|f_2|f_3|f_4|f_5|f_6|f_7|f_8) \oplus P^{-1}(N_1|0|0|0|N_5|0|0|0) \\ &= (b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8) \oplus P^{-1}(N_1 \oplus a|0|0|0|N_5 \oplus c|0|0|0).\end{aligned}$$

Then $\Delta S_{5,1} = b_1 \neq 0$, which contradicts $\Delta L_{4,1} = 0$. □

We also obtain three other impossible differentials under different weak-key assumptions:

- $(0|0|0|0|0|0|0|0, 0|a|0|0|0|c|0|0) \nrightarrow (0|0|0|0|0|d|0|0, 0|0|0|0|0|0|0|0)$ with conditions $KL_L^{(17)} = 0$ or $KL_R^{(16)} = 1$, and $d^{(1)} = 0$,
- $(0|0|0|0|0|0|0|0, 0|0|a|0|0|0|c|0) \nrightarrow (0|0|0|0|0|0|d|0, 0|0|0|0|0|0|0|0)$ with conditions $KL_L^{(25)} = 0$ or $KL_R^{(24)} = 1$, and $d^{(1)} = 0$,
- $(0|0|0|0|0|0|0|0, 0|0|0|a|0|0|0|c) \nrightarrow (0|0|0|0|0|0|0|d, 0|0|0|0|0|0|0|0)$ with conditions $KL_L^{(1)} = 0$ or $KL_R^{(32)} = 1$, and $d^{(1)} = 0$.

We denote this type of impossible differentials above as **5+2 WKID** (weak-key impossible differentials). Due to the feature of Feistel structure, we also deduce another type of 7-round impossible differentials with the FL/FL^{-1} layers inserted between the second and the third rounds. We call them **2+5 WKID**, which are depicted as follows.

- $(0|0|0|0|0|0|0|0, 0|0|0|0|d|0|0|0) \nrightarrow (a|0|0|0|c|0|0|0, 0|0|0|0|0|0|0|0)$ with conditions $KL'^{(9)}_L = 0$ or $KL'^{(8)}_R = 1$, and $d^{(1)} = 0$,
- $(0|0|0|0|0|0|0|0, 0|0|0|0|0|d|0|0) \nrightarrow (0|a|0|0|0|c|0|0, 0|0|0|0|0|0|0|0)$ with conditions $KL'^{(17)}_L = 0$ or $KL'^{(16)}_R = 1$, and $d^{(1)} = 0$,
- $(0|0|0|0|0|0|0|0, 0|0|0|0|0|0|d|0) \nrightarrow (0|0|a|0|0|0|c|0, 0|0|0|0|0|0|0|0)$ with conditions $KL'^{(25)}_L = 0$ or $KL'^{(24)}_R = 1$, and $d^{(1)} = 0$,
- $(0|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|d) \nrightarrow (0|0|0|a|0|0|0|c, 0|0|0|0|0|0|0|0)$ with conditions $KL'^{(1)}_L = 0$ or $KL'^{(32)}_R = 1$, and $d^{(1)} = 0$,

where KL' represents the subkey used in FL-function.

3.2 Impossible Differential Attack on 10-Round Camellia-128

We first propose an attack that works for $3 \times 2^{126} (= \frac{3}{4} \times 2^{128})$ keys, which is mounted by adding one round on the top and two rounds on the bottom of the **5+2 WKID** (See Fig. 2). The attack procedure is as follows.

Data Collection.

1. Choose 2^n structures of plaintexts, and each structure contains 2^{32} plaintexts

$$(L_0, R_0) = (\alpha_1|x_1|x_2|x_3|\alpha_2|x_4|x_5|x_6, P(\beta_1|y_1|y_2|y_3|\beta_2|y_4|y_5|y_6)),$$

   where x_i and y_i $(i = 1, ..., 6)$ are fixed values in each structure, while α_j, β_j $(j = 1, 2)$ takes all the possible values, and P is the linear transformation.
2. For each structure, ask for the encryption of the plaintexts and get 2^{32} ciphertexts. Store them in a hash table H indexed by $C_{L,\{1,5\}}$, the XOR of $C_{L,2}$ and $C_{L,3}$, the XOR of $C_{L,2}$ and $C_{L,4}$, the XOR of $C_{L,2}$ and $C_{L,6}$, the XOR of $C_{L,2}$ and $C_{L,7}$, the XOR of $C_{L,2}$ and

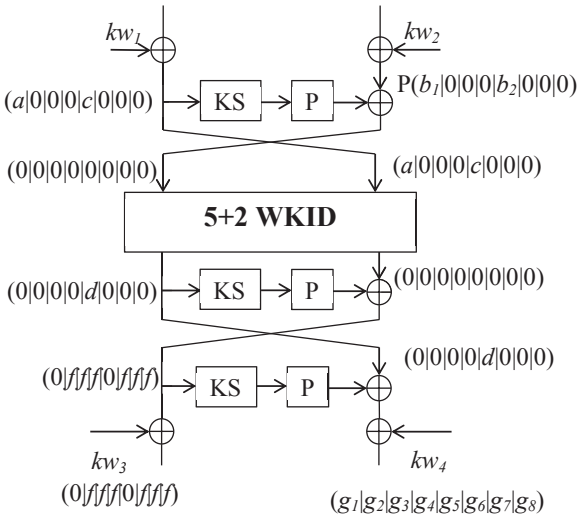

Fig. 2. Impossible Differential Attack on 10-Round Camellia-128 for Weak Keys

$C_{L,8}$. Then by birthday paradox, we get $2^{n+63} \times 2^{-56} = 2^{n+7}$ pairs of ciphertexts with the differences

$$(\Delta C_L, \Delta C_R) = (0|f|f|f|0|f|f|f, g_1|g_2|g_3|g_4|g_5|g_6|g_7|g_8),$$

and the differences of corresponding plaintext pairs satisfy

$$(\Delta L_0, \Delta R_0) = (a|0|0|0|c|0|0|0, P(b_1|0|0|0|b_2|0|0|0)),$$

where a, c, f and b_i $(i = 1, 2)$ are non-zero bytes, and g_i are unknown bytes. For every pair, compute the value

$$P^{-1}(\Delta C_R) = P^{-1}(g_1|g_2|g_3|g_4|g_5|g_6|g_7|g_8) = (g'_1|g'_2|g'_3|g'_4|g'_5|g'_6|g'_7|g'_8).$$

Keep only the pairs whose ciphertexts satisfy $g'_1 = 0$. The probability of this event is 2^{-8}, thus the expected number of remaining pairs is $2^{n+7} \times 2^{-8} = 2^{n-1}$.

Key Recovery.

1. For each pair obtained in the data collection phase, guess the 16-bit value $K_{1,\{1,5\}}$, partially encrypt its plaintext $(L_{0,\{1,5\}}, L'_{0,\{1,5\}})$ to get the intermediate value $(S_{1,\{1,5\}}, S'_{1,\{1,5\}})$ and the difference $\Delta S_{1,\{1,5\}}$. Then discard the pairs whose intermediate values do not satisfy $\Delta S_{1,1} = b_1$ and $\Delta S_{1,5} = b_2$. The probability of a pair being kept is 2^{-16}, so the expected number of remaining pairs is $2^{n-1} \times 2^{-16} = 2^{n-17}$.
2. In this step, the ciphertext of every remaining pair is considered.
   (a) Guess the 8-bit value $K_{10,8}$ for every remaining pair, partially decrypt the ciphertext $(C_{L,8}, C'_{L,8})$ to get the intermediate value $(S_{10,8}, S'_{10,8})$ and the difference $\Delta S_{10,8}$, and discard the pairs whose intermediate values do not satisfy $\Delta S_{10,8} = g'_8$. The expected number of remaining pairs is $2^{n-17} \times 2^{-8} = 2^{n-25}$.
   (b) For $l = 2, 3, 4, 6, 7$, guess the 8-bit value $K_{10,l}$. For every remaining pair, partially decrypt the ciphertext $(C_{L,l}, C'_{L,l})$ to get the intermediate value $(S_{10,l}, S'_{10,l})$ and the difference $\Delta S_{10,l}$, and keep only the pairs whose intermediate values satisfy $\Delta S_{10,l} = g'_l \oplus g'_5$. Since for each l, each pair will remain with probability 2^{-8}, the expected number of remaining pairs is $2^{n-25} \times 2^{5\times(-8)} = 2^{n-65}$.

(c) Guess the 8-bit value $K_{10,1}$, partially decrypt the ciphertext $C_{L,1}$ of every remaining pair to get the intermediate value $S_{10,1}$, which is also the value of $S'_{10,1}$.
(d) Partially decrypt (S_{10}, S'_{10}) to get the intermediate values $(R_{9,5}, R'_{9,5})$, and discard the pairs whose intermediate values do not satisfy $\Delta R^{(1)}_{9,5} = 0$. As the probability of a pair being discarded is 0.5, the expected number of remaining pairs is $2^{n-65} \times 2^{-1} = 2^{n-66}$.

3. For every remaining pair, guess the 8-bit value $K_{9,5}$, partially decrypt the output value $(R_{9,5}, R'_{9,5})$ to get the intermediate value $(S_{9,5}, S'_{9,5})$ and the difference $\Delta S_{9,5}$. If there is a pair satisfies $\Delta S_{9,5} = \Delta C_{L,2}$, we discard the guessed key and try another one. Otherwise we exhaustively search for the remaining 48 bits of the key under this guessed key, if the correct key is obtained, we halt the attack; otherwise, another key guess should be tried.

Complexity. Since the probability of the event $\Delta S_{9,5} = \Delta C_{L,2}$ happens in step 3 of key recovery phase is 2^{-8}, the expected number of remaining guesses for 72-bit target subkeys is about $\epsilon = 2^{80} \times (1 - 2^{-8})^{2^{n-66}}$. If we choose $\epsilon = 1$, then n is 79.8, and the proposed attack requires $2^{n+32} = 2^{111.8}$ chosen plaintexts. The time and memory complexities are dominated by step 2 of data collection phase, which are about $2^{111.8}$ 10-round encryptions and $2^{n-1} \times 4 \times 2^4 = 2^{84.8}$ bytes.

Extending the Attack to the Whole Key Space. On the basis of the above impossible differential attack for weak keys, we construct a multiplied attack on 10-Round Camellia-128.

– **Phase 1.** Perform an impossible differential attack by using the **5+2 WKID**

$$(0|0|0|0|0|0|0|0, a|0|0|0|c|0|0|0) \nrightarrow (0|0|0|0|d|0|0|0, 0|0|0|0|0|0|0|0).$$

This phase is extremely similar to the weak-key attack that is described above. However, it is slightly different when the attack is finished. That is, if there is a key kept, then the key is the correct key, and we halt the procedure of the attack. Otherwise, we conclude that the correct key does not belong to this set of weak keys, which means that $kl_1^{(9)} = 1$ and $kl_2^{(8)} = 0$. In this case, we get 2-bit information of the key and perform the next phase.

– **Phases 2 to 4.** Perform an impossible differential attack by using each **5+2 WKID** in the following:

$$(0|0|0|0|0|0|0|0, 0|a|0|0|0|c|0|0) \nrightarrow (0|0|0|0|0|d|0|0, 0|0|0|0|0|0|0|0),$$

$$(0|0|0|0|0|0|0|0, 0|0|a|0|0|0|c|0) \nrightarrow (0|0|0|0|0|0|d|0, 0|0|0|0|0|0|0|0),$$

$$(0|0|0|0|0|0|0|0, 0|0|0|a|0|0|0|c) \nrightarrow (0|0|0|0|0|0|0|d, 0|0|0|0|0|0|0|0).$$

The procedure is similar to Phase 1, and either recover the correct key or get another 2-bit information about the key and execute the next phase.

– **Phase 5.** Announce the intermediate key

$$K_A^{(95,103,111,119)} = 0 \text{ and } K_A^{(6,14,22,30)} = 1,$$

then exhaustively search for the remaining 120 bit value of K_A and recover the key K_L.

The upper bound of the time complexity is $2^{111.8} \times 4 + 2^{120} \approx 2^{120}$. The data complexity is about $2^{113.8}$. The memory could be reused in different phase, so the memory requirement is about $2^{84.8}$ bytes.

3.3 Attack on 11-Round Camellia-192

We add one round on the bottom of 10-round attack and give an attack on 11-round Camellia-192.

Data Collection. Choose $2^{80.64}$ structures of plaintexts. Each structure contains 2^{32} plaintexts:

$$(L_0, R_0) = (\alpha_1|x_1|x_2|x_3|\alpha_2|x_4|x_5|x_6, P(\beta_1|y_1|y_2|y_3|\beta_2|y_4|y_5|y_6)),$$

where x_i and y_i $(i = 1, ..., 6)$ are fixed values in each structure, while α_j and β_j $(j = 1, 2)$ take all the possible values, and P is the linear transformation. Ask for the encryption of the corresponding ciphertext for each plaintext, compute $P^{-1}(C_L)$ and store the plaintext-ciphertext pairs (L_0, R_0, C_L, C_R) in a hash table indexed by 8-bit value $(P^{-1}(C_L))_1$. By birthday paradox, we get $2^{143.64} \times 2^{-8} = 2^{135.64}$ pairs whose ciphertext differences satisfy $P^{-1}(\Delta C_R) = (h'_1|h'_2|h'_3|h'_4|h'_5|h'_6|h'_7|h'_8)$ and $P^{-1}(\Delta C_L) = (0|g'_2|g'_3|g'_4|g'_5|g'_6|g'_7|g'_8)$, where h'_i and g'_i are unknown values.

Key Recovery.

1. For $l = 1, 5$, guess the 8-bit value of $K_{1,l}$, partially encrypt their plaintext $(L_{0,l}, L'_{0,l})$ and discard the pairs whose intermediate value do not satisfy $\Delta S_{1,l} = (P^{-1}(\Delta R_0))_l$. The expected number of remaining pairs is $2^{135.64} \times 2^{-16} = 2^{119.64}$.
2. In this step, we consider the ciphertext of each remaining pair.
   (a) For $l = 1, 2, 3, 4, 6, 7, 8$, guess the 8-bit value of $K_{11,l}$. Partially decrypt the ciphertext $(C_{L,l}, C'_{L,l})$ and keep only the pairs which satisfy $\Delta S_{11,l} = h'_l$. The expected number of remaining pairs is $2^{119.64} \times 2^{7\times(-8)} = 2^{63.64}$.
   (b) Guess the 8-bit value $K_{11,5}$. Partially decrypt the ciphertext $(C_{L,5}, C'_{L,5})$, then compute the intermediate value (R_{10}, R'_{10}), where $\Delta R_{10} = (0|f|f|f|0|f|f|f)$ and $f = \Delta S_{11,5} \oplus h'_5$.
3. Application of the 10-round attack.
   (a) Guess the 8-bit value $K_{10,8}$, partially decrypt $(R_{10,8}, R'_{10,8})$ and discard the pairs whose intermediate values do not satisfy $\Delta S_{10,8} = g'_8$. The expected number of remaining pairs is $2^{63.64} \times 2^{-8} = 2^{55.64}$.
   (b) For $l = 2, 3, 4, 6, 7$, guess the 8-bit value $K_{10,l}$. Partially decrypt the intermediate value $(R_{10,l}, R'_{10,l})$ and keep only the pairs whose intermediate values satisfy $\Delta S_{10,l} = g'_l \oplus g'_5$. The expected number of remaining pairs is $2^{55.64} \times 2^{5\times(-8)} = 2^{15.64}$.
   (c) Guess the 8-bit value $K_{10,1}$, partially decrypt the intermediate value $R_{10,1}$ and calculate the intermediate values $(R_{9,5}, R'_{9,5})$. Discard the pairs whose intermediate values do not satisfy $\Delta R^{(1)}_{9,5} = 0$. Then the expected number of remaining pairs is $2^{15.64} \times 2^{-1} = 2^{14.64}$.
   (d) Guess the 8-bit value $K_{9,5}$, partially decrypt the intermediate value $(R_{9,5}, R'_{9,5})$ to get the difference $\Delta S_{9,5}$. If there is a pair satisfies $\Delta S_{9,5} = \Delta R_{10,2}$, we discard the guessed key and try another one. Otherwise we exhaustively search for the rest 48 bits of K_L and K_R under this key, if the correct key is obtained, we halt the attack; otherwise, another key should be tried.

Complexity. The data complexity of the attack is $2^{112.64}$ chosen plaintexts. The time complexity is dominated by step 3 (d) which requires about $2^{144} \times (1 + (1 - 2^{-8}) + (1 - 2^{-8})^2 + ... + (1 - 2^{-8})^{2^{13.7}-1}) \times 2 \times \frac{1}{11} \times \frac{1}{8} \approx 2^{146.54}$ 11-round encryptions. The memory complexity is about $2^{133.56} \times 4 \times 2^4 = 2^{141.64}$ bytes.

Reduce the Time Complexity to $2^{138.54}$. Assume 16-bit value α_2 and β_2 are fixed in data collection phase of above attack, then we can collect $2^{n+31} \times 2^{-8} = 2^{n+23}$ pairs, where n represents the number of structures. Nevertheless, it is unnecessary for us to guess 8-bit subkey $K_{1,5}$ in this case. Then there are totally 136-bit values of subkey to be guessed in the attack, therefore, the expected number of remaining guesses of target subkey is about $\epsilon = 2^{136} \times (1 - 2^{-8})^{2^{n-90}}$ after the attack. If we chose $\epsilon = 1$, n is 104.56. Then the data complexity increases to $2^{n+16} = 2^{120.56}$, but the time complexity reduces to $2^{138.54}$, the memory requirement reduces to $2^{133.56}$ bytes.

Extending the Attack to the Whole Key Space. Similar to 10-round attack on Camellia-128, we mount a multiplied attack on Camellia-192 for the whole key space. The expected time of the attack is about $4 \times 2^{146.54} + 2^{192} \times (1 - \frac{3}{4})^4 = 2^{184}$. The expected data of the attack is $2^{114.64}$. The memory requirement is about $2^{141.64}$ bytes.

3.4 The Attack on 12-Round Camellia-256

We add one round on the bottom of 11-round attack, and present a 12-round attack on Camellia-256. The attack procedure is similar to the 11-round attack. First choose $2^{81.17}$ structures and collect $2^{144.17}$ plaintext-ciphertext pairs in data collection phase. After guessing the subkey $K_{1,\{1,5\}}$, we guess the 64-bit value K_{12} and compute the intermediate value (R_{11}, R'_{11}), then apply the 11-round attack to perform the remaining steps. In summary, the proposed attack requires $2^{81.17+32} = 2^{113.17}$ chosen plaintexts. The time complexity is about $2^{210.55}$ 12-round encryptions, and the memory requirement is about $2^{150.17}$ bytes. Similar to the above subsection, the time complexity and memory requirement can also reduce to $2^{202.55}$ and $2^{142.12}$, respectively, but data complexity increases to $2^{121.12}$ in this case.

We also construct another type of impossible differential attack of Camellia-256, which adds four rounds on the top and one round on the bottom of the **2+5 WKID** (see section 3.1). The attack is performed under the chosen ciphertext attack scenario. Similar to the attack based on the **5+2 WKID**, the data and time complexity are about $2^{113.17}$ and $2^{216.3}$, respectively.

Extending the Attack to the Whole Key Space. On the basis of two types of impossible differential attacks for weak keys, we mount a multiplied attack on 12-round Camellia-256 for the whole key space as below.

- **Phases 1 to 8.** Preform an impossible differential attack by using of all conditional impossible differentials **2+5 WKID** list in section 3.1. For each phase, if success, output the actual key, else perform the next phase.
- **Phase 9.** Announce 16-bit value of the master key

$$K_R^{(31,39,47,55,95,103,111,119)} = 0 \text{ and } K_R^{(6,14,22,30,70,78,86,94)} = 1,$$

  then exhaustively search for the remaining 240 bit value of K_R, K_L and recover the actual key.

The expected time of the attack is $2^{216.3} \times 8 + 2^{256} \times (\frac{1}{4})^8 \approx 2^{240}$ encryptions, and the expected data complexity is about $2^{116.17}$.

3.5 The Attacks Including Two FL/FL^{-1} Layers

If we do not start from the first round, we can take the attacks that include two FL/FL^{-1} layers into account. We first illustrate some new observations of FL and FL^{-1} functions, then present attacks on variants of 14-round Camellia-256 and 12-round Camellia-192.

Proposition 4. *If the output difference of FL function is $\Delta Y = (a|0|0|0|0|0|0|0)$, then the input difference should satisfy $\Delta X = (b_1|0|0|0|b_5|0|0|b_8)$ with $b_1 = a$, $b_5^{(8)} = 0$ and $b_8^{(1\sim7)} = 0$, where a is a non-zero byte.*

Proposition 5. *If the output difference of FL^{-1} function is $\Delta X = (a|a|a|0|a|0|0|a)$, and the input difference $\Delta Y = (b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8)$, then $b_7^{(8)} = 0$, $b_3^{(8)} = a^{(8)}$ and $b_8^{(1\sim7)} = a^{(1\sim7)}$, where a is a non-zero byte, b_i are unknown bytes.*

Proposition 6. *Suppose the input difference of the i-round of Camellia satisfies* $(\Delta L_{i-1}, \Delta R_{i-1}) = (b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8, P(c'_1|c'_2|c'_3|c'_4|c'_5|c'_6|c'_7|c'_8))$, *and the output difference is* $(\Delta L_i, \Delta R_i) = (a_1|0|0|0|a_5|0|0|a_8, b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8)$ *with* $a_5^{(8)} = 0$ *and* $a_8^{(1\sim7)} = 0$, *where* b'_i, c'_i *are arbitrary bytes, and* a_1 *is a nonzero byte, then the following results hold.*

(1) The intermediate value $\Delta S_i = P^{-1}(\Delta L_i \oplus \Delta R_{i-1}) = (c'_1 \oplus a_8|c'_2 \oplus a_1 \oplus a_5 \oplus a_8|c'_3 \oplus a_1 \oplus a_5 \oplus a_8|c'_4 \oplus a_1 \oplus a_5|c'_5 \oplus a_1 \oplus a_5 \oplus a_8|c'_6 \oplus a_5 \oplus a_8|c'_7 \oplus a_5|c'_8 \oplus a_1 \oplus a_8)$.
(2) $\Delta S_{i,1}^{(1\sim7)} = c'^{(1\sim7)}_1$, *and* $a_8^{(8)} = \Delta S_{i,1}^{(8)} \oplus c'^{(8)}_1$.
(3) $\Delta S_{i,7}^{(8)} = c'^{(8)}_7$, *and* $a_5^{(1\sim7)} = \Delta S_{i,5}^{(1\sim7)} \oplus c'^{(1\sim7)}_7$.
(4) $a_1 = \Delta S_{i,8} \oplus c'_8 \oplus a_8$.

Attack on 14-Round Camellia-256 Our 14-round attack of Camellia-256 works from round 10 to round 23, where the **5+2 WKID** is applied from round 14 to round 20.

First of all, we demonstrate the relation of subkeys used in the round 10, 11, 12, 13, 21, 22, 23 and the second FL/FL^{-1} layer (KL_3, KL_4) as follows, i.e., $K_{10} = K_L^{(110\sim128,1\sim45)}$, $K_{11} = K_A^{(46\sim109)}$, $K_{12} = K_A^{(110\sim128,1\sim45)}$, $K_{13}^{(1\sim8)} = K_R^{(61\sim68)}$, $KL_{3,L}^{(1\sim9)} = K_L^{(61\sim69)}$, $KL_{3,R}^{(1\sim8)} = K_L^{(93\sim100)}$, $KL_{4,L} = K_L^{(125\sim128,1\sim28)}$, $KL_{4,R} = K_L^{(29\sim60)}$, $K_{21}^{(33\sim40)} = K_A^{(127,128,1\sim6)}$, $K_{22} = K_A^{(31\sim94)}$, $K_{23} = K_L^{(112\sim128,1\sim47)}$.

With the key relation, we can first launch the impossible differential attack in weak-key setting, then extend it to an attack for all keys, which is similar to above attacks.

Data Collection. We choose the chosen ciphertext scenario to perform the attack and begin with choosing one structure of ciphertexts which contains 2^{120} ciphertexts:

$$(C_L, C_R) = (P(y_1|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8), \alpha_1|\alpha_2|\alpha_3|\alpha_4|\alpha_5|\alpha_6|\alpha_7|\alpha_8).$$

Where y_1 is fixed, while α_i ($i = 1, ..., 8$) and β_j ($j = 2, ...8$) take all possible values. Ask for the decryption to get the corresponding plaintext for each ciphertext, which results in 2^{239} pairs which satisfy the difference:

$$(\Delta C_L, \Delta C_R) = (P(0|g'_2|g'_3|g'_4|g'_5|g'_6|g'_7|g'_8), f_1|f_2|f_3|f_4|f_5|f_6|f_7|f_8).$$

Key Recovery.

1. Guess 130-bit value $(K_L^{(1\sim47,110\sim128)}|K_A^{(46\sim109)})$, for every plaintext-ciphertext pair (P, C), perform the following substeps.
   (a) Partially encrypt the plaintext P to get the intermediate value (L_{11}, R_{11}). Since 38 bits of the subkey used in FL^{-1} function, which are $KL_{4,R}^{(1\sim19)} = K_L^{(29\sim47)}$ and $KL_{4,L}^{(1\sim19)} = K_L^{(125\sim128,1\sim15)}$, have been guessed, 38-bit intermediate value $R_{FL,\{1,2\}}|R_{FL,3}^{(1\sim3)}|R_{FL,\{5,6\}}|R_{FL,7}^{(1,2)}|R_{FL,8}^{(8)}$ can be computed, where R_{FL} represents the value after the FL^{-1} function.
   (b) Partially decrypt the ciphertext C to get the intermediate values (L_{22}, R_{22}) and $P^{-1}(L_{22})$. Note that now we can compute $S_{22,\{3\sim8\}}$ as the 48-bit value $K_{22,\{3\sim8\}} = K_L^{(47\sim94)}$ is known.
   (c) Store the values (L_{11}, R_{11}) and (L_{22}, R_{22}) into a hash table Γ indexed by the following 143-bit values.
   - $R_{22,\{1,5\}}$, $R_{22,2} \oplus R_{22,3}$, $R_{22,2} \oplus R_{22,4}$, $R_{22,2} \oplus R_{22,6}$, $R_{22,2} \oplus R_{22,7}$, $R_{22,2} \oplus R_{22,8}$.
   - $S_{22,3} \oplus P^{-1}(L_{22})_3 \oplus P^{-1}(L_{22})_5$, $S_{22,4} \oplus P^{-1}(L_{22})_4 \oplus P^{-1}(L_{22})_5$, $S_{22,6} \oplus P^{-1}(L_{22})_6 \oplus P^{-1}(L_{22})_5$, $S_{22,7} \oplus P^{-1}(L_{22})_7 \oplus P^{-1}(L_{22})_5$, $S_{22,8} \oplus P^{-1}(L_{22})_8$.

– $R^{(8)}_{12,7}$, $R_{FL,1} \oplus (R^{(1\sim7)}_{12,8}|R^{(8)}_{12,3})$, $R_{FL,2} \oplus (R^{(1\sim7)}_{12,8}|R^{(8)}_{12,3})$, $R^{(1\sim3)}_{FL,3} \oplus R^{(1\sim3)}_{12,8}$, $R_{FL,6}$, $R^{(1,2)}_{FL,7}$, $R_{FL,5} \oplus (R^{(1\sim7)}_{12,8}|R^{(8)}_{12,3})$, $R^{(8)}_{FL,8} \oplus R^{(8)}_{12,3}$.

Then each two values lie in the same row of Γ form a pair that satisfies the following conditions.

– The difference $\Delta R_{22} = (0|f|f|f|0|f|f|f)$, where f is a nonzero value.
– The difference $P^{-1}(\Delta L_{22}) = (0|g'_2|g'_3|g'_4|g'_5|g'_6|g'_7|g'_8)$ satisfies $g'_3 \oplus g'_5 = \Delta S_{22,3}$, $g'_4 \oplus g'_5 = \Delta S_{22,4}$, $g'_6 \oplus g'_5 = \Delta S_{22,6}$, $g'_7 \oplus g'_5 = \Delta S_{22,7}$, $g'_8 = \Delta S_{22,8}$.
– Assume the difference ΔR_{12} (equals to ΔL_{11}) is represented as $(b_1|b_2|b_3|b_4|b_5|b_6|b_7|b_8)$, then it satisfies $b^{(8)}_7 = 0$, and the output difference of FL^{-1} function satisfies $\Delta R_{FL,1} = (b^{(1\sim7)}_8|b^{(8)}_3)$, $\Delta R_{FL,2} = (b^{(1\sim7)}_8|b^{(8)}_3)$, $\Delta R^{(1\sim3)}_{FL,3} = b^{(1\sim3)}_8$, $\Delta R_{FL,5} = (b^{(1\sim7)}_8|b^{(8)}_3)$, $\Delta R_{FL,6} = 0$, $\Delta R^{(1,2)}_{FL,7} = 0$ and $\Delta R^{(8)}_{FL,8} = b^{(8)}_3$.

This step performs a 135-bit filtration from 2^{239} pairs, so the expected number of remaining pairs is 2^{104}.

2. Guess 12-bit value $KL^{(20\sim23,25\sim32)}_{4,R}$, compute the output differences $\Delta R^{(4\sim7)}_{FL,3}$, $\Delta R^{(3\sim7)}_{FL,7}$ and $R_{FL,4}$ (from $b^{(8)}_7 = 0$ we conclude $\Delta R^{(8)}_{FL,3} = b^{(8)}_3$). Discard the pairs that do not satisfy $\Delta R^{(4\sim7)}_{FL,3} = b^{(4\sim7)}_8$, $\Delta R^{(3\sim7)}_{FL,7} = 0$ and $\Delta R_{FL,4} = 0$, then the expected number of remaining pairs is 2^{87}. Moveover, from $\Delta R_{FL,4} = 0$ and $b^{(8)}_7 = 0$, we get $\Delta R^{(8)}_{FL,7} = 0$ and $\Delta R^{(1\sim7)}_{FL,8} = b^{(1\sim7)}_8$. Therefore, at the end of this substep, all remaining pairs satisfy the condition $\Delta R_{FL} = (b|b|b|0|b|0|0|b)$, where $b = (b^{(1\sim7)}_8|b^{(8)}_3)$.
3. Guess 7-bit value $K^{(9\sim15)}_{22}$, compute the intermediate value $\Delta S_{22,2}$ ($K^{(16)}_{22}$ ($K^{(46)}_A$) has already been guessed in the step 1), and discard the pairs which do not satisfy $\Delta S_{22,2} = g'_2 \oplus g'_5$. Each pair will be kept with probability 2^{-8}, so the expected number of remaining pairs is 2^{79}.
4. Compute the intermediate value $P^{-1}(\Delta R_{11}) = (c'_1|c'_2|c'_3|c'_4|c'_5|c'_6|c'_7|c'_8)$, then perform the following substeps.
   (a) Guess 17-bit subkeys $K_{12,1}$, $K_{12,7}$ and $K^{(1)}_{12,8}$, calculate the value $\Delta S_{12,\{1,7,8\}}$ (7-bit value $K^{(1\sim7)}_{12,8}$ ($K^{(39\sim45)}_A$) has been guessed in step 3), and discard the pairs which do not satisfy $\Delta S^{(1\sim7)}_{12,1} = c'^{(1\sim7)}_1$ and $\Delta S^{(8)}_{12,7} = c'^{(8)}_7$ according to proposition 7. The expected number of remaining pairs is 2^{71}. Then we compute the value $a_8 = \Delta S_{12,1} \oplus c'_1$, $a^{(1\sim7)}_5 = \Delta S^{(1\sim7)}_{12,7} \oplus c'^{(1\sim7)}_7$ and $a_1 = \Delta S_{12,8} \oplus c'_8 \oplus a_8$.
   (b) For $i = 2$ to 6, guess 8-bit subkey $K_{12,i}$, compute the difference $\Delta S_{12,i}$ and discard the pairs which do not satisfy $\Delta S_{12,j} = c'_j \oplus a_1 \oplus a_5 \oplus a_8$ ($j = 2, 3, 4$), $\Delta S_{12,5} = c'_5 \oplus a_1 \oplus a_8$ and $\Delta S_{12,6} = c'_6 \oplus a_5 \oplus a_8$. Then we expect about 2^{31} pairs remain.
5. Since all of the 128-bit value of K_A have been guessed in step 1, 3 and 4, we compute the values R_{21} and R'_{21} for every remaining pair and keep only the pairs whose $\Delta R^{(1)}_{21,5} = 0$. Then we partially decrypt $R_{21,5}$ and $R'_{21,5}$ to get the value $\Delta S_{21,5}$, keep only the pairs whose $\Delta S_{21,5} = f$, which results in 2^{22} remaining pairs.
6. Guess 17-bit value $KL^{(1\sim9)}_{3,L}$ and $KL_{3,R,1}$, compute $\Delta L_{FL,5}$, $\Delta L^{(8)}_{FL,8}$ and $\Delta L_{FL,1}$. Then discard the pairs whose $(\Delta L^{(1\sim7)}_{FL,5}|\Delta L^{(8)}_{FL,8}) \neq 0$. The expected number of remaining pairs is about 2^{14}.
7. Guess 8-bit value $K_{13,1}$, partially encrypt $L_{FL,1}$ and $L'_{FL,1}$ to get the value $\Delta S_{13,1}$ of every remaining pair. If $\Delta S_{13,1}$ equals to $\Delta R_{FL,\{1,2,3,5,8\}}$, delete this value from the list of all the 2^8 possible values $K_{13,1}$.
8. After analyzing of all remaining pairs, if the list is not empty, announce that the value in the list along with above 223-bit guessed values are the candidates of 231-bit target value of

subkey $K_A|K_R^{(61\sim68)}|K_L^{(1\sim51,53\sim69,93\sim100,110\sim128)}$, then recover the whole master key K_L and K_R by key searching. Otherwise, try the other 223-bit guess.

*C*omplexity. The time complexity is dominated by step 1, which requires about 5 rounds' encryptions to compute the intermediate values for every plaintext and ciphertext pair. Then the time complexity is $2^{120} \times 2^{130} \times 5/14 \approx 2^{248.5}$ 14-round encryptions. The memory requirement is dominated by data collection, which needs 2^{125} bytes to store the known plaintexts and the corresponding ciphertexts. Similarly, the expected time of the attack for the whole key space is about $2^{250.5}$ 14-round encryptions.

Attack on 12-Round Camellia-192 Making use of **2+5 WKID**, we mount the weak-key impossible differential attack on 12-round Camellia-192, which is from round 3 to round 14, where the **2+5 WKID** is applied from round 5 to round 11. The attack procedure is similar to that of 14-round Camellia-256. To summarize, the time complexity of the attack is about $2^{180.1}$ 12-round encryptions. The memory requirement is dominated by step 1, which needs $2^{124.1}$ bytes to store the plaintext-ciphertext pairs. For the attack that works for the whole key space, the data complexity is about $2^{120.1}$ chosen plaintexts, and the time complexity is about 2^{184} 12-round encryptions.

4 8-Round Impossible Differentials of Camellia and Their Applications [2]

In this section, we first present a method to construct a set of differentials, which contains at least one 8-round impossible differential of Camellia with two FL/FL^{-1} layers for any fixed key. Based on this differential set, we propose a new attack strategy to recover the correct key. Finally, we mount impossible differential attacks on reduced-round Camellia-128/192/256 with the whitening and FL/FL^{-1} layers from some intermediate round.

4.1 The Construction of 8-Round Impossible Differentials of Camellia

In this section, we present some 8-round impossible differentials of Camellia with two key-dependent layers by exploiting some properties of the keyed transformation FL/FL^{-1}.

Proposition 7. *If the input difference of FL is $(a|0|0|0|a'|0|0|0)$, where $a^{(1)} = a'^{(8)} = 0$ and*

$$a'^{(i)} = \begin{cases} 0, & kl_L^{(i+1)} = 0; \\ a^{(i+1)}, & kl_L^{(i+1)} = 1; \end{cases} \quad \textit{for } 1 \le i \le 7, \tag{1}$$

then the output of FL is $(a|0|0|0|0|0|0|0)$.

Proof. By Lemma 2, we can obtain

$$\begin{aligned} \Delta Y_R &= ((\Delta X_L \cap kl_L) \lll 1) \oplus \Delta X_R = (((a|0|0|0) \cap kl_L) \lll 1) \oplus (a'|0|0|0) \\ &= ((a^{(2\sim8)}|0 \cap kl_{L,1}) \oplus a')|0|0|0. \end{aligned}$$

According to $a^{(1)} = a'^{(8)} = 0$ and the equation (1), we derive that $\Delta Y_R = 0$. Furthermore, $\Delta Y_L = \Delta X_L \oplus \Delta Y_R \oplus (\Delta Y_R \cap kl_R) = \Delta X_L = a|0|0|0$. Therefore, the output of FL is $(a|0|0|0|0|0|0|0)$. □

By Propositions 7, we construct an 8-round impossible differential of Camellia with two FL/FL^{-1} layers for any fixed subkey.

[2] By Ya Liu, Dawu Gu, Zhiqiang Liu and Wei Li.

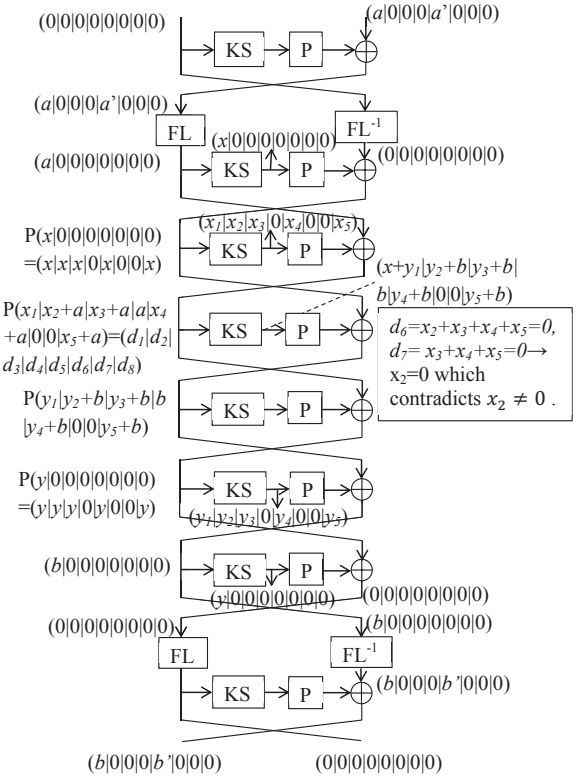

Fig. 3. The Structure of 8-Round Impossible Differentials of Camellia

Proposition 8. *For an 8 rounds of Camellia with two FL/FL^{-1} layers inserted after the first and seventh rounds, the input difference of the first round is $(0|0|0|0|0|0|0|0, a|0|0|0|a'|0|0|0)$ and the output difference of the eighth round is $(b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$ with a and b being nonzero bytes and $a^{(1)} = b^{(1)} = a'^{(8)} = a'^{(8)} = 0$. Four subkeys $kl_i (i = 1, \cdots, 4)$ are used in two FL/FL^{-1} layers. If a' and b' satisfy the following equations:*

$$a'^{(i)} = \begin{cases} 0, & \text{if } kl_1^{(i+1)} = 0; \\ a^{(i+1)}, & \text{if } kl_1^{(i+1)} = 1; \end{cases} \quad b'^{(i)} = \begin{cases} 0, & \text{if } kl_4^{(i+1)} = 0; \\ b^{(i+1)}, & \text{if } kl_4^{(i+1)} = 1; \end{cases} \quad \text{for } 1 \le i \le 7,$$

then

$$(0|0|0|0|0|0|0|0, a|0|0|0|a'|0|0|0) \nrightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$$

is an 8-round impossible differential of Camellia with two FL/FL^{-1} layers (See Fig. 3).

Proof. By proposition 7, we obtain that the input difference of the second round and the output difference of the seventh round are $(a|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$ and $(0|0|0|0|0|0|0|0, b|0|0|0|0|0|0|0)$, respectively. In [23], Wu *et al.* constructed an 8-round impossible differential of Camellia without the FL/FL^{-1} layers: $(0|0|0|0|0|0|0|0, a|0|0|0|0|0|0|0) \nrightarrow_8 (b|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$ where a and b are nonzero bytes. Thus, $(a|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0) \nrightarrow_6 (0|0|0|0|0|0|0|0, b|0|0|0|0|0|0|0)$ is a 6-round impossible differential. In other word, the input difference $(a|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$ cannot result in the output difference $(0|0|0|0|0|0|0|0, b|0|0|0|0|0|0|0)$ after six-round encryption. Therefore,

$$(0|0|0|0|0|0|0|0, a|0|0|0|a'|0|0|0) \nrightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$$

is an 8-round impossible differential of Camellia with two FL/FL^{-1} layers. □

For any fixed subkey, an 8-round impossible differential with two FL/FL^{-1} layers can be constructed. Each possible value of $kl_1^{(2\sim 8)} \mid kl_4^{(2\sim 8)}$ corresponds to the existence of an 8-round impossible differential. For example, if the subkeys $kl_1^{(2\sim 8)} = kl_4^{(2\sim 8)} = 0_{(7)}$, then $(0|0|0|0|0|0|0|0,\ a|0|0|0|0|0|0|0) \nrightarrow_8 (b|0|0|0|0|0|0|0,\ 0|0|0|0|0|0|0|0)$ is an 8-round impossible differential of Camellia with two keyed layers, where $a^{(1)} = b^{(1)} = 0$. If $kl_1^{(2\sim 8)} = kl_4^{(2\sim 8)} = 1_{(7)}$, then $(0|0|0|0|0|0|0|0,\ a|0|0|0|a'|0|0|0) \nrightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$ is an 8-round impossible differential of Camellia with two keyed layers, where a, b, a' and b' are nonzero bytes and satisfy $a^{(1)} = b^{(1)} = a'^{(8)} = b'^{(8)} = 0$, $a'^{(1\sim 7)} = a^{(2\sim 8)}$ and $b'^{(1\sim 7)} = b^{(2\sim 8)}$. All possible values of $kl_1^{(2\sim 8)} \mid kl_4^{(2\sim 8)}$ are from $0_{(14)}$ to $1_{(14)}$. Denote their corresponding impossible differentials by Δ_i for $0 \le i \le 2^{14}-1$. However, it is possible that different values of $kl_1^{(2\sim 8)}$ may result in the same values of a', and different values of $kl_4^{(2\sim 8)}$ may lead to the same values of b'. Therefore, some of 2^{14} differentials are equal to each other. Let A be a set including all differentials $\Delta_i (0 \le i \le 2^{14}-1)$.

$$A = \{\Delta_i \mid 0 \le i \le 2^{14}-1\} \triangleq \{\delta_j \mid 1 \le j \le t\},\ \text{where } t \le 2^{14}.$$

According to Proposition 8, 8-round differentials of A must have the forms:

$$\Delta = (0|0|0|0|0|0|0|0, a|0|0|0|a'|0|0|0) \nrightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$$

with a and b being nonzero bytes and $a^{(1)} = b^{(1)} = a'^{(8)} = b'^{(8)} = 0$. Among them, a' and b' are either zero or nonzero bytes. We divide all differentials of A into three cases in order to simplify our analysis. The first one is $a' = b' = 0$. The second one is $a' = 0$ and $b' \neq 0$, or $a' \neq 0$ and $b' = 0$. The last one is $a' \neq 0$ and $b' \neq 0$.

By proposition 8, we only know the existence of an 8-round impossible differential of Camellia with two FL/FL^{-1} layers for any fixed key, but cannot distinguish it from other differentials of A. Therefore, we require to propose a new attack strategy to recover the correct key based on this differential set.

The Attack Strategy. Select a differential δ_i from A. Based on it, we mount an impossible differential attack on reduced-round Camellia given enough plaintext pairs. More concretely, we select enough plaintexts such that all wrong keys will be removed with high probability if δ_i is an impossible differential.

1. If one subkey remains, we recover the secret key by the key schedule and verify whether it is correct by some plaintext-ciphertext pairs. If success, end this attack. Otherwise, try another differential $\delta_j (j \neq i)$ of A and perform a new impossible differential attack.
2. If no subkey or more than one subkeys is left, select another differential of A to execute a new impossible differential attack.

□

Our attack strategy can really recover the correct key. As a matter of fact, if δ_i is an impossible differential, we make sure the expected number of remaining wrong keys will be almost zero given enough chosen plaintexts. Therefore, we only consider those differentials which result in one subkey remaining. By Proposition 8, we know the differential set A must contain an impossible differential. So we try each differential of A until the correct key is recovered. The worst scenario is that the correct key is retrieved from the last try.

4.2 Impossible Differential Attack on 13-round Camellia-256

Based on three scenarios of differentials in A, we present an impossible differential attack on 13-round Camellia-256 with the FL/FL^{-1} and whitening layers. For each of three cases, we

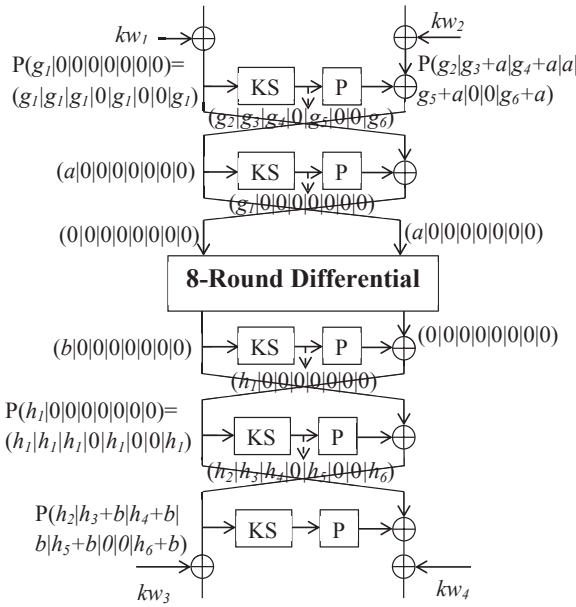

Fig. 4. Impossible Differential Attack on 13-round Camellia-256 for Case 1

put two additional rounds on the top and three additional rounds on the bottom of the 8-round differentials of A. On the basis of this structure, we can attack 13-round Camellia-256 from rounds 4 to 16 or from rounds 10 to 22. Similarly, we put three additional rounds on the plaintext side and two additional rounds on the ciphertext side to attack 13-round Camellia-256 from rounds 3 to 15 or from rounds 9 to 21. Some previously known skills such as building hash tables and the early abort technique [15] are also adopted in order to reduce the time complexity. In this section, we only elaborate the attack procedure of impossible differential cryptanalysis of 13-round Camellia-256 from rounds 4 to 16. Before introducing our attack, we list some notations, i.e.,

$$k_a \triangleq kw_1 \oplus k_4, k_b \triangleq kw_2 \oplus k_5, k_c \triangleq kw_3 \oplus k_{16}, k_d \triangleq kw_4 \oplus k_{15}, k_e \triangleq kw_3 \oplus k_{14}.$$

We use these equivalent subkeys k_a, k_b, k_c, k_d and k_e instead of the round subkeys k_4, k_5, k_{14}, k_{15} and k_{16} so as to remove the whitening layers. This new cipher acts as the original one.

Based on the attack strategy in section 4.1, we mount an impossible differential attack on 13-round Camellia-256 by using differentials of A until the correct key is recovered. In the following, we discuss this attack by three cases.

Case 1 $a' = b' = 0$**:** At this time, the differential $\Delta = (0|0|0|0|0|0|0|0, a|0|0|0|0|0|0|0) \rightarrow_8 (b|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$, where a and b are nonzero bytes and $a^{(1)} = b^{(1)} = 0$ (See Fig. 4).

Data Collection. Select a structure of plaintexts, which contains 2^{55} plaintexts with the following forms:

$$(P(\alpha_1|x_1|x_2|x_3|x_4|x_5|x_6|x_7), P(\alpha_2|\alpha_3|\alpha_4|\alpha_5|\alpha_6|x_8|x_9|\alpha_7)), \tag{2}$$

where $\alpha_5^{(1)}, x_i (1 \leq i \leq 9)$ are fixed and $\alpha_j (1 \leq j \leq 7, i \neq 5), \alpha_5^{(2\sim 8)}$ takes all possible values. Clearly, each structure forms 2^{109} plaintext pairs, the differences of which have the forms: $(P(g_1|0|0|0|0|0|0|0), P(g_2|g_3 \oplus a|g_4 \oplus a|a|g_5 \oplus a|0|0|g_6 \oplus a))$ with a and $g_i (1 \leq i \leq 6)$ being nonzero bytes and $a^{(1)}$=0. We take all possible values of $(\alpha_5^{(1)}, x_4, x_8, x_9)$ and 2^{43} different values of $x_i (1 \leq i \leq 7, i \neq 4)$ to obtain 2^{68} special structures. In total, there are 2^{123} chosen

plaintexts which form 2^{177} plaintext pairs. Encrypt these plaintext pairs to obtain the corresponding ciphertext pairs. If the left halves of their ciphertexts differences have the form: $P(h_1|h_2 \oplus b|h_3 \oplus b|b|h_5 \oplus b|0|0|h_8 \oplus b)$ with $b^{(1)} = 0$, then these pairs will be kept. The expected number of remaining pairs is about 2^{160}.

Key Recovery.

1. Guess $k_{a,1}$. For each remaining pair, check whether the equation $\Delta S_{4,1} = (P^{-1}(\Delta P_R))_1$ holds. If $\Delta S_{4,1} \neq (P^{-1}(\Delta P_R))_1$ for some pair, then this pair will be discarded. Next guess each possible value of $k_{a,l}$ for $l = 2, 3, 5, 8$. Keep only the pairs satisfying $\Delta S_{4,l} = (P^{-1}(\Delta P_R))_l \oplus (P^{-1}(\Delta P_R))_4$. The total probability of this event is about 2^{-40}. Thus the expected number of remaining pairs is about 2^{120}. Finally, guess $k_{a,\{4,6,7\}}$ and compute the inputs of the fifth round.
2. Guess $k_{b,1}$ and test whether $\Delta S_{5,1}$ is equal to $(P^{-1}(\Delta P_L))_1$ for each remaining pair. If $\Delta S_{5,1} \neq (P^{-1}(\Delta P_L))_1$ for one pair, then this pair will be removed. The probability that to happen is about 2^{-8}. Thus about 2^{112} pairs will be kept.
3. Guess $k_{c,l}$ for $2 \leq l \leq 8$. Verify whether $\Delta S_{16,l}$ is equal to $(P^{-1}(\Delta C_R))_l$ for every remaining pair. If $\Delta S_{16,l} \neq (P^{-1}(\Delta C_R))_l$ for some pair, then this pair is discarded. The total probability of this event is 2^{-56}. Therefore, we expect about 2^{56} pairs remain. Next guess $k_{c,1}$ and compute the outputs of the 15-th round for each of the remaining pairs.
4. Guess $k_{d,l}$ for $l = 1, 2, 3, 5, 8$. Verify whether the equations, $\Delta S_{15,1} = (P^{-1}(\Delta C_L))_1$ and $\Delta S_{15,j} = (P^{-1}(\Delta C_L))_j \oplus (P^{-1}(\Delta C_L))_4$ for $j = 2, 3, 5, 8$, hold for every remaining pair. The total probability that to happen is about 2^{-40}. Thus there are about 2^{16} pairs remain. Next guess other bytes of k_d and calculate the outputs of the 14-th round.
5. Guess $k_{e,1}$ and compute the output difference of the S-Boxes in the 14-th round. If $\Delta S_{14,1}$ is equal to $(P^{-1}(\Delta L_{14}))_1$, then we remove this value of $k_{e,1}$ with $(k_a, k_{b,1}, k_c, k_d)$. The probability of this event is about 2^{-8}. After trying all possible values of $(k_a, k_{b,1}, k_c, k_d, k_{e,1})$, if only one joint subkey remains, then Δ is likely to be an impossible differential. At this time, we recover the secret key by the key schedule and verify whether it is correct by some plaintext-ciphertext pairs. If no subkey or more than one subkeys is left, then Δ is possible to exist. At this time, try another differential of A. As a matter of fact, if Δ is an impossible differential, the expected number of the wrong subkeys remaining is about $2^{208} \times (1 - 2^{-8})^{2^{16}} \approx 2^{-161.4}$. We consider that all wrong subkeys are removed and only the correct subkey is left. Therefore, we require to perform the following Step 6 only if one subkey is left.
6. We can recover the secret key from this unique 208-bit subkey $(k_a, k_{b,1}, k_c, k_d, k_{e,1})$. By the key schedule of Camellia-256, we can obtain:

$$k_a = kw_1 \oplus k_4 = (K_L \lll 0)_L \oplus (K_R \lll 15)_R, \tag{3}$$

$$k_b = kw_2 \oplus k_5 = (K_L \lll 0)_R \oplus (K_A \lll 15)_L, \tag{4}$$

$$k_c = kw_3 \oplus k_{16} = (K_B \lll 111)_L \oplus (K_B \lll 60)_R, \tag{5}$$

$$k_d = kw_4 \oplus k_{15} = (K_B \lll 111)_R \oplus (K_B \lll 60)_L, \tag{6}$$

$$k_e = kw_3 \oplus k_{14} = (K_B \lll 111)_L \oplus (K_R \lll 60)_R. \tag{7}$$

We first guess each possible values of K_B. By the equations (5) and (6), we discard some wrong candidates of K_B with the probability 2^{-128}. Therefore, only one value of K_B is left. Then we calculate 8 bits of K_R by the equation (7). Guess the remaining unknown 120 bits of K_R. By property 4 of [17], we can compute the corresponding value for (K_L, K_A). According to the equations (3) and (4), we can discard some wrong candidates of (K_L, K_A). Therefore, the number of the remaining main keys is approximately $2^{120} \times 2^{-72} = 2^{48}$. By about 2^{48} trail encryptions, if some key is correct, stop the attack. Otherwise, try another differential of A.

Case 2 $a' = 0$ **and** $b' \neq 0$**, or** $a' \neq 0$ **and** $b' = 0$**:** We only attack a special scenario, i.e., $a' = 0$, $b' \neq 0$ and $b'^{(1\sim7)} = b^{(2\sim8)}$. The others can be attacked in the similar way. At this moment, the differential is $\Delta' = (0|0|0|0|0|0|0|0, a|0|0|0|0|0|0|0) \rightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$, where a, b and b' are non-zero bytes, $b'^{(1\sim7)} = b^{(2\sim8)}$ and $a^{(1)} = b^{(1)} = b'^{(8)} = 0$.

Data Collection. We apply 2^{68} special structures of Case 1 above. Totally, there are 2^{123} chosen plaintexts which form 2^{177} pairs. At this moment, the form of the ciphertext difference is random.

Key Recovery.

1. Guess $k_{c,l}$ for $2 \leq l \leq 8$ and $l \neq 5$. Verify whether the equation $\Delta S_{16,l} = (P^{-1}(\Delta C_R))_l$ holds for every remaining pair. If $\Delta S_{16,l} \neq (P^{-1}(\Delta C_R))_l$ for some pair, then this pair is discarded. The whole probability of this event is 2^{-48}. Therefore, we expect about 2^{129} pairs remain. Next guess $k_{c,\{1,5\}}$ and compute the outputs of the 15-th round for each of the remaining pairs.
2. We first guess $k_{d,1}$ and check whether the equation $\Delta S_{15,1} = (P^{-1}(\Delta C_L))_1$ holds for each remaining pair. If $\Delta S_{15,1} = (P^{-1}(\Delta C_L))_1$ for one pair, then this pair will be kept. Otherwise, this pair will be discarded. Second, guess $k_{d,8}$ and keep only the pairs satisfying $\Delta S_{15,8}^{(1)} = (P^{-1}(\Delta C_L))_8^{(1)}$. Third, guess $k_{d,\{2\sim7\}}$. Test whether $\Delta S_{15,l} = (P^{-1}(\Delta C_L))_l \oplus (((P^{-1}(\Delta C_L))_8 \oplus \Delta S_{15,8})^{(2\sim8)}|0)$ for $l = 6, 7$ and $\Delta S_{15,l} = (P^{-1}(\Delta C_L))_l \oplus (P^{-1}(\Delta C_L))_8 \oplus \Delta S_{15,8} \oplus (P^{-1}(\Delta C_L))_7 \oplus \Delta S_{15,7}$ for $l = 2, 3, 4, 5$. The total probability of this step is about 2^{-57}. So the expected number of remaining pairs is approximately 2^{72}. Compute the outputs of the 14-th round for each remaining pair.
3. Guess $k_{e,l}$ for $l = 1, 5$. Verify whether the equation $\Delta S_{14,l} = (P^{-1}(\Delta L_{14}))_l$ holds for each remaining pair. If this equation is correct for some pair, then this pair will be kept. The probability of this event is about 2^{-16}. About 2^{56} pairs will be kept.
4. Guess each of possible values k_a as like Case 1 for all remaining pairs. Finally, we expect about 2^{16} pairs remain and calculate the inputs of the fifth round.
5. Guess $k_{b,1}$. This step is similar to Step 5 of Case 1. If only one joint subkey is left, then we consider Δ' is an impossible differential and recover the secret key by the key schedule. Otherwise try another differential of A. In fact, the expected number of the wrong subkeys remaining is approximately $2^{216} \times (1 - 2^{-8})^{2^{16}} \approx 2^{-153.4}$ if Δ' is an impossible differential.
6. This step is similar to Step 6 of Case 1. The difference is that the equation (7) can give 16 bits of K_R. Therefore, we only require to guess 112 bits of K_R. About 2^{40} keys will be left. By about 2^{40} trail encryptions, if some key is correct, stop the attack. Otherwise, try another differential of A.

Case 3 $a' \neq 0$ **and** $b' \neq 0$**:** We only discuss an example, i.e., $a'^{(1\sim7)} = a^{(2\sim8)}$ and $b'^{(1\sim7)} = b^{(2\sim8)}$. At this moment, the differential is $\Delta'' = (0|0|0|0|0|0|0|0, a|0|0|0|a'|0|0|0) \rightarrow_8 (b|0|0|0|b'|0|0|0, 0|0|0|0|0|0|0|0)$, where a, b, a' and b' are nonzero bytes and $a^{(1)} = b^{(1)} = a'^{(8)} = b'^{(8)} = 0$.

Data Collection. Continue to adopt 2^{123} chosen plaintexts in Case 1. Because each structure of Case 1 takes all possible values of $\alpha_5^{(1)}, x_4, x_8$ and x_9, 2^{123} chosen plaintexts of Case 1 are equivalent to 2^{43} structures, each of which contains 2^{80} plaintexts with the forms: $(P(\beta_1|y_1|y_2|y_3|\beta_2|y_4|y_5|y_6), \beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8|\beta_9|\beta_{10})$, where $y_i (1 \leq i \leq 6)$ are fixed and $\beta_j (1 \leq j \leq 10)$ takes all possible values. It is obvious that one structure generates 2^{159} pairs. Totally, there are approximately 2^{202} plaintext pairs satisfying the input differences.

Key Recovery.

1. Guess each byte of $k_c, k_d, k_{e,\{1,5\}}$. This step is similar to Case 2 above. After guessing these subkeys, we expect about 2^{81} pairs remain.
2. Guess $k_{a,1}$, $k_{a,8}$, $k_{a,\{6,7\}}$ and $k_{a,\{2\sim5\}}$ in turn. After our test, about 2^{24} pairs will be kept. Compute the inputs of the fifth round for every remaining pair.
3. Guess $k_{b,5}$ and kept these pairs satisfying $\Delta S_{5,5} = (P^{-1}(\Delta P_L))_5$. Finally, there are about 2^{16} pairs remain. Next guess $k_{b,1}$ and test whether $\Delta S_{5,1}$ is equal to $(P^{-1}(\Delta P_L))_1$ for the remaining pairs. If $\Delta S_{5,1} = (P^{-1}(\Delta P_L))_1$ for some pair, then this value $k_{b,1}$ with the guessed value $(k_a, k_{b,5}, k_c, k_d, k_{e,\{1,5\}})$ are removed. After guessing all possible values $(k_a, k_{b,\{1,5\}}, k_c, k_d, k_{e,\{1,5\}})$, if only one joint subkey is left, then we consider Δ'' is an impossible differential. At this moment, we execute the following step. Otherwise try another differential of A. As a matter of fact, the expected number of the wrong subkeys remaining is approximately $2^{224} \times (1-2^{-8})^{2^{16}} \approx 2^{-145.4}$ if Δ'' is an impossible differential.
4. Similarly, we require to recover the secret key only if one subkey is left. Compared with Step 6 of Case 2 above, the difference is the equation (5) can give 16 bits of K_L. Therefore, the number of the remaining main keys is approximately 2^{32}. By about 2^{32} trail encryptions, if some key is correct, stop the attack. Otherwise, try another differential of A.

The Algorithm of Impossible Differential Attack on 13-Round Camellia-256:
For each differential δ_i of A, do
If δ_i belongs to Case 1, we perform the attacking procedure of Case 1.
If δ_i belongs to Case 2, we perform the similar attacking procedure of Case 2.
If δ_i belongs to Case 3, we perform the similar attacking procedure of Case 3.
If the correct key is recovered, end this algorithm. Otherwise, try another differential of A.
□

Table 2. Time Complexity of Cases 1

Step	Time Complexity (1-round encryptions)
2	$2^{160} \times 2 \times 2^8 \times 5 \times \frac{1}{8} + 2^{120} \times 2 \times 2^{64} \times \frac{3}{8} \approx 2^{183.6}$
3	$2^{120} \times 2 \times 2^{64} \times 2^8 \times \frac{1}{8} = 2^{190}$
4	$2^{112} \times 2 \times 2^{72} \times 2^8 \times 7 \times \frac{1}{8} + 2^{56} \times 2 \times 2^{136} \times \frac{1}{8} = 2^{193}$
5	$2^{56} \times 2 \times 2^{136} \times 2^8 \times 5 \times \frac{1}{8} + 2^{16} \times 2 \times 2^{200} \times \frac{3}{8} \approx 2^{215.6}$
6	$2^{208} \times 2 \times (1 + (1-2^{-8}) + \cdots + (1-2^{-8})^{2^{16}}) \times \frac{1}{8} \approx 2^{214}$
7	$2^{120} \times 6 + 2^{48} \times 13 \approx 2^{122.4}$

Analysis of Complexity In table 2, we list the time complexity of each step in Case 1. We find that the total time complexity is about 2^{216} 1-round encryptions. Similarly, we can compute the time complexities of Case 2 and Case 3. For Case 2, the total time complexity is approximately 2^{224} 1-round encryptions. For Case 3, the total time complexity is approximately $2^{240.8}$ 1-round encryptions. Thus the total time complexity is at most $2^{14} \times 2^{240.8} \times \frac{1}{13} \approx 2^{251.1}$ 13-round encryptions. Furthermore, the total data and memory complexities are 2^{123} chosen plaintexts and 2^{208} bytes, respectively.

4.3 Impossible Differential Attack on 12-round Camellia-192

In this part, an impossible differential attack on 12-round Camellia-192 is executed. We set two additional rounds on the top and on the bottom of our 8-round differentials, respectively. By applying it, we can attack 12-round Camellia-192 from rounds 4 to 15 with the 8-round

impossible differentials inserted rounds 6 to 13. Similarly, we can also attack 12-round Camellia-192 from rounds i to $i+11$ where $i = 3, 5, 9, 10$. Some equivalent subkeys k_a and k_b are defined as before. In addition, let

$$k'_d = kw_3 \oplus k_{15} = (K_B \lll 111)_L \oplus (K_B \lll 60)_L, \tag{8}$$

$$k'_e = kw_4 \oplus k_{14} = (K_B \lll 111)_R \oplus (K_R \lll 60)_R. \tag{9}$$

Case 1 $a' = b' = 0$**:** The differential is Δ.

Data Collection. We select the same plaintexts of Case 1 mentioned in section 4.2. I.e., 2^{123} chosen plaintexts can form 2^{177} pairs. Encrypt these plaintext pairs. Keep only the pairs which have the form of ciphertext differences: $(P(h_1|0|0|0|0|0|0|0), P(h_2|h_3 \oplus b|h_4 \oplus b|b|h_5 \oplus b|0|0|h_6 \oplus b))$, where b and $h_i (1 \le i \le 6)$ are nonzero bytes and $b^{(1)} = 0$. The expected number of remaining pairs is 2^{104}.

Key Recovery. Guess all possible values $(k_a, k_{b,1}, k'_d, k'_{e,1})$ and discard those subkeys which acquire the input and output differences of Δ. This step is similar to section 4.2. If Δ is an impossible differential, about $2^{144} \times (1-2^{-8})^{2^{16}} \approx 2^{-225.4}$ wrong subkeys are expected to remain. Therefore, we will recover the secret key by the key schedule of Camellia-192 only if one subkey is left. Otherwise, try another differential of A. By the key schedule of Camellia-192, we can recover the secret key from the 144-bit subkey $(k_a, k_{b,1}, k'_d, k'_{e,1})$. We first guess all possible values of K_B. By the equation (8), we can get rid of some wrong candidates of K_B with the probability 2^{-64}. So about 2^{64} values of K_B remain. Then we can compute 8 bits of K_R by the equation (9). Guessing the remaining unknown 56 bits of K_R, we calculate (K_L, K_A) and remove some wrong values of (K_L, K_A, K_R) by the equations (3) and (4). The expected number of remaining secret keys is approximately $2^{64} \times 2^{56} \times 2^{-64} \times 2^{-8} = 2^{48}$. By about 2^{48} trail encryptions, if the correct key is retrieved, end the attack. Otherwise, try another differential of A.

Case 2 $a' = 0, b' \neq 0$ **or** $a' \neq 0, b' = 0$**:** For simplicity, we consider a special differential Δ'.

We still select 2^{123} plaintexts above. In total, there are 2^{68} special structures, each of which contains 2^{55} plaintexts. Encrypt these plaintext pairs. If the left halves of their ciphertexts differences have the forms: $P(h|0|0|0|h'|0|0|0)$ with h and h' being nonzero bytes, then these pairs will be kept. Consequently, the expected number of remaining pairs is about 2^{129}. Similarly, we can remove some subkeys $(k_a, k_{b,1}, k'_d, k'_{e,\{1,5\}})$ which obtain the input and output differences of Δ' for some pair. If only one subkey is left, we recover the secret key by the key schedule. Otherwise, try another differential of A. In fact, if Δ' is an impossible differential, about $2^{-217.4} (\approx 2^{152} \times (1-2^{-8})^{2^{16}})$ wrong subkeys will be left.

Case 3 $a' \neq 0, b' \neq 0$**:** A special differential Δ'' will be considered.

The similar attacking procedure can be performed as before. We select 2^{43} structure, each of which contains 2^{80} plaintexts. Totally, they can form 2^{202} pairs. After filtering some pairs by the ciphertext differences, about 2^{154} pairs are expected to remain. The following steps can be preformed in the similar way.

By the careful analysis, we found that the time complexity of Case 3 is maximal. Therefore, the total time complexity is at most $2^{14} \times 2^{173.2} \approx 2^{187.2}$ 12-round encryptions. The data and memory complexities are 2^{123} chosen plaintexts and 2^{160} bytes, respectively.

4.4 Impossible Differential Attack on 11-round Camellia-128

For Camellia-128, we put two additional rounds on the top and one additional round on the bottom of 8-round differentials. Based on it, we attack 11-round Camellia-128 from rounds 4 to 14 or rounds 10 to 20. Similarly, we can also attack Camellia-128 from rounds 5 to 15 and rounds 11 to 21 by setting one additional round on the top and two rounds on the bottom. Here we present an attack on 11-round Camellia-128 from rounds 4 to 14 briefly. Similarly, we divide all possible differentials into three different cases as before. For Case 1, we take 2^{67} special structures (2). Totally, the data complexity is 2^{122} chosen plaintexts which form 2^{176} pairs. Their input differences have the form $(P(g_1|0|0|0|0|0|0|0), P(g_2|g_3 \oplus a|g_4 \oplus a|a|g_5 \oplus a|0|0|g_6 \oplus a))$, where a and $g_i (1 \leq i \leq 6)$ are nonzero bytes and $a^{(1)} = 0$. Encrypt these pairs to acquire the corresponding ciphertext pairs. Then we discard some pairs whose ciphertext differences don't satisfy these form: $(b|0|0|0|0|0|0|0, P(h|0|0|0|0|0|0|0))$ with b and h being non-zero bytes and $b^{(1)} = 0$. The number of remaining pairs after this test is 2^{63}. Guess $k_{e,1}$ and verify whether the equation $\Delta S_{14,1} = (P^{-1}(\Delta C_R))_1$ holds. It is obvious that there are about 2^{55} pairs remain. Next guess $(k_a, k_{b,1})$, operate the similar step as section 4.2. If only one subkey is left, we retrieve the secret key by the key schedule. Otherwise, try anther differential of A. As a matter of fact, if Δ is an impossible differential, the expected number of remaining pairs is $2^{80} \times (1 - 2^{-8})^{15} \approx 2^{-104.7}$. For other two cases, we can accomplish the similar attack procedure.

We find that the dominant time complexity of all steps in three cases is the data collection. Therefore, the total data, time and memory complexities are 2^{122} chosen plaintext, 2^{122} 11-round encryptions and 2^{102} bytes, respectively.

5 Conclusion

In this paper, we have presented new insight on impossible differential cryptanalysis of reduced-round Camellia with the FL/FL^{-1} and whitening layers. First, we propose impossible differential attacks on reduced-round Camellia for 75% of the keys, which are then extended to attacks that work for the whole key space. Specifically, we attack 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256 which start from the first round and include the whitening layers. Meanwhile, we also attack 12-round Camellia-192 and 14-round Camellia-256 with two FL/FL^{-1} layers. Second, we construct a set of differentials including at least one 8-round impossible differential of Camellia with two layers FL/FL^{-1}. These impossible differentials have the same length as the best known impossible differential of Camellia without FL/FL^{-1} layers. Therefore, our result shows that the keyed functions cannot thwart impossible differential attack effectively. Based on it, we propose a new strategy to derive an effective attack on reduced-round Camellia which do not start the first round but include the whitening and FL/FL^{-1} layers. More concretely, we mount impossible differential attacks on 11-round Camellia-128, 12-round Camellia-192 and 13-round Camellia-256.

References

1. Aoki, K., Ichikawa, T., Kanda, M., Matsui, M., Moriai, S., Nakajima, J., Tokita, T.: Camellia: A 128-bit block cipher suitable for multiple platforms - design and analysis. In: Stinson, D.R., Tavares, S.E. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 2012, pp. 39–56. Springer (2000)
2. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In: EUROCRYPT. pp. 12–23 (1999)
3. Chen, J., Jia, K., Yu, H., Wang, X.: New impossible differential attacks of reduced-round Camellia-192 and Camellia-256. In: Parampalli, U., Hawkes, P. (eds.) ACISP. Lecture Notes in Computer Science, vol. 6812, pp. 16–33. Springer (2011)
4. CRYPTREC-Cryptography Research and Evaluation Committees: report. Archive (2002), http://www.ipa.go.jp/security/enc/CRYPTREC/index-e.html

5. Hatano, Y., Sekine, H., Kaneko, T.: Higher order differential attack of Camellia (II). In: Nyberg, K., Heys, H.M. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 2595, pp. 129–146. Springer (2002)
6. International Standardization of Organization (ISO): International standard - ISO/IEC 18033-3. Tech. rep., Information technology - Security techniques - Encryption algrithm - Part 3: Block Ciphers (July 2005)
7. Knudsen, L.R.: DEAL - a 128-bit block cipher. Tech. rep., Department of Informatics, University of Bergen, Norway (1998), technical report
8. Kühn, U.: Improved cryptanalysis of MISTY1. In: Daemen, J., Rijmen, V. (eds.) FSE. Lecture Notes in Computer Science, vol. 2365, pp. 61–75. Springer (2002)
9. Lee, S., Hong, S., Lee, S., Lim, J., Yoon, S.: Truncated differential cryptanalysis of Camellia. In: Kim, K. (ed.) ICISC. Lecture Notes in Computer Science, vol. 2288, pp. 32–38. Springer (2001)
10. Lei, D., Li, C., Feng, K.: New observation on Camellia. In: Preneel, B., Tavares, S.E. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 3897, pp. 51–64. Springer (2005)
11. Lei, D., Li, C., Feng, K.: Square like attack on Camellia. In: Qing, S., Imai, H., Wang, G. (eds.) ICICS. Lecture Notes in Computer Science, vol. 4861, pp. 269–283. Springer (2007)
12. Li, L., Chen, J., Jia, K.: New impossible differential cryptanalysis of reduced-round Camellia. In: Lin, D., Tsudik, G., Wang, X. (eds.) CANS. Lecture Notes in Computer Science, vol. 7092, pp. 26–39. Springer (2011)
13. Li, L., Chen, J., Wang, X.: Security of reduced-round Camellia against impossible differential attack. IACR Cryptology ePrint Archive 2011, 524 (2011)
14. Liu, Y., Gu, D., Liu, Z., Li, W., Man, Y.: Improved results on impossible differential cryptanalysis of reduced-round Camellia-192/256. IACR Cryptology ePrint Archive 2011, 671 (2011)
15. Lu, J., Kim, J., Keller, N., Dunkelman, O.: Improving the efficiency of impossible differential cryptanalysis of reduced Camellia and MISTY1. In: Malkin, T. (ed.) CT-RSA. Lecture Notes in Computer Science, vol. 4964, pp. 370–386. Springer (2008)
16. Lu, J., Wei, Y., Kim, J., Fouque, P.A.: Cryptanalysis of reduced versions of the Camellia block cipher. In: Preproceeding of SAC (2011)
17. Lu, J., Wei, Y., Kim, J., Pasalic, E.: The higher-order meet-in-the-middle attack and its application to the Camellia block cipher. In: Presented in part at the First Asian Workshop on Symmetric Key Cryptography (ASK 2011) (Augst 2011), a full version is available at https://sites.google.com/site/jiqiang/
18. Mala, H., Shakiba, M., Dakhilalian, M., Bagherikaram, G.: New results on impossible differential cryptanalysis of reduced-round Camellia-128. In: Jr., M.J.J., Rijmen, V., Safavi-Naini, R. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 5867, pp. 281–294. Springer (2009)
19. NESSIE: New european schemes for signatures, integrity, and encryption, final report of eurpean project IST-1999-12324. Archive (1999), http://www.cosic.esat.kuleuven.be/nessie/Bookv015.pdf
20. Shirai, T.: Differential, linear, boomerange and rectangle cryptanalysis of reduced-round Camellia. Proceedings of 3rd NESSIE Workshop, Munich, Germany (November 6-7 2002)
21. Sugita, M., Kobara, K., Imai, H.: Security of reduced version of the block cipher Camellia against truncated and impossible differential cryptanalysis. In: Boyd, C. (ed.) ASIACRYPT. Lecture Notes in Computer Science, vol. 2248, pp. 193–207. Springer (2001)
22. Wu, W., Feng, D., Chen, H.: Collision attack and pseudorandomness of reduced-round Camellia. In: Handschuh, H., Hasan, M.A. (eds.) Selected Areas in Cryptography. Lecture Notes in Computer Science, vol. 3357, pp. 252–266. Springer (2004)
23. Wu, W., Zhang, W., Feng, D.: Impossible differential cryptanalysis of reduced-round ARIA and Camellia. J. Comput. Sci. Technol. 22(3), 449–456 (2007)