Meet-in-the-Middle Attack on Reduced Versions of the Camellia Block Cipher[⋆]

Jiqiang Lu[1,⋆⋆], Yongzhuang Wei[2,3], Enes Pasalic[4], and Pierre-Alain Fouque[5]

[1] Institute for Infocomm Research,
Agency for Science, Technology and Research,
1 Fusionopolis Way, Singapore 138632
lvjiqiang@hotmail.com, jlu@i2r.a-star.edu.sg
[2] Guilin University of Electronic Technology,
Guilin City, Guangxi Province 541004, China
[3] State Key Lab of Information Security, Institute of Software,
Chinese Academy of Sciences, Beijing 100190, China
walker_wei@msn.com
[4] University of Primorska FAMNIT, Koper, Slovenia
enes.pasalic6@gmail.com
[5] Département d'Informatique, École Normale Supérieure,
45 Rue d'Ulm, Paris 75005, France
Pierre-Alain.Fouque@ens.fr

Abstract. The Camellia block cipher has a 128-bit block length and a user key of 128, 192 or 256 bits long, which employs a total of 18 rounds for a 128-bit key and 24 rounds for a 192 or 256-bit key. It is a Japanese CRYPTREC-recommended e-government cipher, a European NESSIE selected cipher, and an ISO international standard. In this paper, we describe a few 5 and 6-round properties of Camellia and finally use them to give meet-in-the-middle attacks on 10-round Camellia under 128 key bits, 11-round Camellia under 192 key bits and 12-round Camellia under 256 key bits, all of which include the FL/FL^{-1} functions but do not include whitening operations.

Key words: Block cipher, Camellia, Meet-in-the-middle attack.

1 Introduction

Camellia [1] is a 128-bit block cipher with a user key length of 128, 192 or 256 bits, which employs a total of 18 rounds if a 128-bit key is used and a total of 24 rounds if a 192/256-bit key is used. It has a Feistel structure with key-dependent logical functions FL/FL^{-1} inserted after every six rounds, plus four

[⋆] The work was supported by the French ANR project SAPHIR II (No. ANR-08-VERS-014), the Natural Science Foundation of China (No. 61100185), Guangxi Natural Science Foundation (No. 2011GXNSFB018071), and the Foundation of Guangxi Key Lab of Wireless Wideband Communication and Signal Processing (No. 11101).

[⋆⋆] The author was with École Normale Supérieure (France) when this work was done.

additional whitening operations at both ends. Camellia became a CRYPTREC e-government recommended cipher [7] in 2002, a NESSIE selected block cipher [26] in 2003, and was adopted as an ISO international standard [16] in 2005. In this work, we consider the version of Camellia that has the $\mathrm{FL}/\mathrm{FL}^{-1}$ functions, and for simplicity, we denote by Camellia-128/192/256 the three versions of Camellia that use 128, 192 and 256 key bits, respectively.

The security of Camellia has been analysed against a variety of cryptanalytic techniques, including differential cryptanalysis [5], truncated differential cryptanalysis [17], higher-order differential cryptanalysis [17, 20], linear cryptanalysis [25], integral cryptanalysis [8, 15, 19], boomerang attack [30], rectangle attack [4], collision attack and impossible differential cryptanalysis [3, 18]; and many cryptanalytic results on Camellia have been published [2, 6, 11–13, 21–24, 27–29, 31, 32], of which impossible differential cryptanalysis is the most efficient technique in terms of the numbers of attacked rounds, that broke 11-round Camellia-128, 12-round Camellia-192 and 14-round Camellia-256 [2, 22], presented most recently at FSE 2012 and ISPEC 2012.[1]

The meet-in-the-middle (MitM) attack was introduced in 1977 by Diffie and Hellman [10]. It usually treats a block cipher $\mathbf{E} : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n$ as a cascade of two sub-ciphers $\mathbf{E} = \mathbf{E}^a \circ \mathbf{E}^b$. Given a guess for the subkeys used in $\mathbf{E}^a$ and $\mathbf{E}^b$, if a plaintext produces just after $\mathbf{E}^a$ the same value as the corresponding ciphertext produces just before $\mathbf{E}^b$, then this guess for the subkeys is likely to be correct; otherwise, this guess must be incorrect. Thus, we can find the correct subkey, given a sufficient number of matching plaintext-ciphertext pairs in a known-plaintext attack scenario. In a chosen-plaintext attack scenario, things may get better, and as in [9], by choosing a set of plaintexts with a particular property we may be able to express the concerned value-in-the-middle as a function of plaintext and a smaller number of unknown constants than the number of unknown constants (of the same length) from the subkey involved.

In 2011 Lu et al. [24] proposed an extension of the MitM attack, known as the higher-order MitM (HO-MitM) attack, which is based on using multiple plaintexts to cancel some key-dependent component(s) or parameter(s) when constructing a basic unit of "value-in-the-middle". The HO-MitM attack technique can lead to some better cryptanalytic results than the MitM attack technique in certain circumstances. In particular, Lu et al. found some 5 and 6-round HO-MitM properties of Camellia that were used to break 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256, but the corresponding 5 and 6-round MitM properties can enable us to break only 12-round Camellia-256.

In this paper, we analyse the security of Camellia (with the $\mathrm{FL}/\mathrm{FL}^{-1}$ functions) against the MitM attack in detail, following the work in [24]. In all those 5 and 6-round (higher-order) MitM properties of Camellia due to Lu et al. [24], the

[1] When our work was done in 2011, the best previously published cryptanalytic results on Camellia with $\mathrm{FL}/\mathrm{FL}^{-1}$ functions were square attack on 9-round Camellia-128 [11], impossible differential attack on 10-round Camellia-192 [6], and higher-order differential and impossible differential attacks on 11-round Camellia-256 [6, 13]. We incorporate the newly emerging main results in this editorially revised version.

Table 1. Main cryptanalytic results on Camellia with FL/FL^{-1} functions

Cipher	Attack Type	Rounds	Data	Memory	Time	Source
Camellia-128	Square	9	2^{48}CP	2^{53}Bytes	2^{122}Enc.	[11]
	Impossible differential	10	2^{118}CP	2^{93}Bytes	2^{118}Enc.	[23]
		11	$2^{120.5}$CP	$2^{115.5}$Bytes	$2^{123.8}$Enc.	[2]§
		11†	2^{122}CP	2^{102}Bytes	2^{122}Enc.	[22]§
	HO-MitM	10	2^{93}CP	2^{109}Bytes	$2^{118.6}$Enc.	[24]
	MitM	10	2^{64}CP	2^{105}Bytes	$2^{121.5}$Enc.	Sect. 4
Camellia-192	Boomerang	9	$2^{123.9}$ACPC	2^{72}Bytes	$2^{169.9}$MA	[27]
	Impossible differential	10	2^{121}CP	$2^{155.2}$Bytes	2^{144}Enc.	[6]
		10†	2^{121}CP	$2^{155.2}$Bytes	$2^{175.3}$Enc.	[6]
		11	2^{118}CP	2^{141}Bytes	$2^{163.1}$Enc.	[23]
		12	$2^{120.6}$CP	$2^{171.6}$Bytes	$2^{171.4}$Enc.	[2]§
		12†	2^{123}CP	2^{160}Bytes	$2^{187.2}$Enc.	[22]§
	HO-MitM	11	2^{78}CP	2^{174}Bytes	$2^{187.4}$Enc.	[24]
		11	2^{94}CP	2^{174}Bytes	$2^{180.2}$Enc.	[24]
	MitM	11	2^{80}CP	2^{105}Bytes	$2^{189.4}$Enc.	Sect. 5
		11	$2^{59.4}$CP	$2^{167.6}$Bytes	$2^{169.8}$Enc.	Sect. 5
Camellia-256	Rectangle	10	$2^{126.5}$CP	$2^{126.5}$Bytes	$2^{240.9}$MA	[27]
	Integral	10	$2^{60.5}$CP	2^{63}Bytes	$2^{254.3}$Enc.	[23, 32]
	Higher-order differential	11‡	2^{93}CP	2^{98}Bytes	$2^{255.6}$Enc.	[13, 23]
	Impossible differential	11†	2^{121}CP	2^{166}Bytes	$2^{206.8}$Enc.	[6]
		13†	2^{123}CP	2^{208}Bytes	$2^{251.1}$Enc.	[22]§
		14	$2^{121.2}$CP	$2^{180.2}$Bytes	$2^{238.3}$Enc.	[2]§
		14	2^{120}CC	2^{125}Bytes	$2^{250.5}$Enc.	[22]§
	HO-MitM	12	2^{94}CP	2^{174}Bytes	$2^{237.3}$Enc.	[24]
	MitM	12	2^{64}CP	2^{185}Bytes	$2^{219.9}$Enc.	Sect. 6

§: Newly emerging results; †: Include whitening operations; ‡: Can include whitening operations by making use of an equivalent structure of Camellia.

basic unit of value-in-the-middle is one byte long. Nevertheless, we observe that if we consider only a smaller number of bits of the concerned byte, instead of the whole 8 bits, a few 5 and 6-round MitM properties with a smaller number of unknown 1-bit constant parameters can be obtained. This is due to the fact that an output bit of the FL^{-1} function only relies on a small fraction of the bits of the subkey used in the FL^{-1} function (as well as a few input bits to FL^{-1}), thus reducing the number of unknown 1-bit constant parameters when we consider a fraction of the bits of the concerned byte. As a consequence, the 5 and 6-round MitM properties can be used to conduct MitM attacks on 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256, (all of which include the FL/FL^{-1} functions). Table 1 summarises previous, our and the newly emerging main cryptanalytic results on Camellia (with FL/FL^{-1} functions), where CP, ACPC and CC refer respectively to the numbers of chosen plaintexts, adaptively chosen plaintexts and ciphertexts, and chosen ciphertexts, Enc. refers to the required number of encryption operations of the relevant reduced version of Camellia, and MA refers to the required numbers of memory accesses.

The remainder of the paper is organised as follows. In the next section, we describe the notation and the Camellia block cipher. We give the 5 and 6-round MitM properties in Section 3, and present our cryptanalytic results on Camellia-128/192/256 in Sections 4–6, respectively. Concluding remarks are given in Section 7.

2 Preliminaries

In this section we give the notation used throughout this paper, and then briefly describe the Camellia block cipher.

2.1 Notation

The bits of a value are numbered from left to right, starting with 1. We use the following notation throughout this paper.

$\oplus$	bitwise logical exclusive OR (XOR) of two bit strings of the same length
$\cap$	bitwise logical AND of two bit strings of the same length
$\cup$	bitwise logical OR of two bit strings of the same length
$\lll$	left rotation of a bit string
$\|\|$	bit string concatenation
$\circ$	functional composition. When composing functions X and Y, X $\circ$ Y denotes the function obtained by first applying X and then Y
$\overline{X}$	bitwise logical complement of a bit string X
$X[i_1,\cdots,i_j]$	the j-bit string of bits $(i_1,\cdots,i_j)$ of a bit string X

2.2 The Camellia Block Cipher

Camellia [1] has a Feistel structure, a 128-bit block length, and a user key length of 128, 192 or 256 bits. It uses the following five functions:

- $\mathbf{S}:\{0,1\}^{64}\rightarrow\{0,1\}^{64}$ is a non-linear substitution constructed by applying eight 8×8-bit S-boxes $\mathrm{S}_1,\mathrm{S}_2,\mathrm{S}_3,\mathrm{S}_4,\mathrm{S}_5,\mathrm{S}_6,\mathrm{S}_7$ and S_8 in parallel to the input, where S_1 and S_8 are identical, S_2 and S_5 are identical, S_3 and S_6 are identical, and S_4 and S_7 are identical.
- $\mathbf{P}:GF(2^8)^8\rightarrow GF(2^8)^8$ is a linear permutation which is equivalent to pre-multiplication by a 8×8 byte matrix P; the matrix P and its reverse P^{-1} are as follows.

$$\mathrm{P}=\begin{pmatrix}1&0&1&1&0&1&1&1\\1&1&0&1&1&0&1&1\\1&1&1&0&1&1&0&1\\0&1&1&1&1&1&1&0\\1&1&0&0&0&1&1&1\\0&1&1&0&1&0&1&1\\0&0&1&1&1&1&0&1\\1&0&0&1&1&1&1&0\end{pmatrix},\ \mathrm{P}^{-1}=\begin{pmatrix}0&1&1&1&0&1&1&1\\1&0&1&1&1&0&1&1\\1&1&0&1&1&1&0&1\\1&1&1&0&1&1&1&0\\1&1&0&0&1&0&1&1\\0&1&1&0&1&1&0&1\\0&0&1&1&1&1&1&0\\1&0&0&1&0&1&1&1\end{pmatrix}.$$

- $\mathbf{F}: \{0,1\}^{64} \times \{0,1\}^{64} \rightarrow \{0,1\}^{64}$ is a Feistel function. If X and Y are 64-bit blocks, $\mathbf{F}(X,Y) = \mathbf{P}(\mathbf{S}(X \oplus Y))$.
- $\mathbf{FL}/\mathbf{FL}^{-1}: \{0,1\}^{64} \times \{0,1\}^{64} \rightarrow \{0,1\}^{64}$ are key-dependent linear functions. If $X = (X_L||X_R)$ and $Y = (Y_L||Y_R)$ are 64-bit blocks, then $\mathbf{FL}(X,Y) = ((((X_L \cap Y_L) \lll 1 \oplus X_R) \cup Y_R) \oplus X_L)||((X_L \cap Y_L) \lll 1 \oplus X_R)$, and $\mathbf{FL}^{-1}(X,Y) = (X_L \oplus (X_R \cup Y_R))||(((X_L \oplus (X_R \cup Y_R)) \cap Y_L) \lll 1 \oplus X_R)$.

Camellia uses a total of four 64-bit whitening subkeys KW_j, $2\lfloor \frac{N_r-6}{6} \rfloor$ 64-bit subkeys KI_l for the $\mathbf{FL}$ and $\mathbf{FL}^{-1}$ functions, and N_r 64-bit round subkeys K_i, $(1 \leqslant j \leqslant 4, 1 \leqslant l \leqslant 2\lfloor \frac{N_r-6}{6} \rfloor, 1 \leqslant i \leqslant N_r)$, all derived from a N_k-bit key K, where N_r is 18 for Camellia-128, and 24 for Camellia-192/256, N_k is 128 for Camellia-128, 192 for Camellia-192, and 256 for Camellia-256. The key schedule is as follows. First, generate two 128-bit strings K_L and K_R from K in the following way: For Camellia-128, K_L is the 128-bit key K, and K_R is zero; for Camellia-192, K_L is the left 128 bits of K, and K_R is the concatenation of the right 64 bits of K and the complement of the right 64 bits of K; and for Camellia-256, K_L is the left 128 bits of K, and K_R is the right 128 bits of K. Second, depending on the key size, generate one or two 128-bit strings K_A and K_B from (K_L, K_R) by a non-linear transformation (see [1] for its detail). Finally, the subkeys are as follows.[2]

- For Camellia-128: $K_2 = (K_A \lll 0)[65 \sim 128], K_3 = (K_L \lll 15)[1 \sim 64], K_9 = (K_A \lll 45)[1 \sim 64], K_{10} = (K_L \lll 60)[65 \sim 128], K_{11} = (K_A \lll 60)[1 \sim 64], \cdots$.
- For Camellia-192/256: $K_7 = (K_B \lll 30)[1 \sim 64], K_8 = (K_B \lll 30)[65 \sim 128], K_{13} = (K_R \lll 60)[1 \sim 64], K_{14} = (K_R \lll 60)[65 \sim 128], K_{15} = (K_B \lll 60)[1 \sim 64], K_{16} = (K_B \lll 60)[65 \sim 128], K_{17} = (K_L \lll 77)[1 \sim 64], K_{18} = (K_L \lll 77)[65 \sim 128], K_{21} = (K_A \lll 94)[1 \sim 64], K_{22} = (K_A \lll 94)[65 \sim 128], K_{23} = (K_L \lll 111)[1 \sim 64], \cdots$.

Below is the encryption procedure Camellia, where P is a 128-bit plaintext, represented as 16 bytes, and L_0, R_0, L_i, R_i, $\widehat{L}_i$ and $\widehat{R}_i$ are 64-bit variables.

```
1. $L_0||R_0 = P \oplus (KW_1||KW_2)$
2. For $i = 1$ to $N_r$:
     if $i = 6$ or 12 (or 18 for Camellia-192/256),
         $\widehat{L}_i = \mathbf{F}(L_{i-1}, K_i) \oplus R_{i-1}$, $\widehat{R}_i = L_{i-1}$;
         $L_i = \mathbf{FL}(\widehat{L}_i, KI_{\frac{i}{3}-1})$, $R_i = \mathbf{FL}^{-1}(\widehat{R}_i, KI_{\frac{i}{3}})$;
     else
         $L_i = \mathbf{F}(L_{i-1}, K_i) \oplus R_{i-1}$, $R_i = L_{i-1}$;
3. Ciphertext $C = (R_{N_r} \oplus KW_3)||(L_{N_r} \oplus KW_4)$.
```

We refer to the ith iteration of Step 2 in the above description as Round i, and write $K_{i,j}$ for the j-th byte of K_i, $(1 \leqslant j \leqslant 8)$.

[2] Here we give only the subkeys concerned in this paper, $(K_A \lll 0)[65 \sim 128]$ represents bits $(65, 66, \cdots, 128)$ of $(K_A \lll 0)$, and so on.

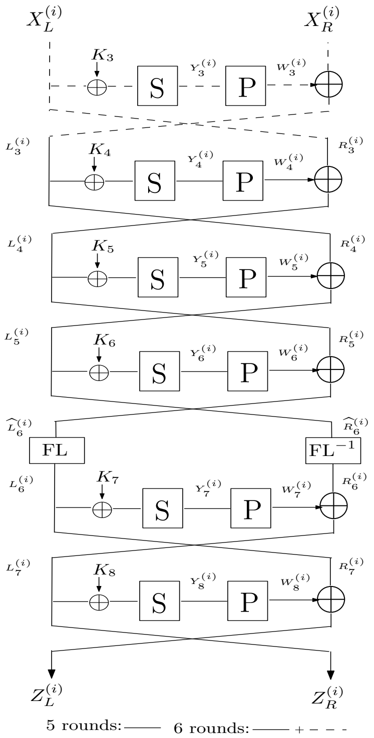

Fig. 1. 5/6-round Camellia with $\mathbf{FL}/\mathbf{FL}^{-1}$

3 Properties for 5 and 6-Round Camellia

We assume the 5-round Camellia is from Rounds 4 to 8 (including the $\mathbf{FL}/\mathbf{FL}^{-1}$ functions between Rounds 6 and 7), and the 6-round Camellia is from Rounds 3 to 8; see Fig. 1. These properties are given below, and their proof is given in the Appendix.

Proposition 1. *Suppose a set of 256 sixteen-byte values* $X^{(i)} = (X_L^{(i)}||X_R^{(i)}) = (m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, x^{(i)}, m_9, m_{10}, m_{11}, m_{12}, m_{13}, m_{14}, m_{15})$ *with* $x^{(i)}$ *taking all the possible values in* $\{0,1\}^8$ *and the other 15 bytes* $m_1, m_2, \cdots, m_{15}$ *fixed to arbitrary values,* $(i = 1, \cdots, 256)$*. Then:*

1. *If* $Z^{(i)} = (Z_L^{(i)}||Z_R^{(i)})$ *is the result of encrypting* $X^{(i)}$ *using Rounds 4 to 8 with the* $\mathbf{FL}/\mathbf{FL}^{-1}$ *functions between Rounds 6 and 7, then* $\mathbf{P}^{-1}(Z_R^{(i)})[49 \sim (49+\omega)]$ *can be expressed with a function of* $x^{(i)}$ *and* $100 + 15 \times \omega$ *constant 1-bit parameters* $c_1, c_2, \cdots, c_{100+15\times\omega}$*, written* $\Theta_{c_1,c_2,\cdots,c_{100+15\times\omega}}(x^{(i)})$*, where* $0 \leqslant \omega \leqslant 6$*.*

2. If $Z^{(i)} = (Z_L^{(i)} || Z_R^{(i)})$ is the result of encrypting $X^{(i)}$ using Rounds 3 to 8 with the $\mathbf{FL}/\mathbf{FL}^{-1}$ *functions between Rounds 6 and 7, then* $\mathbf{P}^{-1}(Z_R^{(i)})[41 \sim (41+\omega)]$ *can be expressed with a function of $x^{(i)}$ and* $164 + 15 \times \omega$ *constant 1-bit parameters $c'_1, c'_2, \cdots, c'_{164+15\times\omega}$, written $\Upsilon_{c'_1, c'_2, \cdots, c'_{164+15\times\omega}}(x^{(i)})$, where* $0 \leqslant \omega \leqslant 6$.

Note that it can be seen from the proof that several 1-bit constants can be cancelled if we take XOR under two different inputs, but such a resulting attack is termed a HO-MitM attack [24], which has a slightly different tradeoff on data/time/memory compared with our attack given below, mostly on memory.

4 Attacking 10-Round Camellia-128

A simple analysis on the key schedule of Camellia-128 reveals the following property.

Property 1 *For Camellia-128, given a value of* $(K_{2,1}, K_{2,2}, K_{2,3}, K_{2,5}, K_{2,8}, K_{3,1})$ *there are only* 60 *unknown bits of* $(K_{9,7}, K_{10,3}, K_{10,4}, K_{10,5}, K_{10,6}, K_{10,8}, K_{11})$.

The 5-round property given in Proposition 1-1 can enable us to break 10-round Camellia-128 with $\mathbf{FL}/\mathbf{FL}^{-1}$ functions. Below is the procedure for attacking Rounds 2 to 11, where the 5-round property with $\omega = 0$ is used from Rounds 4 to 8, and the approach used to choose plaintexts with δ was introduced in [23].

1. For each of 2^{100} possible values of the 100 one-bit parameters $c_1, c_2, \cdots, c_{100}$, precompute $\Theta_{c_1, c_2, \cdots, c_{100}}(z)$ sequentially for $z = 0, 1, \cdots, 255$. Store the 2^{100} 256-bit sequences in a hash table $\mathcal{L}_\Theta$.
2. Randomly choose six 8-bit constants $\gamma_1, \gamma_2, \cdots, \gamma_6$, and define a secret parameter δ to be
$$\delta = \gamma_4 \oplus \gamma_5 \oplus \gamma_6 \oplus \mathrm{S}_4(\gamma_1 \oplus K_{2,4}) \oplus \mathrm{S}_6(\gamma_2 \oplus K_{2,6}) \oplus \mathrm{S}_7(\gamma_3 \oplus K_{2,7}).$$
3. Guess a value for $(K_{2,1}, K_{2,2}, K_{2,3}, K_{2,5}, K_{2,8}, K_{3,1}, \delta)$, and we denote the guessed value by $(K^*_{2,1}, K^*_{2,2}, K^*_{2,3}, K^*_{2,5}, K^*_{2,8}, K^*_{3,1}, \delta^*)$. Then for $x = 0, 1, \cdots, 255$, choose plaintext $P^{(x)} = (P_L^{(x)}, P_R^{(x)})$ in the following way ($x = 0, 1, \cdots, 255$), where $\alpha_1, \alpha_2, \cdots, \alpha_5, \beta_1, \beta_2, \cdots, \beta_7$ are randomly chosen 8-bit constants:
$$P_L^{(x)} = \begin{pmatrix} \mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_1 \\ \mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_2 \\ \mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_3 \\ \gamma_1 \\ \mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_4 \\ \gamma_2 \\ \gamma_3 \\ \mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_5 \end{pmatrix}^{\mathrm{T}},$$
$$P_R^{(x)} = \mathbf{P} \begin{pmatrix} \mathrm{S}_1(\mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_1 \oplus K^*_{2,1}) \\ \mathrm{S}_2(\mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_2 \oplus K^*_{2,2}) \\ \mathrm{S}_3(\mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_3 \oplus K^*_{2,3}) \\ \gamma_4 \\ \mathrm{S}_5(\mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_4 \oplus K^*_{2,5}) \\ \gamma_5 \\ \gamma_6 \\ \mathrm{S}_8(\mathrm{S}_1(x \oplus K^*_{3,1}) \oplus \alpha_5 \oplus K^*_{2,8}) \end{pmatrix}^{\mathrm{T}} \oplus \begin{pmatrix} x \oplus \delta^* \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \end{pmatrix}^{\mathrm{T}}.$$

In a chosen-plaintext attack scenario, obtain the ciphertexts for the plaintexts; we denote by $C^{(x)}$ the ciphertext for plaintext $P^{(x)}$.

4. Guess a value for $(K_{9,7}, K_{10,3}, K_{10,4}, K_{10,5}, K_{10,6}, K_{10,8}, K_{11})$, and we denote the guessed value by $(K^*_{9,7}, K^*_{10,3}, K^*_{10,4}, K^*_{10,5}, K^*_{10,6}, K^*_{10,8}, K^*_{11})$. Then partially decrypt every ciphertext $C^{(x)}$ with $(K^*_{10,3}, K^*_{10,4}, K^*_{10,5}, K^*_{10,6}, K^*_{10,8}, K^*_{11})$ to get the corresponding value for bytes $(1, 2, \cdots, 8, 15)$ just before Round 10; and we denote it by $(L_9^{(x)}, R_{9,7}^{(x)})$. Next, compute
$$T^{(x)} = \mathbf{P}^{-1}(L_9^{(x)})[49] \oplus \mathrm{S}_7(R_{9,7}^{(x)} \oplus K^*_{9,7})[49].$$
Finally, check whether the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Theta$; if so, record the guessed value $(K^*_{2,1}, K^*_{2,2}, K^*_{2,3}, K^*_{2,5}, K^*_{2,8}, K^*_{3,1}, K^*_{9,7}, K^*_{10,3}, K^*_{10,4}, K^*_{10,5}, K^*_{10,6}, K^*_{10,8}, K^*_{11})$ and execute Step 5; otherwise, repeat Step 1 with another subkey guess (if all the subkey possibilities are tested in Step 4, repeat Step 3 with another subkey guess).
5. For every recorded value for $(K_{10,3}, K_{10,4}, K_{10,5}, K_{10,6}, K_{10,8})$, exhaustively search the remaining 11 key bytes.

The attack requires $256 \times 2^{56} = 2^{64}$ chosen plaintexts. The one-off precomputation requires a memory of $2^{100} \times 256 \times \frac{1}{8} = 2^{105}$ bytes, and has a time complexity of $2^{100} \times 256 \times 2 \times \frac{1}{10} \approx 2^{109.7}$ 10-round Camellia-128 encryptions under the rough estimate that a computation of $\Theta_{c_1,c_2,\cdots,c_{100}}(z)$ equals 2 one-round Camellia-128 encryptions in terms of time. If the guessed value $(K^*_{2,1}, K^*_{2,2}, K^*_{2,3}, K^*_{2,5}, K^*_{2,8}, K^*_{3,1}, \delta^*)$ is correct, the input to Round 4 must have the form $(m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, x, m_9, m_{10}, m_{11}, m_{12}, m_{13}, m_{14}, m_{15})$, where $m_1, \cdots, m_{15}$ are indeterminate constants.

The time complexity of Step 3 is $2^{56} \times 256 \times \frac{1+5}{8 \times 10} \approx 2^{60.3}$ 10-round Camellia-128 encryptions. Folllowing Property 1, we learn that the time complexity of Step 4 is approximately $2^{56+60} \times 256 \times \frac{8+5+1}{8 \times 10} \approx 2^{121.5}$ 10-round Camellia-128 encryptions.

In Step 4, if the guessed value $(K^*_{2,1}, K^*_{2,2}, K^*_{2,3}, K^*_{2,5}, K^*_{2,8}, K^*_{3,1}, \delta^*, K^*_{9,7}, K^*_{10,3}, K^*_{10,4}, K^*_{10,5}, K^*_{10,6}, K^*_{10,8}, K^*_{11})$ is correct, the sequence $(T^{(0)}, \cdots, T^{(255)})$ must match a sequence in $\mathcal{L}_\Theta$; if the guessed value $(K^*_{2,1}, K^*_{2,2}, K^*_{2,3}, K^*_{2,5}, K^*_{2,8}, K^*_{3,1}, \delta^*, K^*_{9,7}, K^*_{10,3}, K^*_{10,4}, K^*_{10,5}, K^*_{10,6}, K^*_{10,8}, K^*_{11})$ is wrong, the probability that the sequence $(T^{(0)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Theta$ is $1 - \binom{2^{100}}{0}(2^{-256})^0(1 - 2^{-256})^{2^{100}} \approx 2^{-256} \times 2^{100} = 2^{-156}$, (assuming the event has a binomial distribution). Consequently, it is expected that at most $2^{56+60} \times 2^{-156} = 2^{-40}$ values for $(K_{2,1}, K_{2,2}, K_{2,3}, K_{2,5}, K_{2,8}, K_{3,1}, K_{9,7}, K_{10,3}, K_{10,4}, K_{10,5}, K_{10,6}, K_{10,8}, K_{11})$ are recorded in Step 4. Since a total of 40 bits of K_L can be known from $(K_{10,3}, K_{10,4}, K_{10,5}, K_{10,6}, K_{10,8})$, Step 5 takes at most 2^{88} 10-round Camellia-128 encryptions to find the correct 128-bit user key.

Therefore, the attack has a memory complexity of 2^{105} bytes and a total time complexity of approximately $2^{121.5}$ 10-round Camellia-128 encryptions.

Note that we can also attack Rounds 8 to 17 by applying the 5-round property with $\omega = 0$ from Rounds 10 to 14, where we guess $(K_{8,1}, K_{8,2}, K_{8,3}, K_{8,5}, K_{8,8},$

$K_{9,1}, K_{15,7}, K_{16,3}, K_{16,4}, K_{16,5}, K_{16,6}, K_{16,8}, K_{17}$), plus a secret 8-bit parameter δ (which has a similar meaning as the δ defined above). Similarly, the attack has the same data and memory complexity as the above 10-round Camellia-128 attack, but a total time complexity of approximately $2^{56+65} \times 256 \times \frac{8+5+1}{8\times 10} \approx 2^{126.5}$ 10-round Camellia-128 encryptions.

5 Attacking 11-Round Camellia-192

Both the 5 and 6-round properties given in Proposition 1 can be used to attack 11-round Camellia-192 with $\mathbf{FL}/\mathbf{FL}^{-1}$ functions. We first describe such an attack based on the 5-round property. The following property holds for Camellia-192.

Property 2 *For Camellia-192, there is no overlapping bit between* $(K_{13}, K_{14}, K_{15,1})$ *and* $(K_{21,7}, K_{22,3}, K_{22,4}, K_{22,5}, K_{22,6}, K_{22,8}, K_{23})$.

Then we can give the following procedure for attacking Rounds 13 to 23 of Camellia-192 with $\mathbf{FL}/\mathbf{FL}^{-1}$ functions, where we choose where we choose $\omega = 0$.

1. For each of 2^{100} possible values of the 100 one-bit parameters $c_1, c_2, \cdots, c_{100}$, precompute $\Theta_{c_1,c_2,\cdots,c_{100}}(z)$ sequentially for $z = 0, 1, \cdots, 255$. Store the 2^{100} 256-bit sequences in a hash table $\mathcal{L}_\Theta$.
2. Guess a value for $(K_{13}, K_{14}, K_{15,1})$, and we denote the guessed value by $(K^*_{13}, K^*_{14}, K^*_{15,1})$. Then for $x = 0, 1, \cdots, 255$, choose plaintext $P^{(x)} = (P^{(x)}_L, P^{(x)}_R)$ in the following way, where $\alpha_1, \alpha_2, \cdots, \alpha_8, \beta_1, \beta_2, \cdots, \beta_7$ are randomly chosen 8-bit constants:

$$P^{(x)}_L = \mathbf{P}\begin{pmatrix} \mathrm{S}_1(\mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_1 \oplus K^*_{14,1}) \\ \mathrm{S}_2(\mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_2 \oplus K^*_{14,2}) \\ \mathrm{S}_3(\mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_3 \oplus K^*_{14,3}) \\ \mathrm{S}_4(\alpha_4 \oplus K^*_{14,4}) \\ \mathrm{S}_5(\mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_5 \oplus K^*_{14,5}) \\ \mathrm{S}_6(\alpha_6 \oplus K^*_{14,6}) \\ \mathrm{S}_7(\alpha_7 \oplus K^*_{14,7}) \\ \mathrm{S}_8(\mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_8 \oplus K^*_{14,8}) \end{pmatrix}^{\mathtt{T}} \oplus \begin{pmatrix} x \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \end{pmatrix}^{\mathtt{T}}.$$

$$P^{(x)}_R = \mathbf{F}(P^{(x)}_L, K^*_{13}) \oplus \begin{pmatrix} \mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_1 \\ \mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_2 \\ \mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_3 \\ \alpha_4 \\ \mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \mathrm{S}_1(x \oplus K^*_{15,1}) \oplus \alpha_8 \end{pmatrix}^{\mathtt{T}},$$

   In a chosen-plaintext attack scenario, obtain the ciphertexts for the plaintexts; we denote by $C^{(x)}$ the ciphertext for plaintext $P^{(x)}$.
3. Guess a value for $(K_{21,7}, K_{22,3}, K_{22,4}, K_{22,5}, K_{22,6}, K_{22,8}, K_{23})$, and we denote the guessed value by $(K^*_{21,7}, K^*_{22,3}, K^*_{22,4}, K^*_{22,5}, K^*_{22,6}, K^*_{22,8}, K^*_{23})$. Then, partially decrypt every ciphertext $C^{(x)}$ with $(K^*_{22,3}, K^*_{22,4}, K^*_{22,5}, K^*_{22,6}, K^*_{22,8},$

K^*_{23}) to get the corresponding value for bytes $(1, 2, \cdots, 8, 14)$ just before Round 22; and we denote it by $(L^{(x)}_{21}, R^{(x)}_{21,7})$. Next, compute

$$T^{(x)} = \mathbf{P}^{-1}(L^{(x)}_{21})[49] \oplus \mathrm{S}_6(R^{(x)}_{21,7} \oplus K^*_{21,7})[49].$$

Finally, check whether the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Theta$; if so, record the guessed $(K^*_{13}, K^*_{14}, K^*_{15,1}, K^*_{21,7}, K^*_{22,3}, K^*_{22,4}, K^*_{22,5}, K^*_{22,6}, K^*_{22,8}, K^*_{23})$ and execute Step 4; otherwise, repeat Step 3 with another subkey guess (if all the subkey possibilities are tested in Step 3, repeat Step 2 with another subkey guess).

4. For every recorded value for (K_{13}, K_{23}), exhaustively search the remaining 64 key bits.

There are $2^{64+8} = 2^{72}$ possible values for $(K_{13}, K_{14}, K_{15,1})$, and thus the attack requires $256 \times 2^{72} = 2^{80}$ chosen plaintexts. The one-off precomputation requires a memory of $2^{100} \times 256 \times \frac{1}{8} = 2^{105}$ bytes, and has a time complexity of $2^{100} \times 256 \times 2 \times \frac{1}{11} \approx 2^{106.6}$ 11-round Camellia-192 encryptions under the rough estimate that a computation of $\Theta_{c_1,c_2,\cdots,c_{100}}(z)$ equals 2 one-round Camellia-192 encryptions in terms of time. If the guessed value $(K^*_{13}, K^*_{14}, K^*_{15,1})$ is correct, the input to Round 16 must have the form $(m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, x, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6, \beta_7)$, where $m_1, m_2, \cdots, m_8$ are indeterminate constants.

The time complexity of Step 2 is $2^{72} \times 256 \times \frac{1+8+8}{8\times 11} \approx 2^{77.7}$ 11-round Camellia-192 encryptions. By Property 2, the time complexity of Step 3 is about $2^{72+112} \times 256 \times \frac{8+5+1}{8\times 11} \approx 2^{189.4}$ 11-round Camellia-192 encryptions.

In Step 3, if the guessed value $(K^*_{13}, K^*_{14}, K^*_{15,1}, K^*_{21,7}, K^*_{22,3}, K^*_{22,4}, K^*_{22,5}, K^*_{22,6}, K^*_{22,8}, K^*_{23})$ is correct, the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ must match a sequence in $\mathcal{L}_\Theta$; for a wrong guess of $(K_{13}, K_{14}, K_{15,1}, K_{21,7}, K_{22,3}, K_{22,4}, K_{22,5}, K_{22,6}, K_{22,8}, K_{23})$, the probability that the sequence $(T^{(0)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Theta$ is approximately $1 - \binom{2^{100}}{0}(2^{-256})^0(1-2^{-256})^{2^{100}} \approx 2^{-256} \times 2^{100} = 2^{-156}$, (assuming the event has a binomial distribution). Consequently, it is expected that about $2^{72+112} \times 2^{-156} = 2^{28}$ values for $(K_{13}, K_{14}, K_{15,1}, K_{21,7}, K_{22,3}, K_{22,4}, K_{22,5}, K_{22,6}, K_{22,8}, K_{23})$ are recorded in Step 3. Since 64 bits of K_L can be known from K_{23} and K_R can be known from K_{13}, Step 4 takes at most $2^{64} \times 2^{28} = 2^{92}$ 11-round Camellia-192 encryptions.

Therefore, the attack has a memory complexity of 2^{105} bytes, and has a total time complexity of $2^{189.4}$ 11-round Camellia-192 encryptions.

Next, we briefly describe a 11-round Camellia-192 attack based on that 6-round property, where we choose $\omega = 0$. We attack Rounds 13 to 23, and the attack procedure is similar to the 12-round Camellia-256 attack presented in Section 6, except the following two points: (1) There are 164 one-bit parameters $c'_1, c'_2, \cdots, c'_{164}$ in the off-line precomputation phase; and (2) We append only three rounds (i.e., Rounds 21 to 23) after the 6-round property. There are only 2^{40} possible values for $(K_{13,1}, K_{13,2}, K_{13,3}, K_{13,5}, K_{13,8}, K_{14,1})$. After a similar analysis, we get that the off-line precomputation requires a memory of $2^{164} \times 256 \times \frac{1}{8} = 2^{169}$ bytes and has a time complexity of $2^{164} \times 256 \times 3 \times \frac{1}{11} \approx 2^{170.2}$ 11-round Camellia-192 encryptions. The attack requires $256 \times 2^{40+8} =$

2^{56} chosen plaintexts, and the time complexity in the key-recovery phase is approximately $2^{48+112}\times 256\times\frac{8+5+1}{8\times 11}\approx 2^{165.4}$ 11-round Camellia-192 encryptions. We can obtain a data–memory–time tradeoff [14] version from this 11-round Camellia-192 attack, which has a data complexity of $2^{59.4}$ chosen plaintexts, a memory complexity of $2^{167.6}$ bytes and a total time complexity of $2^{169.8}$ 11-round Camellia-192 encryptions.

6 Attacking 12-Round Camellia-256

We have the following property for Camellia-256.

Property 3 *For Camellia-256, given a value for* $(K_{7,1}, K_{7,2}, K_{7,3}, K_{7,5}, K_{7,8}, K_{8,1})$ *there are only 158 unknown bits for* $(K_{15,6}, K_{16,2}, K_{16,3}, K_{16,5}, K_{16,7}, K_{16,8}, K_{17}, K_{18})$.

We can use the 6-round property given in Proposition 1-2 to mount an attack on 12-round Camellia-256 with $\mathbf{FL}/\mathbf{FL}^{-1}$ functions. We attack Rounds 7 to 18, and use the property with $\omega = 1$. The attack procedure is as follows.

1. For each of 2^{179} possible values of the 179 one-bit parameters $c'_1, c'_2, \cdots, c'_{179}$, precompute $\Upsilon_{c'_1,c'_2,\cdots,c'_{179}}(z)$ sequentially for $z = 0, 1, \cdots, 255$. Store the 2^{179} 512-bit sequences in a hash table $\mathcal{L}_\Upsilon$.
2. Randomly choose six 8-bit constants $\gamma_1, \gamma_2, \cdots, \gamma_6$, and define a secret parameter δ to be
$$\delta = \gamma_4 \oplus \gamma_5 \oplus \gamma_6 \oplus \mathrm{S}_4(\gamma_1 \oplus K_{7,4}) \oplus \mathrm{S}_6(\gamma_2 \oplus K_{7,6}) \oplus \mathrm{S}_7(\gamma_3 \oplus K_{7,7}).$$
3. Guess a value for $(K_{7,1}, K_{7,2}, K_{7,3}, K_{7,5}, K_{7,8}, K_{8,1}, \delta)$, and we denote the guessed value by $(K^*_{7,1}, K^*_{7,2}, K^*_{7,3}, K^*_{7,5}, K^*_{7,8}, K^*_{8,1}, \delta^*)$. Then for $x = 0, 1, \cdots,$ 255, choose plaintext $P^{(x)} = (P_L^{(x)}, P_R^{(x)})$ in the same way as in the 10-round Camellia-128 attack described in Section 4, $(x = 0, 1, \cdots, 255)$. In a chosen-plaintext attack scenario, obtain the ciphertexts for the plaintexts; we denote by $C^{(x)}$ the ciphertext for plaintext $P^{(x)}$.
4. Guess a value for $(K_{15,6}, K_{16,2}, K_{16,3}, K_{16,5}, K_{16,7}, K_{16,8}, K_{17}, K_{18})$, and we denote the guessed value by $(K^*_{15,6}, K^*_{16,2}, K^*_{16,3}, K^*_{16,5}, K^*_{16,7}, K^*_{16,8}, K^*_{17}, K^*_{18})$. Then partially decrypt every ciphertext $C^{(x)}$ with $(K^*_{16,2}, K^*_{16,3}, K^*_{16,5}, K^*_{16,7}, K^*_{16,8}, K^*_{17}, K^*_{18})$ to get the corresponding value for bytes $(1, 2, \cdots, 8, 14)$ just before Round 16; and we denote it by $(L_{15}^{(i,x)}, R_{15,6}^{(i,x)})$. Next, compute
$$T^{(x)} = \mathbf{P}^{-1}(L_{15}^{(x)})[41 \sim 42] \oplus \mathrm{S}_6(R_{15,6}^{(x)} \oplus K^*_{15,6})[41 \sim 42].$$
Finally, check whether the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Upsilon$; if so, record the guessed value $(K^*_{7,1}, K^*_{7,2}, K^*_{7,3}, K^*_{7,5}, K^*_{7,8}, K^*_{8,1}, K^*_{15,6}, K^*_{16,2}, K^*_{16,3}, K^*_{16,5}, K^*_{16,7}, K^*_{16,8}, K^*_{17}, K^*_{18})$ and execute Step 5; otherwise, repeat Step 4 with another subkey guess (if all the subkey possibilities are tested in Step 4, repeat Step 3 with another subkey guess).

5. For every recorded value for (K_{17}, K_{18}), exhaustively search the remaining 16 key bytes.

The attack requires $256 \times 2^{56} = 2^{64}$ chosen plaintexts. The one-off precomputation requires a memory of $2^{179} \times 256 \times \frac{2}{8} = 2^{185}$ bytes, and has a time complexity of $2^{179} \times 256 \times 3 \times \frac{1}{12} = 2^{185}$ 12-round Camellia-256 encryptions under the rough estimate that a computation of $\Upsilon_{c'_1,c'_2,\cdots,c'_{179}}(z)$ equals 3 one-round Camellia-256 encryptions in terms of time. If the guessed value $(K^*_{7,1}, K^*_{7,2}, K^*_{7,3}, K^*_{7,5}, K^*_{7,8}, K^*_{8,1}, \delta^*)$ is correct, the input to Round 9 must have the form $(m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8, x, m_9, m_{10}, m_{11}, m_{12}, m_{13}, m_{14}, m_{15})$, where $m_1, m_2, \cdots, m_{15}$ are indeterminate constants.

The time complexity of Step 3 is $2^{56} \times 256 \times \frac{1+5}{8\times 12} = 2^{60}$ 12-round Camellia-256 encryptions. By Property 3, the time complexity of Step 4 is approximately $2^{56+158} \times 256 \times \frac{8+8+5+1}{8\times 12} \approx 2^{219.9}$ 12-round Camellia-256 encryptions.

In Step 4, for the correct guess of $(K_{7,1}, K_{7,2}, K_{7,3}, K_{7,5}, K_{7,8}, K_{8,1}, \delta, K_{15,6}, K_{16,2}, K_{16,3}, K_{16,5}, K_{16,7}, K_{16,8}, K_{17}, K_{18})$, the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ must match a sequence in $\mathcal{L}_\Upsilon$; for a wrong guess of $(K_{7,1}, K_{7,2}, K_{7,3}, K_{7,5}, K_{7,8}, K_{8,1}, \delta, K_{15,6}, K_{16,2}, K_{16,3}, K_{16,5}, K_{16,7}, K_{16,8}, K_{17}, K_{18})$, the probability that the sequence $(T^{(0)}, T^{(1)}, \cdots, T^{(255)})$ matches a sequence in $\mathcal{L}_\Upsilon$ is approximately $1 - \binom{2^{179}}{0}(2^{-512})^0(1-2^{-512})^{2^{179}} \approx 2^{-512} \times 2^{179} = 2^{-333}$, (assuming the event has a binomial distribution). Consequently, it is expected that at most $2^{56+158} \times 2^{-333} = 2^{-119}$ values for $(K_{7,1}, K_{7,2}, K_{7,3}, K_{7,5}, K_{7,8}, K_{8,1}, K_{15,6}, K_{16,2}, K_{16,3}, K_{16,5}, K_{16,7}, K_{16,8}, K_{17}, K_{18})$ are recorded in Step 4. Since K_L can be known from (K_{17}, K_{18}), Step 5 takes at most 2^{128} 12-round Camellia-256 encryptions.

Therefore, the attack has a memory complexity of 2^{185} bytes and a total time complexity of approximately $2^{219.9}$ 12-round Camellia-256 encryptions.

It is worthy to observe that we can also apply the 6-round property with $\omega = 1$ to break two other series of 12-round Camellia-256 with $\mathbf{FL}/\ \mathbf{FL}^{-1}$ functions, namely Rounds 1 to 12 and Rounds 13 to 24. Similarly, the attack has the same data and memory complexity as the above 12-round Camellia-256 attack, but has a total time complexity of approximately $2^{56+176} \times 256 \times \frac{8+8+5+1}{8\times 12} \approx 2^{237.9}$ 12-round Camellia-256 encryptions.

7 Concluding Remarks

In this paper, we have analysed the security of Camellia against the MitM attack in detail, following the work in [24]. We have presented 5 and 6-round properties of Camellia, that can be used to conduct MitM attacks on 10-round Camellia-128, 11-round Camellia-192 and 12-round Camellia-256, all of which include the FL/FL^{-1} functions but do not include whitening operations. The presented attacks are theoretical, like most cryptanalytic attacks on block ciphers.

Our results show that as far as Camellia is concerned, the semi-advanced MitM attack technique is more efficient than or at least as efficient as the advanced cryptanalytic techniques studied, except impossible differential cryptanalysis; in this latter case the MitM attacks are now one or two rounds inferior to the best newly emerging impossible differential cryptanalysis results in [2,22].

We attribute these MitM attacks to the fact that the FL^{-1} function does not have a good avalanche effect (i.e., an output bit relied on a large number of the bits of the input and the subkey used). If the FL^{-1} function were modified to have a good avalanche effect, then those MitM properties would involve a large number of unknown 1-bit constant parameters, and the resulting MitM attacks would be ineffective for the resulting cipher, but nevertheless it does not necessarily resist the HO-MitM attack technique, for those HO-MitM attacks described in [24] work as long as that integral property of Camellia holds (canceling the FL^{-1} function). Actually, if the $\mathrm{FL}/\mathrm{FL}^{-1}$ functions had had a good avalanche effect, the Camellia cipher could have withstood the best currently known cryptanalytic results that are the newly emerging impossible differential cryptanalysis results [2, 22]. In this sense, the $\mathrm{FL}/\mathrm{FL}^{-1}$ functions do play an important role in the security of Camellia.

Acknowledgments. The authors are grateful to the anonymous referees for their comments on this paper.

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.) SAC 2000. LNCS, vol. 2012, pp. 39–56. Springer, Heidelberg (2001)
2. Bai, D., Li, L.: New impossible differential attacks on Camellia. In: Ryan, M.D., Smyth, B., Wang, G., (eds.) ISPEC 2012. LNCS, vol. 7232, pp. 80–96. Springer, Heidelberg (2012)
3. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23. Springer, Heidelberg (1999)
4. Biham, E., Dunkelman O., Keller, N.: The rectangle attack — rectangling the Serpent. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 340–357. Springer, Heidelberg (2001)
5. Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology 4(1), 3–72. Springer (1991)
6. Chen, J., Jia, K., Yu, H., Wang, X.: New impossible differential attacks of reduced-round Camellia-192 and Camellia-256. In: Hawkes, P., Parampalli, U. (eds.) ACISP 2011. LNCS, vol. 6812, pp. 16–33. Springer, Heidelberg (2011)
7. CRYPTREC — Cryptography Research and Evaluatin Committees, report 2002.
8. Daemen, J., Knudsen, L.R., Rijmen, V.: The block cipher Square. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 149–165. Springer, Heidelberg (1997)
9. Demirci, H., Selçuk, A. A.: A meet-in-the-middle attack on 8-round AES. In: Nyberg, K. (ed.) FSE 2008. LNCS, vol. 5086, pp. 116–126. Springer, Heidelberg (2008)
10. Diffie, W., Hellman, M.: Exhaustive cryptanalysis of the NBS data encryption standard. Computer 10(6), pp. 74–84. IEEE (1977)
11. Duo, L., Li, C., Feng, K.: New observation on Camellia. In: Preneel, B., Tavares, S.E. (eds.) SAC 2005. LNCS, vol. 3897, pp. 51–64. Springer, Heidelberg (2006)

12. Duo, L., Li, C., Feng, K.: Square like attack on Camellia. In: Qing, S., Imai, H., Wang, G. (eds.) ICICS 2007. LNCS, vol. 4861, pp. 269–283. Springer, Heidelberg (2007)
13. Hatano, Y., Sekine, H., Kaneko, T.: Higher order differential attack of Camellia(II). In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, pp.39–56. Springer, Heidelberg (2003)
14. Hellman, M.E.: A cryptanalytic time–memory trade-off. IEEE Transcations on Information Theory 26(4), 401–406 (1980)
15. Hu, Y., Zhang, Y., Xiao, G.: Integral cryptanalysis of SAFER+. Electronics Letters 35(17), 1458–1459. IEE (1999)
16. International Standardization of Organization (ISO), International Standard – ISO/IEC 18033-3, Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers, 2005
17. Knudsen, L.R.: Truncated and higher order differentials. In: Preneel, B. (ed.) FSE 1994. LNCS, vol. 1008, pp. 196–211. Springer, Heidelberg (1995)
18. Knudsen, L.R.: DEAL — a 128-bit block cipher. Technical report, Department of Informatics, University of Bergen, Norway (1998)
19. Knudsen, L.R., Wagner, D.: Integral cryptanalysis. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 112–127. Springer, Heidelberg (2002)
20. Lai, X.: Higher order derivatives and differential cryptanalysis. In: Communications and Cryptography, pp. 227–233. Academic Publishers (1994)
21. Li, L., Chen, Z., Jia, K.: New impossible differential cryptanalysis of reduced-round Camellia. In: Lin, D., Tsudik, G., Wang, X. (eds.) CANS 2011. LNCS, vol. 7092, pp. 26–39. Springer, Heidelberg (2011)
22. Liu, Y., Li, L., Gu, D., Wang, X., Liu, Z., Chen, J., Li, W.: New observations on impossible differential cryptanalysis of reduced-round Camellia. In: Canteaut, A. (ed.) FSE 2012. LNCS 7549, to appear. Springer, Heidelberg (2012)
23. Lu, J., Wei, Y., Kim, J., Fouque, P.-A.: Cryptanalysis of reduced versions of the Camellia block cipher. In: Miri, A., Vaudenay, S. (eds.) Pre-proceedings of SAC 2011. `http://sac2011.ryerson.ca/SAC2011/LWKF.pdf`. An editorially revised version is to appear in IET Information Security.
24. Lu, J., Wei, Y., Kim, J., Pasalic, E.: The higher-order meet-in-the-middle attack and its application to the Camellia block cipher. Presented in part at the First Asian Workshop on Symmetric Key Cryptography (ASK 2011), August 2011, Singapore. `https://sites.google.com/site/jiqiang/HO-MitM.pdf`
25. Matsui, M.: Linear cryptanalysis method for DES cipher. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 386–397. Springer, Heidelberg (1994)
26. NESSIE — New European Schemes for Signatures, Integrity, and Encryption, final report of European project IST-1999-12324 (2004)
27. Shirai, T.: Differential, linear, boomerang and rectangle cryptanalysis of reduced-Round Camellia. In Proceedings of the Third NESSIE Workshop (2002)
28. 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 2001. LNCS, vol. 2248, pp. 193–207. Springer, Heidelberg (2001)
29. Wu, W., Feng, D., Chen, H.: Collision attack and pseudorandomness of reduced-round Camellia. In: Handschuh, H., Hasan, M.A. (eds.) SAC 2004. LNCS, vol. 3357, pp. 256–270. Springer, Heidelberg (2005)
30. Wagner, D.: The boomerang attack. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 156–170. Springer, Heidelberg (1999)

31. Yeom, Y., Park, S., Kim, Iljun.: On the security of Camellia against the square attack. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2356, pp. 89–99. Springer, Heidelberg (2002)
32. Yeom, Y., Park, S., Kim, I.: A study of integral type cryptanalysis on Camellia. In Proceedings of the 2003 Symposium on Cryptography and Information Security, pp. 453–456. IEICE (2003)

Appendix: Proof of Proposition 1

First, we have the following property for the $\mathbf{FL}/\mathbf{FL}^{-1}$ functions.

Property 4 (from [24]) *Let $x_1, x_2, \cdots, x_8, y_1, y_2, \cdots, y_8$ be 8-bit blocks and KI be a 64-bit subkey.*

1. *If $(y_1||y_2||\cdots||y_8) = \mathbf{FL}(x_1||x_2||\cdots||x_8, KI)$, then*

$$\begin{aligned}
y_1 &= ((((x_1[2\sim 8]||x_2[1]) \cap KI[2\sim 9]) \oplus x_5) \cup KI[33\sim 40]) \oplus x_1,\\
y_2 &= ((((x_2[2\sim 8]||x_3[1]) \cap KI[10\sim 17]) \oplus x_6) \cup KI[41\sim 48]) \oplus x_2,\\
y_3 &= ((((x_3[2\sim 8]||x_4[1]) \cap KI[18\sim 25]) \oplus x_7) \cup KI[49\sim 56]) \oplus x_3,\\
y_4 &= ((((x_4[2\sim 8]||x_1[1]) \cap KI[26\sim 32,1]) \oplus x_8) \cup KI[57\sim 64]) \oplus x_4,\\
y_5 &= ((x_1[2\sim 8]||x_2[1]) \cap KI[2\sim 9]) \oplus x_5,\\
y_6 &= ((x_2[2\sim 8]||x_3[1]) \cap KI[10\sim 17]) \oplus x_6,\\
y_7 &= ((x_3[2\sim 8]||x_4[1]) \cap KI[18\sim 25]) \oplus x_7,\\
y_8 &= ((x_4[2\sim 8]||x_1[1]) \cap KI[26\sim 32,1]) \oplus x_8.
\end{aligned}$$

2. *If $(y_1||y_2||\cdots||y_8) = \mathbf{FL}^{-1}(x_1||x_2||\cdots||x_8, KI)$, then*

$$\begin{aligned}
y_1 &= (x_5 \cup KI[33\sim 40]) \oplus x_1,\\
y_2 &= (x_6 \cup KI[41\sim 48]) \oplus x_2,\\
y_3 &= (x_7 \cup KI[49\sim 56]) \oplus x_3,\\
y_4 &= (x_8 \cup KI[57\sim 64]) \oplus x_4,\\
y_5 &= ((((x_5[2\sim 8]||x_6[1]) \cup KI[34\sim 41]) \oplus (x_1[2\sim 8]||x_2[1])) \cap\\
&\quad KI[2\sim 9]) \oplus x_5,\\
y_6 &= ((((x_6[2\sim 8]||x_7[1]) \cup KI[42\sim 49]) \oplus (x_2[2\sim 8]||x_3[1])) \cap\\
&\quad KI[10\sim 17]) \oplus x_6,\\
y_7 &= ((((x_7[2\sim 8]||x_8[1]) \cup KI[50\sim 57]) \oplus (x_3[2\sim 8]||x_4[1])) \cap\\
&\quad KI[18\sim 25]) \oplus x_7,\\
y_8 &= ((((x_8[2\sim 8]||x_5[1]) \cup KI[58\sim 64,33]) \oplus (x_4[2\sim 8]||x_1[1])) \cap\\
&\quad KI[26\sim 32,1]) \oplus x_8.
\end{aligned}$$

When encrypting $X^{(i)}$, we denote by $Y_t^{(i)}$ the value immediately after the **S** operation of Round t, and by $W_t^{(i)}$ the value immediately after the **P** operation of Round t, $(3 \leqslant t \leqslant 8)$.

We have Eq. (1) for Rounds 4 to 8 and have Eq. (2) for Rounds 3 to 8.

$$\mathbf{P}^{-1}(Z_R^{(i)}) = \mathbf{P}^{-1}(\mathbf{FL}^{-1}(X_L^{(i)} \oplus W_5^{(i)}, KI_2)) \oplus Y_7^{(i)}. \qquad (1)$$

$$\mathbf{P}^{-1}(Z_R^{(i)}) = \mathbf{P}^{-1}(\mathbf{FL}^{-1}(X_R^{(i)} \oplus W_3^{(i)} \oplus W_5^{(i)}, KI_2)) \oplus Y_7^{(i)}. \qquad (2)$$

We first prove Proposition 1-1, and focus on encrypting $X^{(i)}$ through Rounds 4 to 8 below. The output of Round 4 is as follows, where $a_1, a_2, \cdots, a_8$ are 8-bit constants completely determined by $m_1, m_2, \cdots, m_{15}$ and K_4.

$$L_4^{(i)} = (x^{(i)} \oplus a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8), R_4^{(i)} = (m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8).$$

The output of Round 5 is as follows, where $b, b_1, \cdots, b_8$ are 8-bit constants completely determined by $m_1, m_2, \cdots, m_8$, $a_1, a_2, \cdots, a_8$ and K_5:

$$\begin{aligned} R_5^{(i)} &= (x^{(i)} \oplus a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8), \\ L_5^{(i)} &= (L_{5,1}^{(i)}, L_{5,2}^{(i)}, L_{5,3}^{(i)}, L_{5,4}^{(i)}, L_{5,5}^{(i)}, L_{5,6}^{(i)}, L_{5,7}^{(i)}, L_{5,8}^{(i)}), \end{aligned}$$

with

$$\begin{array}{lll} L_{5,1}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus b) \oplus b_1, & L_{5,2}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus b) \oplus b_2, & L_{5,3}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus b) \oplus b_3, \\ L_{5,4}^{(i)} = b_4, & L_{5,5}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus b) \oplus b_5, & L_{5,6}^{(i)} = b_6, \\ L_{5,7}^{(i)} = b_7, & L_{5,8}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus b) \oplus b_8. & \end{array}$$

The output immediately before the FL/FL^{-1} functions is as follows, where $d_1 = b_1 \oplus K_{6,1}, d_2 = b_2 \oplus K_{6,2}, d_3 = b_3 \oplus K_{6,3}, d_4 = b_5 \oplus K_{6,5}, d_5 = b_8 \oplus K_{6,8}$; and $e_1, e_2, \cdots, e_8$ are 8-bit constants completely determined by $a_1, a_2, \cdots, a_8$ and $b_1, b_2, \cdots, b_8$:

$$\begin{aligned} \widehat{R}_6^{(i)} &= (L_{5,1}^{(i)}, L_{5,2}^{(i)}, L_{5,3}^{(i)}, L_{5,4}^{(i)}, L_{5,5}^{(i)}, L_{5,6}^{(i)}, L_{5,7}^{(i)}, L_{5,8}^{(i)}), \\ \widehat{L}_6^{(i)} &= (\widehat{L}_{6,1}^{(i)}, \widehat{L}_{6,2}^{(i)}, \widehat{L}_{6,3}^{(i)}, \widehat{L}_{6,4}^{(i)}, \widehat{L}_{6,5}^{(i)}, \widehat{L}_{6,6}^{(i)}, \widehat{L}_{6,7}^{(i)}, \widehat{L}_{6,8}^{(i)}), \end{aligned}$$

with

$$\begin{aligned} \widehat{L}_{6,1}^{(i)} = \;& \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_1) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_3) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus \\ & x^{(i)} \oplus e_1, \\ \widehat{L}_{6,2}^{(i)} = \;& \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_2) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus \\ & \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus e_2, \\ \widehat{L}_{6,3}^{(i)} = \;& \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_2) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_3) \oplus \\ & \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus e_3, \\ \widehat{L}_{6,4}^{(i)} = \;& \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_2) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus e_4, \\ \widehat{L}_{6,5}^{(i)} = \;& \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_2) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus e_5, \\ \widehat{L}_{6,6}^{(i)} = \;& \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_2) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus \\ & \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus e_6, \\ \widehat{L}_{6,7}^{(i)} = \;& \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_5) \oplus e_7, \\ \widehat{L}_{6,8}^{(i)} = \;& \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_1) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus b) \oplus d_4) \oplus e_8. \end{aligned}$$

By Property 4-1, we know that $\mathbf{FL}(\widehat{L}_6^{(i)}, KI_1)[49 \sim 56]$ is determined only by $\widehat{L}_{6,3}^{(i)}, \widehat{L}_{6,4}^{(i)}, \widehat{L}_{6,7}^{(i)}, KI_1[18 \sim 25]$. Thus, $Y_7^{(i)}[49 \sim (49+\omega)] = \mathrm{S}_7(\mathbf{FL}(\widehat{L}_6^{(i)}, KI_1)[49 \sim 56] \oplus K_{7,7})[49 \sim (49+\omega)]$ is determined only by $(x^{(i)}, b, d_1, d_2, \cdots, d_5, e_3, e_4, l_1, KI_1[26 \sim 32, 1])$, where $l_1 = e_7 \oplus K_{7,7}$.

Since $X_L^{(i)} \oplus W_5^{(i)} = \widehat{R}_6^{(i)}$, by Property 4-2 we know that $\mathbf{P}^{-1}(\mathbf{FL}^{-1}(X_L^{(i)} \oplus W_5^{(i)}, KI_2))[49 \sim (49+\omega)] = \mathbf{P}^{-1}(\mathbf{FL}^{-1}(\widehat{R}_6^{(i)}, KI_2))[49 \sim (49+\omega)]$ is determined only by $(x^{(i)}, b, b_1[2 \sim (2+\omega)], b_2[2 \sim (2+\omega)], b_3[1 \sim (2+\omega)], b_4[1 \sim (1+\omega)], b_5[1 \sim (2+\omega)], b_6[1 \sim (2+\omega)], b_7[1 \sim (2+\omega)], b_8[1 \sim (1+\omega)], KI_2[2 \sim (2+\omega), 10 \sim (10+\omega), 18 \sim (18+\omega), 34 \sim (34+\omega), 42 \sim (42+\omega), 49 \sim (50+\omega), 57 \sim (57+\omega)])$.

So $\mathbf{P}^{-1}(\mathbf{FL}^{-1}(X_L^{(i)} \oplus W_5^{(i)}, KI_2))[49 \sim (49+\omega)] \oplus Y_7^{(i)}[49 \sim (49+\omega)]$ is determined by $x^{(i)}$ and $b, d_1, d_2, \cdots, d_5,\ e_3, e_4, l_1, b_1[2 \sim (2+\omega)], b_2[2 \sim (2+\omega)], b_3[1 \sim (2+\omega)], b_4[1 \sim (1+\omega)], b_5[1 \sim (2+\omega)], b_6[1 \sim (2+\omega)], b_7[1 \sim (2+\omega)], b_8[1 \sim (1+\omega)], KI_1[26 \sim 32, 1], KI_2[2 \sim (2+\omega), 10 \sim (10+\omega), 18 \sim (18+\omega), 34 \sim (34+\omega), 42 \sim (42+\omega), 49 \sim (50+\omega), 57 \sim (57+\omega)])$, a total of $100 + 15 \times \omega$ constant 1-bit parameters. Proposition 1-1 follows from Eq. (1).

We next prove Proposition 1-2. The output $(L_3^{(i)}, R_3^{(i)})$ of Round 3 is as follows, where $\widehat{a}_1, \widehat{a}_2, \cdots, \widehat{a}_8$ are 8-bit constants completely determined by $m_1, m_2, \cdots, m_{15}$ and K_3.

$$L_3^{(i)} = (x^{(i)} \oplus \widehat{a}_1, \widehat{a}_2, \widehat{a}_3, \widehat{a}_4, \widehat{a}_5, \widehat{a}_6, \widehat{a}_7, \widehat{a}_8), R_3^{(i)} = (m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8).$$

The output $(L_4^{(i)}, R_4^{(i)})$ of Round 4 is as follows, where $\widehat{b}, \widehat{b}_1, \cdots, \widehat{b}_8$ are 8-bit constants completely determined by $m_1, m_2, \cdots, m_8$, $\widehat{a}_1, \widehat{a}_2, \cdots, \widehat{a}_8$ and K_4:

$$\begin{aligned} R_4^{(i)} &= (x^{(i)} \oplus \widehat{a}_1, \widehat{a}_2, \widehat{a}_3, \widehat{a}_4, \widehat{a}_5, \widehat{a}_6, \widehat{a}_7, \widehat{a}_8), \\ L_4^{(i)} &= (L_{4,1}^{(i)}, L_{4,2}^{(i)}, L_{4,3}^{(i)}, L_{4,4}^{(i)}, L_{4,5}^{(i)}, L_{4,6}^{(i)}, L_{4,7}^{(i)}, L_{4,8}^{(i)}), \end{aligned}$$

with

$$\begin{array}{lll} L_{4,1}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_1, & L_{4,2}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_2, & L_{4,3}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_3, \\ L_{4,4}^{(i)} = \widehat{b}_4, & L_{4,5}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_5, & L_{4,6}^{(i)} = \widehat{b}_6, \\ L_{4,7}^{(i)} = \widehat{b}_7, & L_{4,8}^{(i)} = \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_8. & \end{array}$$

The output $(L_5^{(i)}, R_5^{(i)})$ of Round 5 is as follows, where $\widehat{d}_1, \widehat{d}_2, \cdots, \widehat{d}_5$ are 8-bit constants completely determined by $\widehat{b}_1, \widehat{b}_2, \cdots, \widehat{b}_8$ and K_5; and $\widehat{e}_1, \widehat{e}_2, \cdots, \widehat{e}_8$ are 8-bit constants completely determined by $\widehat{a}_1, \widehat{a}_2, \cdots, \widehat{a}_8$, $\widehat{b}_1, \widehat{b}_2, \cdots, \widehat{b}_8$ and K_5:

$$\begin{aligned} R_5^{(i)} &= (L_{4,1}^{(i)}, L_{4,2}^{(i)}, L_{4,3}^{(i)}, L_{4,4}^{(i)}, L_{4,5}^{(i)}, L_{4,6}^{(i)}, L_{4,7}^{(i)}, L_{4,8}^{(i)}), \\ L_5^{(i)} &= (L_{5,1}^{(i)}, L_{5,2}^{(i)}, L_{5,3}^{(i)}, L_{5,4}^{(i)}, L_{5,5}^{(i)}, L_{5,6}^{(i)}, L_{5,7}^{(i)}, L_{5,8}^{(i)}), \end{aligned}$$

with

$$\begin{aligned} L_{5,1}^{(i)} = {} & \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_1) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_3) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \\ & x^{(i)} \oplus \widehat{e}_1, \end{aligned}$$

$$\begin{aligned}
L_{5,2}^{(i)} = {} & \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_2) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \\
& \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \widehat{e}_2, \\
L_{5,3}^{(i)} = {} & \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_2) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_3) \oplus \\
& \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \widehat{e}_3, \\
L_{5,4}^{(i)} = {} & \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \widehat{e}_4, \\
L_{5,5}^{(i)} = {} & \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_1) \oplus \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_2) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \widehat{e}_5, \\
L_{5,6}^{(i)} = {} & \mathrm{S}_2(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_2) \oplus \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \\
& \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \widehat{e}_6, \\
L_{5,7}^{(i)} = {} & \mathrm{S}_3(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_3) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \mathrm{S}_8(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_5) \oplus \widehat{e}_7, \\
L_{5,8}^{(i)} = {} & \mathrm{S}_1(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_1) \oplus \mathrm{S}_5(\mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{d}_4) \oplus \widehat{e}_8.
\end{aligned}$$

By Property 4-1, we know that $\mathbf{FL}(\widehat{L}_6^{(i)}, KI_1)[41 \sim 48]$ is determined only by $\widehat{L}_{6,2}^{(i)}, \widehat{L}_{6,3}^{(i)}, \widehat{L}_{6,6}^{(i)}$, $KI_1[10 \sim 17]$, where

$$\begin{aligned}
\widehat{L}_{6,2}^{(i)} = {} & \mathrm{S}_1(L_{5,1}^{(i)} \oplus K_{6,1}) \oplus \mathrm{S}_2(L_{5,2}^{(i)} \oplus K_{6,2}) \oplus \mathrm{S}_4(L_{5,4}^{(i)} \oplus K_{6,4}) \oplus \mathrm{S}_5(L_{5,5}^{(i)} \oplus K_{6,5}) \oplus \\
& \mathrm{S}_7(L_{5,7}^{(i)} \oplus K_{6,7}) \oplus \mathrm{S}_8(L_{5,8}^{(i)} \oplus K_{6,8}) \oplus \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_2, \\
\widehat{L}_{6,3}^{(i)} = {} & \mathrm{S}_1(L_{5,1}^{(i)} \oplus K_{6,1}) \oplus \mathrm{S}_2(L_{5,2}^{(i)} \oplus K_{6,2}) \oplus \mathrm{S}_3(L_{5,3}^{(i)} \oplus K_{6,3}) \oplus \mathrm{S}_5(L_{5,5}^{(i)} \oplus K_{6,5}) \oplus \\
& \mathrm{S}_6(L_{5,6}^{(i)} \oplus K_{6,6}) \oplus \mathrm{S}_8(L_{5,8}^{(i)} \oplus K_{6,8}) \oplus \mathrm{S}_1(x^{(i)} \oplus \widehat{b}) \oplus \widehat{b}_3, \\
\widehat{L}_{6,6}^{(i)} = {} & \mathrm{S}_2(L_{5,2}^{(i)} \oplus K_{6,2}) \oplus \mathrm{S}_3(L_{5,3}^{(i)} \oplus K_{6,3}) \oplus \mathrm{S}_5(L_{5,5}^{(i)} \oplus K_{6,5}) \oplus \mathrm{S}_7(L_{5,7}^{(i)} \oplus K_{6,7}) \oplus \\
& \mathrm{S}_8(L_{5,8}^{(i)} \oplus K_{6,8}) \oplus \widehat{b}_6.
\end{aligned}$$

Letting $\widehat{n}_l = \widehat{e}_l \oplus K_{6,l}$ and $\widehat{o}_1 = \widehat{b}_6 \oplus K_{7,6}$, $(l = 1, 2, \cdots, 8)$, then we can learn that $Y_7^{(i,j)}[41 \sim (41{+}\omega)]$ is determined only by $(x^{(i)}, \widehat{b}, \widehat{b}_2, \widehat{b}_3, \widehat{o}_1, \widehat{d}_1, \widehat{d}_2, \cdots, \widehat{d}_5, \widehat{n}_1, \widehat{n}_2, \cdots, \widehat{n}_8, KI_1[10 \sim 17])$.

Since $\mathbf{FL}^{-1}(X_R^{(i)} \oplus W_3^{(i)} \oplus W_5^{(i)}, KI_2) = R_6^{(i)}$, then $\mathbf{P}^{-1}(\mathbf{FL}^{-1}(X_R^{(i)} \oplus W_3^{(i)} \oplus W_5^{(i)}, KI_2))[41 \sim (41 + \omega)] = \mathbf{P}^{-1}(\mathbf{FL}^{-1}(\widehat{R}_6^{(i)}, KI_2))[41 \sim (41 + \omega)]$ is determined only by $(x^{(i)}, \widehat{b}, \widehat{d}_1, \widehat{d}_2, \cdots, \widehat{d}_5, \widehat{e}_1[2 \sim (2 + \omega)], \widehat{e}_2[1 \sim (2 + \omega)], \widehat{e}_3[1 \sim (1 + \omega)], \widehat{e}_4[2 \sim (2 + \omega)], \widehat{e}_5[1 \sim (2 + \omega)], \widehat{e}_6[1 \sim (2 + \omega)], \widehat{e}_7[1 \sim (1 + \omega)], \widehat{e}_8[1 \sim (2 + \omega)], KI_2[2 \sim (2 + \omega), 10 \sim (10 + \omega), 26 \sim (26 + \omega), 34 \sim (34 + \omega), 41 \sim (42 + \omega), 49 \sim (49 + \omega), 58 \sim (58 + \omega)])$.

Hence, $\mathbf{P}^{-1}(\mathbf{FL}(X_R^{(i)} \oplus W_4^{(i)} \oplus W_6^{(i)}, KI_1))[41 \sim (41{+}\omega)] \oplus Y_7^{(i)}[41 \sim (41{+}\omega)]$ is determined by $x^{(i)}$ and $\widehat{b}, \widehat{b}_2, \widehat{b}_3, \widehat{o}_1, \widehat{d}_1, \widehat{d}_2, \cdots, \widehat{d}_5, \widehat{e}_1[2 \sim (2 + \omega)], \widehat{e}_2[1 \sim (2 + \omega)], \widehat{e}_3[1 \sim (1 + \omega)], \widehat{e}_4[2 \sim (2 + \omega)], \widehat{e}_5[1 \sim (2 + \omega)], \widehat{e}_6[1 \sim (2 + \omega)], \widehat{e}_7[1 \sim (1 + \omega)], \widehat{e}_8[1 \sim (2 + \omega)], \widehat{n}_1, \widehat{n}_2, \cdots, \widehat{n}_8, KI_1[10 \sim 17], KI_2[2 \sim (2 + \omega), 10 \sim (10{+}\omega), 26 \sim (26{+}\omega), 34 \sim (34{+}\omega), 41 \sim (42{+}\omega), 49 \sim (49{+}\omega), 58 \sim (58{+}\omega)])$, a total of $164 + 15 \times \omega$ constant 1-bit parameters. Proposition 1-2 follows from Eq. (2). □