An Improved Impossible Differential Attack on MISTY1

Orr Dunkelman[1,⋆] and Nathan Keller[2,⋆⋆]

[1] 'Ecole Normale Supérieure
Département d'Informatique,
CNRS, INRIA
45 rue d'Ulm, 75230 Paris, France
orr.dunkelman@ens.fr
[2] Einstein Institute of Mathematics, Hebrew University
Jerusalem 91904, Israel
nkeller@math.huji.ac.il

Abstract. MISTY1 is a Feistel block cipher that received a great deal of cryptographic attention. Its recursive structure, as well as the added FL layers, have been successful in thwarting various cryptanalytic techniques. The best known attacks on reduced variants of the cipher are on either a 4-round variant with the FL functions, or a 6-round variant without the FL functions (out of the 8 rounds of the cipher).

In this paper we combine the generic impossible differential attack against 5-round Feistel ciphers with the dedicated Slicing attack to mount an attack on 5-round MISTY1 with all the FL functions with time complexity of $2^{46.45}$ simple operations. We then extend the attack to 6-round MISTY1 with the FL functions present, leading to the best known cryptanalytic result on the cipher. We also present an attack on 7-round MISTY1 without the FL layers.

1 Introduction

MISTY1 [10] is a 64-bit block cipher with presence in many cryptographic standards and applications. For example, MISTY1 was selected to be in the CRYPTREC e-government recommended ciphers in 2002 and in the final NESSIE portfolio of block ciphers, as well as an ISO standard (in 2005).

MISTY1 has a recursive Feistel structure, where the round function is in itself (very close to) a 3-round Feistel construction. To add to the security of the cipher, after every two rounds (and before the first round), an FL function is applied to each of the halves independently. The FL functions are key-dependent

⋆ The first author was supported by the France Telecome Chaire. Some of the work presented in this paper was done while the first author was staying at K.U. Leuven, Belgium and supported by the IAP Programme P6/26 BCRYPT of the Belgian State (Belgian Science Policy).

⋆⋆ The second author is supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

J. Pieprzyk (Ed.): ASIACRYPT 2008, LNCS 5350, pp. 441–454, 2008.

linear functions which play the role of whitening layers (even in the middle of the encryption).

MISTY1 has withstood extensive cryptanalytic efforts. The most successful attacks on it are an impossible differential attack on 4 rounds (when the FL layers are present) [8], an integral attack on 5 rounds (when all but the last FL layers are present) [6], and an impossible differential attack on 6 rounds (without FL layers) [9].

In this paper we show that the generic impossible differential attack against 5-round Feistel constructions [2,5] can be combined with the dedicated slicing attack [8] to yield an attack on 5-round MISTY1 with all the FL functions. The data complexity of the attack is 2^{38} chosen plaintexts, and the time complexity is $2^{46.45}$ simple operations. The main idea behind this attack is to actually attack the FL functions themselves as these functions are keyed linear transformations.

After presenting the 5-round attack, we extend it by one more round, and show that by using key schedule considerations and a delicately tailored attack algorithm, it is possible to attack 6 rounds of MISTY1 with all the FL functions present. The 6-round attack requires 2^{51} chosen plaintexts and has a running time of $2^{123.4}$ encryptions.

Finally, we present an impossible differential attack on 7-round MISTY1 when the FL layers are omitted. The attack uses $2^{50.2}$ known plaintexts, and has a running time of $2^{114.1}$ encryptions. We summarize our results along with previously known results on MISTY1 in Table 1.

Table 1. Summary of the Attacks on MISTY1

Attack	Rounds	FL functions	Complexity	
			Data	Time
Impossible Differential [7]	4	Most	2^{23} CP	$2^{90.4}$
Impossible Differential [7]	4	Most	2^{38} CP	2^{62}
Collision Search [7]	4	Most	2^{20} CP	2^{89}
Collision Search [7]	4	Most	2^{28} CP	2^{76}
Slicing Attack [8]	$4^{\dagger}$	All	$2^{22.25}$ CP	2^{45}
Slicing Attack & Impossible Differential [8]	4	All	$2^{27.2}$ CP	$2^{81.6}$
Impossible Differential [8]	4	All	$2^{27.5}$ CP	2^{116}
Integral [6]	5	Most	$2^{10.5}$ CP	$2^{22.11}$
Impossible Differential (Section 3)	$5^{\dagger}$	All	2^{38} CP	$2^{46.45}$
Impossible Differential (Section 4)	6	All	2^{51} CP	$2^{123.4}$
Higher-Order Differential [1]	5	None	$2^{10.5}$ CP	2^{17}
Impossible Differential [7]	6	None	2^{54} CP	2^{61}
Impossible Differential [7]	6	None	2^{39} CP	2^{106}
Impossible Differential [9]	6	None	2^{39} CP	2^{85}
Impossible Differential (Section 5)	7	None	$2^{50.2}$ KP	$2^{114.1}$

KP – Known plaintext, CP – Chosen plaintext.

† – the attack retrieves 41.36 bits of information about the key.

This paper is organized as follows: In Section 2 we give a brief description of the structure of MISTY1. We present our 5-round attack in Section 3, and discuss its extension to 6 rounds in Section 4. In Section 5 we present a 7-round attack which can be applied when there are no FL layers. Section 6 concludes the paper.

2 The MISTY1 Cipher

MISTY1 [10] is a 64-bit block cipher that has a key size of 128 bits. Since its introduction it withstood several cryptanalytic attacks [1,6,7,8,9], mostly due to its very strong round function (which accepts 32-bit input and 112-bit subkey[1]) and the FL layers (keyed linear transformations) which are applied every two rounds. The security of MISTY1 was acknowledged several times, when it was selected to the NESSIE portfolio, the CRYPTREC's list of recommended ciphers, and as an ISO standard.

MISTY1 has a recursive structure. The general structure of the cipher is a 8-round Feistel construction, where the round function, FO, is in itself close to a 3-round Feistel construction. The input to the FO function is divided into two halves. The left one is XORed with a subkey, enters a keyed permutation FI, and the output is XORed with the right half. After the XOR the two halves are swapped, and the same process (including the swap) is repeated two more times. After that, an additional swap and an XOR of the left half with a subkey are performed.

The FI in itself also has a Feistel-like structure. The 16-bit input is divided into two unequal halves — one of 9 bits, and the second of 7 bits. The left half (which contains 9 bits) enters an S-box, $S9$, and the output is XORed with the 7-bit half (after padding the 7-bit value with two zeroes). The two halves are swapped, the 7-bit half enters a different S-box, $S7$, and the output is XORed with 7 bits out of the 9 of the right half. The two halves are then XORed with a subkey, and swapped again. The 9-bit value again enters $S9$, and the output is XORed with the 7-bit half (after padding). The two halves are then swapped for the last time.

Every two rounds, starting before the first one, the two 32-bit halves enter an FL layer. The FL layer is a simple transformation. The input is divided into two halves of 16 bits each, the AND of the left half with a subkey is XORed to the right half, and the OR of the updated right half with another subkey is XORed to the left half. We outline the structure of MISTY1 and its parts in Figure 1.

The key schedule of MISTY1 takes the 128-bit key, and treats it as eight 16-bit words $K_1, K_2, \ldots, K_8$. From this set of subkeys, another eight 16-bit words are generated according to $K'_i = FI_{K_{i+1}}(K_i)$.[2]

[1] In [7] it was observed that the round function has an equivalent description that accepts 105 equivalent subkey bits.

[2] In case the index of the key j is greater than 8, the used key word is $j - 8$. This convention is used throughout the paper.

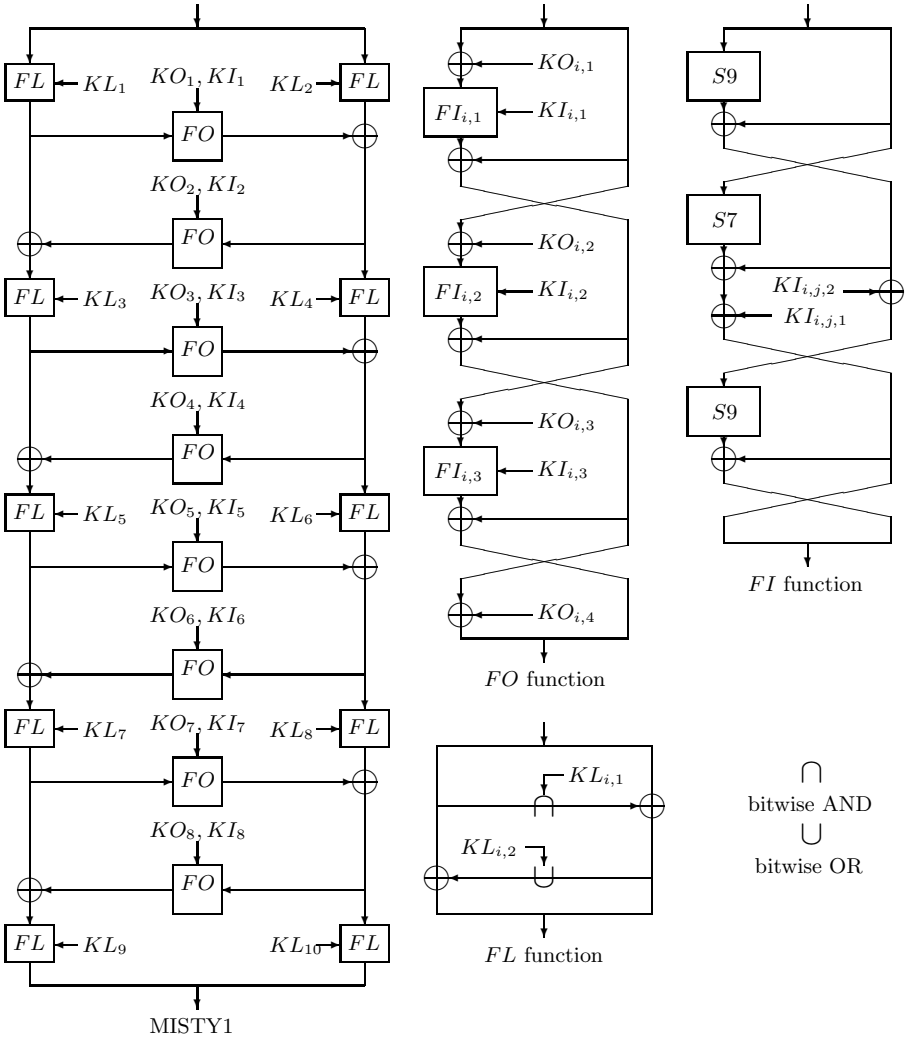

Fig. 1. Outline of MISTY1

In each round, seven words are used as the round subkey, and each of the FL functions accepts two subkey words. We give the exact key schedule of MISTY1 in Table 2.

3 An Impossible Differential Attack on 5-Round MISTY1

Our attack on 5-round MISTY1 with all the FL functions is based on the generic impossible differential attack against 5-round Feistel constructions with a bijective round function [2,5] and on the dedicated slicing attack [8] on reduced-round MISTY1.

Table 2. The Key Schedule Algorithm of MISTY1

$KO_{i,1}$	$KO_{i,2}$	$KO_{i,3}$	$KO_{i,4}$	$KI_{i,1}$	$KI_{i,2}$	$KI_{i,3}$	$KL_{i,1}$	$KL_{i,2}$
K_i	K_{i+2}	K_{i+7}	K_{i+4}	K'_{i+5}	K'_{i+1}	K'_{i+3}	$K_{\frac{i+1}{2}}$ (odd i) $K'_{\frac{i}{2}+2}$ (even i)	$K'_{\frac{i+1}{2}+6}$ (odd i) $K_{\frac{i}{2}+4}$ (even i)

3.1 The New 5-Round Impossible Differential

The generic attack on 5-round Feistel constructions is based on the following impossible differential:

Observation 1 ([2], page 136). *Let $E : \{0,1\}^{2n} \to \{0,1\}^{2n}$ be a 5-round Feistel construction with a bijective round function $f : \{0,1\}^n \to \{0,1\}^n$. Then for all non-zero $\alpha \in \{0,1\}^n$, the differential $(0,\alpha) \to (0,\alpha)$ through E is impossible.*

Our proposition is based on the fact that a similar impossible differential can be constructed even if FL layers are added to the construction, as in MISTY1. Note that since for a given key the FL layers are linear, we can define $FL(\alpha)$ for a difference α as the unique difference β such that $(x \oplus y = \alpha) \Rightarrow (FL(x) \oplus FL(y) = \beta)$.

Proposition 1. *Let E denote a 5-round variant of MISTY1, with all the FL functions present (including an FL layer after round 5). If for the given secret key we have $FL8(FL6(FL4(FL2(\alpha)))) = \beta$, where FLn is FL with the key KL_n, then the differential $(0,\alpha) \to (0,\beta)$ through E is impossible.*

Proof. If the plaintext difference is $(0,\alpha)$, then after the first FL layer, the difference becomes $(0, FL2(\alpha))$. This difference evolves after two rounds (including the second FL layer) to $(x, FL4(FL2(\alpha)))$, where $x \neq 0$ due to the bijectiveness of the round function of MISTY1.

On the other hand, if the output difference is $(0,\beta)$ such that $\beta = FL8(FL6(FL4(FL2(\alpha))))$, then before the last FL layer, the difference is $(0, FL6(FL4(FL2(\alpha))))$, and thus the input difference to round 5 is also $(0, FL6(FL4(FL2(\alpha))))$. Thus, the difference before the third FL layer is $(0, FL4(FL2(\alpha)))$.

However, if the input difference to round 3 is $(x, FL4(FL2(\alpha)))$ and the output difference of round 4 (before the FL layer) is $(0, FL4(FL2(\alpha)))$, then the output difference of the FO function in round 3 is zero. This is impossible since the input difference to this FO function is $x \neq 0$, and the FO function is bijective.

Hence, the differential $(0,\alpha) \to (0,\beta)$ is indeed impossible. □

We note that a similar approach is used in the slicing attack on 4-round MISTY1 [8]. The slicing attack is based on the generic 3-round impossible differential $(0,\alpha) \to (0,\beta)$ for all non-zero α, β which holds for every 3-round Feistel construction with a bijective round function.

3.2 The Structure of the FL Functions

A straightforward way to use the new impossible differential to attack 5-round MISTY1 is to encrypt many pairs with difference $(0, \alpha)$ for non-zero α, consider the pairs whose ciphertext difference is of the form $(0, \beta)$, and discard subkeys of the FL layers for which $FL8(FL6(FL4(FL2(\alpha)))) = \beta$. However, since the subkeys used in $FL2, FL4, FL6$, and $FL8$ are determined by 96 key bits, this approach is very time consuming. Instead, we examine the structure of the FL functions in order to find an efficient way to find the instances for which $FL8(FL6(FL4(FL2(\alpha)))) = \beta$, for a given pair (α, β). We use a series of observations, most of which were first presented in [8].

In the rest of this section, the function $FL8 \circ FL6 \circ FL4 \circ FL2$ is denoted by G.

1. For each $0 \leq i \leq 15$, the i-th bits of both halves of the input to an FL function and the i-th bits of both halves of the subkey used in the FL function, influence only the i-th bits of both halves of the output of the function. As a result, each FL function can be represented as a parallel application of 16 functions $f_i : \{0,1\}^2 \rightarrow \{0,1\}^2$ keyed by two different subkey bits each.
2. Each f_i is linear and invertible.
3. The two observations above hold also for a series of FL functions applied sequentially. In particular, the function $G = FL8 \circ FL6 \circ FL4 \circ FL2$ can be represented as a parallel application of 16 functions $g_i : \{0,1\}^2 \rightarrow \{0,1\}^2$ keyed by eight subkey bits each. The g_i's are all linear and invertible, and hence, can realize only six possible functions.[3] Thus, there are only $6^{16} = 2^{41.36}$ possible G functions.
4. Since each g_i is invertible, the differentials $0 \rightarrow a$ and $a \rightarrow 0$ through g_i are impossible, for each non-zero $a \in \{0,1\}^2$. As a result, most of the differentials of the form $\alpha \rightarrow \beta$ through G are impossible, regardless of the subkeys used in the FL functions. In each of the g_i-s, only 10 out of the 16 possible input/output pairs are possible. Hence, only $(10/16)^{16} = 2^{-10.85}$ of the input/output pairs for G are possible.
5. Assume that $G(\alpha) = \beta$, for fixed α and β. We want to find how many functions of the form G (out of the possible $2^{41.36}$ functions) satisfy this condition. For each g_i, there are 10 possible input/output pairs (the other six pairs are impossible for any subkey). For the $0 \rightarrow 0$ pair, all the six possible g_i functions satisfy this condition. For each of the 9 remaining pairs, two of the six functions satisfy the condition. Since the g_i functions are independent, the expected number of functions satisfying the conditions for all the g_i-s is:

$$\sum_{j=0}^{16} \binom{16}{j} \cdot \left(\frac{9}{10}\right)^j \cdot \left(\frac{1}{10}\right)^{16-j} \cdot 2^j \cdot 6^{16-j} = 2^{20.2}.$$

[3] Since we are interested only in differences, we treat two functions that differ by an additive constant as the same function. The total number of functions for each f_i is actually 24.

The $2^{41.36}$ possible G functions can be enumerated in such a way that the functions satisfying the condition for each (α, β) pair can be found efficiently.

Using these observations on the structure of the FL functions, we are ready to present our attack.

3.3 The New Attack

1. Ask for the encryption of 64 structures of 2^{32} plaintexts each, such that in each structure, the left half of all the plaintexts is equal to some random value A, while the right half obtains all possible values. (As a result, the difference between two plaintexts in the same structure is of the form $(0, \alpha)$).
2. For each structure, find the pairs whose output difference is of the form $(0, \beta)$.
3. For each pair with input difference $(0, \alpha)$ and output difference $(0, \beta)$ check whether $\alpha \rightarrow \beta$ is an impossible differential for the function G (as described in Section 3.2). Discard pairs which fail this test.
4. For each remaining pair, find all the G functions satisfying the condition $G(\alpha) = \beta$ and discard them from the list of all possible G functions.
5. After analyzing all the remaining pairs, output the list of remaining G functions.

Step 2 of the algorithm can be easily implemented by a hash table, resulting in about 2^{31} pairs from each structure. Step 3 can be easily performed by evaluating a simple Boolean function on the input and the output (as we are concerned with cases of a zero input causing a non-zero output or vice versa).[4]

As noted in Section 3.2, out of the 2^{31} pairs, about $2^{31} \cdot 2^{-10.85} = 2^{20.15}$ pairs remain from each structure at this point. Each of these pairs discards about $2^{20.2}$ possible values of G on average (as shown in Section 3.2), and thus, each structure is expected to discard about $2^{40.35}$ G functions. The identification of the discarded functions can be performed very efficiently.

Thus, after analyzing about 64 structures, we are left only with the right G function.[5] The time complexity of the attack is about $64 \cdot 2^{20.15} \cdot 2^{20.2} = 2^{46.35}$ simple operations, and the information retrieved by the attacker is equivalent to 41.36 key bits. In many situations, this is considered a break of the system and the attack terminates.

[4] The exact Boolean expression is as follows: Let the input difference of G be (x_1, x_2) and the output difference of G be (y_1, y_2). Also let $\overline{t}$ be the bitwise NOT of t, let & be a bitwise AND, and | be a bitwise OR. If $\overline{x_1}\&\overline{x_2}\&(y_1|y_2)$ is non-zero then there is a zero input difference transformed to a non-zero output difference. It is also required to check whether the output difference is zero and the input difference is non-zero, which is done by evaluating: $\overline{y_1}\&\overline{y_2}\&(x_1|x_2)$.

[5] We expect $2^{40.35} \cdot 64 = 2^{46.35}$ functions to be discarded (with overlap). Thus, the probability that a specific function remains after the analysis is

$$\left(1 - 2^{-41.36}\right)^{2^{46.35}} \approx e^{-32} = 2^{-46.2}.$$

3.4 Retrieving the Rest of the Secret Key

If the attacker wants to retrieve the actual value of the key, she can use the G function found in the attack to retrieve the value of the subkeys used in the G function. A naive approach is to try the possible 2^{96} subkeys which affect the functions $FL2, FL4, FL6$, and $FL8$, and check (for each subkey) whether it yields the correct G function. A more efficient algorithm is to guess the values of the subkeys K'_3, K_4, K_5, K_6, and K_7, and check whether they induce the correct transformation from the input of G to the right half of the output of G. If this is the case, the attacker can retrieve the suggested value for K_8 efficiently, and if the suggestion is consistent with the correct G function, the attacker obtains a candidate for 96 bits of the key (the knowledge of K'_3 and K_4 allows computing K_3). The time complexity of this approach is roughly 2^{80} evaluations of four FL functions, and the attacker gets a list of $2^{96} \cdot 2^{-41.36} = 2^{54.64}$ 96-bit subkeys. Retrieving the rest of the key by exhaustive search leads to a total time complexity of $2^{86.64}$ encryptions.

We note that possibly this part of the attack can be performed much more efficiently using some different attack technique and exploiting the key information obtained so far.[6]

4 Extending the Attack to 6 Rounds

The simplest way to extend a 5-round attack to 6 rounds is to guess the subkey of the last round, peel the last round off, and apply the 5-round attack. In MISTY1, this requires guessing the key of the last FL layer, as well as 112 subkey bits which enter the sixth FO function. Thus, we need to use a more careful analysis and key schedule considerations to present this attack.

In our attack we guess the subkey of the last FL layer (composed of 64 bits), and examine only ciphertext pairs with a special structure in order to reduce the amount of subkey material in the sixth FO we need to handle. Finally, we repeat the five round attack, taking into consideration the already known subkey material.

The special structure of the pairs examined in the attack is based on the following observation, presented in [7]:

Observation 2. *([7]) Assume that the input values to the function FO_i are known. The question whether the output difference of FO_i is of the form (δ, δ), for a 16-bit value δ, depends only on the 50 subkey bits $KO_{i,1}, KO_{i,2}, KI_{i,1,2}$, and $KI_{i,2,2}$.*

4.1 The Attack's Algorithm

1. Take m structures (generated just like in the 5-round attack).
2. For each guess of the subkey used in the last FL layer (subkeys K'_2, K_4, K'_6, and K_8), partially decrypt all the ciphertexts.

[6] We note that a similar problem is discussed in [8], and several techniques applicable in special cases (e.g., if the attacker can use both chosen plaintext and chosen ciphertext queries) are presented.

3. Find all pairs with plaintext difference $(0, \alpha)$ and ciphertext difference $((\delta, \delta), (x, y))$,[7] such that differential $\alpha \rightarrow (x, y)$ through $FL6 \circ FL4 \circ FL2$ is not impossible (see Section 3.2).
4. **Analysis of Round 6:** For each such pair, with difference $((\delta, \delta), (x, y))$, perform the following steps:
   (a) Given $KO_{6,2} = K_8$ compute the actual values just before the key addition with $KI_{6,2}$ for the pair. If the difference in the 7 left bits does not fit the corresponding 7 difference bits of δ — discard the pair.
   (b) Using the input and output differences of the second $S9$ S-box of the function $FI_{6,2}$, find the pairs of actual input values satisfying this difference relation.[8] From the actual input values obtain (on average) one candidate for the 9 bits of $KI_{6,2,2}$.
   (c) For each possible guess of $KI_{6,2,1}$ (i.e., the remaining unknown bits of K'_7) compute $KO_{6,1} = K_6$, and check whether the difference in the 7 left bits before the key addition in the first FI is equal to the difference in the 7 left bits of y.
   (d) Similarly to Step (4b), deduce $KI_{6,1,2}$ using the input/output differences of the second $S9$ in the function $FI_{6,1}$, suggested by the pair.
5. **Application of the 5-Round Attack:** For each guess of the 89 subkey bits (i.e., $K'_2, K_4, K'_6, K'_7, K_8, KI_{6,1,2}$) and for each pair corresponding to this subkey guess, perform the following:
   (a) Guess the 9 least significant bits of K_5 and use the key schedule to compute bits $7, 8$ of K'_4 and K'_5. Check whether the relation $FL6(FL4(FL2(\alpha))) = (x, y)$ holds at bits $7, 8$ of the left and the right halves of α and β (note that all the subkey bits involved in this relation are already known). If no, discard the pair.
   (b) Guess the remainder of K_5, and compute the full values of K'_4 and K'_5. Check whether the pair can achieve $\alpha \rightarrow (x, y)$, and retrieve the suggested value for the 7 remaining bits of K'_3.
   (c) If at this stage, for a given key guess there are remaining pairs, discard the subkey guess (as it suggests an impossible event). Otherwise, retrieve the remaining key bits by exhaustive search.

4.2 Analysis of the Attack

Starting with m structures, for each guess of the subkey used in the last FL layer (64 bits), about $m \cdot 2^{63} \cdot 2^{-16} \cdot 2^{-10.85} = m \cdot 2^{36.15}$ pairs are expected to enter Step (4). Each of these pairs has probability 2^{-7} to satisfy the differential condition of Step (4a), leaving $m \cdot 2^{29.15}$ pairs for each guess of the first 64 subkey bits. Then, in Step (4b) we obtain (for each pair) one candidate on average for 9 additional subkey bits, reducing the number of pairs associated with a given subkey guess (of 73 bits) to $m \cdot 2^{20.15}$ pairs. These two operations (a 7-bit filtering

[7] The reader is advised that we give the values without the swap operation, to be consistent with our figure describing MISTY1.

[8] This can be done easily by examining the difference distribution table of $S9$.

and a 9-bit subkey suggestion) are performed again in Steps (4c,4d) for each guess of 7 additional subkey bits. As a result, $m \cdot 2^{20.15} \cdot 2^{-16} = m \cdot 2^{4.15}$ pairs are expected to enter Step (5), for each of the 89-bit subkey guesses.

In Step (5), we guess a total of 16 additional key bits, and discard all the pairs for which $FL6(FL4(FL2(\alpha))) \neq (x, y)$. Since all the pairs for which the differential $\alpha \rightarrow (x, y)$ through $FL6 \circ FL4 \circ FL2$ is impossible were discarded in Step (3) of the attack, the probability of a pair to pass the filtering of Step (5) is $2^{-21.15}$. Hence, the number of pairs remaining after Step (5) for each subkey guess is $m \cdot 2^{4.15} \cdot 2^{-21.15} = m \cdot 2^{-17}$. As a result, the probability that a subkey guess is not discarded is $e^{-m \cdot 2^{-17}}$. Thus, the time complexity of Step (5c) is $2^{128} \cdot e^{-m \cdot 2^{-17}}$ encryptions.

We note that the number of pairs entering Step (5b) is $m \cdot 2^{1.5}$ for each subkey guess. Indeed, in Step (5a) we discard the pairs for which $FL6(FL4(FL2(\alpha))) \neq (x, y)$ in four bits. It may seem that the probability of a pair to pass this filtering is 2^{-4}. However, since the pairs for which the differential $\alpha \rightarrow (x, y)$ through $FL6 \circ FL4 \circ FL2$ is impossible were already discarded before, the probability of a pair to pass the filtering[9] is $2^{-2.65}$, and hence the number of remaining pairs is indeed $m \cdot 2^{1.5}$ for each subkey guess.

The two most time consuming steps of the attack are Steps (5b) and (5c). Step (5b) takes $3 \cdot m \cdot 2^{1.5} \cdot 2^{105} = m \cdot 2^{108.1}$ evaluations of FL. We take the moderate assumption that the time complexity of three FL evaluations is not greater than 1/8 of the time required for a 6-round encryption. Hence, the time complexity of Step(5b) is about $m \cdot 2^{103.5}$ MISTY1 encryptions. Step (5c) takes $2^{128} \cdot e^{-m \cdot 2^{-17}}$ encryptions.

The least overall time complexity is achieved when both terms are the same, i.e., when $m \cdot 2^{103.5} = 2^{128} \cdot e^{-m \cdot 2^{-17}}$. Solving this equation numerically, suggests that $m = 2^{18.945}$ is the optimal value. Thus, the data complexity of the attack is $m \cdot 2^{32} \approx 2^{51}$ chosen plaintexts, and the time complexity is $2^{123.4}$ encryptions.

5 Attack on 7-Round MISTY1 with No FL Layers

In this section we show that if the FL layers are removed from the structure of MISTY1, then the generic impossible differential for 5-round Feistel constructions [2,5] can be used to mount an attack on a 7-round variant of the cipher. The attack is based on examining pairs with input difference (α, x) and output difference (α, y), and discarding all the subkeys which lead to the impossible differential $(\alpha, 0) \rightarrow (\alpha, 0)$ in rounds 2–6. However, since each of the FO functions uses 112 key bits, trying all the possible subkeys is infeasible. Instead, we use differential properties of the FO function, along with key schedule considerations, in order to discard the possible subkeys efficiently.

[9] As noted earlier, in the filtering in Step 3, the attacker discards (for a given pair of bits) 6 out of 16 possible values. Hence, in this step, the attacker discards 9 out of the remaining 10 values.

5.1 Differential Properties of the *FO* Function

We start with an observation presented in [7].

Observation 3 ([7]). *Given a pair of input values to the function FO_i, the corresponding output difference depends only on the equivalent of* 75 *subkey bits. These bits are the subkeys $KO_{i,1}, KO_{i,2}, KI_{i,1,2}, KI_{i,2,2}$, and $KI_{i,3,2}$, and the equivalent subkey*

$$AKO_{i,3} = KO_{i,3} \oplus KI_{i,1,1}||00||KI_{i,1,1},$$

where || denotes concatenation.

We refer the reader to [7] for the complete proof of this observation.

Our next proposition is a novel observation concerning MISTY1:

Proposition 2. *Assume that the input values and the output difference of the function FO_i are known, along with one of the following sets of subkey bits:*

1. $KO_{i,1}, KI_{i,1,2}, KI_{i,2,2}, KI_{i,3,2}$, *or*
2. $KO_{i,2}, KI_{i,1,2}, KI_{i,2,2}, KI_{i,3,2}$.

Consider the remaining 32 key bits that influence the output difference (i.e., $KO_{i,2}$ or $KO_{i,1}$, respectively, along with $AKO_{i,3}$). There exists one value of these 32 bits on average which satisfies the input/output condition, and this value can be found efficiently (using only several simple operations).

Proof. Consider the case when the bits of Set (1) are known. The knowledge of bits $KO_{i,1}$ and $KI_{i,1,2}$ allows to encrypt the pair through the first FI layer and (using the output difference of FO_i) obtain the output difference of $FI_{i,2}$. The input difference to $FI_{i,2}$ can be computed from the input of FO_i. Given the input and the output differences to the function $FI_{i,2}$ and the subkey $KI_{i,2,2}$, there exists one pair of inputs on average which satisfies the input/output difference condition. This pair of actual values, along with the input to FO_i, suggests a unique value for the subkey $KO_{i,2}$. Similarly, since the input and output differences to $FI_{i,3}$ and the subkey $KI_{i,3,2}$ are known, they suggest one value of the subkey $AKO_{i,3}$ on average which satisfies these differences.

In the second case, when the bits of Set (2) are known, the knowledge of bits $KO_{i,2}$ and $KI_{i,2,2}$ allows to encrypt the pair through the second FI layer and (using the output difference of FO_i) obtain the output difference of $FI_{i,1}$. The input difference to $FI_{i,1}$ can be computed from the input of FO_i. This input/output difference pair suggests a single value of the subkey $KO_{i,1}$ on average. The single suggestion for $AKO_{i,3}$ can be retrieved as in the first case.

In order to obtain the suggested subkeys efficiently, it is sufficient to precompute the full difference distribution table [4] of the FI function (i.e., a table containing also the actual values which satisfy each input/output difference condition), for each possible value of $KI_{i,j,2}$. Each such table requires about 2^{34} bytes of memory. In the on-line phase of the attack, given the input/output

differences to an FI function, along with the corresponding subkey $KI_{i,j,2}$, the possible actual values of the input can be found using a single table look-up. Hence, the suggested values for the 32 subkey bits can be found using only several simple operations. □

Now we are ready to present the attack.

5.2 The Attack Algorithm

The attack algorithm is as follows:

1. Ask for the encryption of $2^{50.2}$ known plaintexts.
2. Find all pairs (P_1, P_2) and their corresponding ciphertexts (C_1, C_2), respectively, such that $P_1 \oplus P_2 = (\alpha, x)$ and $C_1 \oplus C_2 = (\alpha, y)$ for some x, y and α. The expected number of pairs remaining after this stage is $(2^{50.2})^2/2 \cdot 2^{-32} = 2^{67.4}$.
3. **Examining round 1:** For each of the remaining pairs, perform the following:
   (a) Guess the subkey K_1 and the 9 least significant bits of the subkeys K'_2, K'_4, K'_6 (which compose the subkeys $KO_{1,1}, KI_{1,1,2}, KI_{1,2,2}$, and $KI_{1,3,2}$). Use Proposition 2 to find the suggested value for the subkeys $KO_{1,2} = K_3$ and $AKO_{1,3}$.
   (b) Guess the remaining bits of K'_6 (which are the bits of $KI_{1,1,1}$), and use the value of $AKO_{1,3}$ to obtain the value $KO_{1,3} = K_8$.
   (c) For each value of the subkeys K_3 and K_8, store the list of all the pairs which suggested this value. The expected number of such pairs is $2^{67.4} \cdot 2^{16+9+9+9} \cdot 2^7/2^{82} = 2^{35.4}$.
4. **Examining round 7:** For each possible value of the 82 bits of the key considered in Step 3 (subkeys K_1, K_3, K'_6, K_8, and the 9 least significant bits of K'_2 and K'_4), and for each of the pairs corresponding to each subkey value, perform the following:
   (a) Use Proposition 2 to find the values $KO_{7,1} = K_7$ and $AKO_{7,3}$ (note that the values $KO_{7,2}, KI_{7,1,2}, KI_{7,2,2}$, and $KI_{7,3,2}$ are known at this stage).
   (b) Use the key schedule to find the value of K_6. Use the knowledge of $AKO_{7,3}$ and $KO_{7,3} = K_6$ to get the value of $KI_{7,1,1}$, along with a 9-bit filtering condition (only pairs for which $AKO_{7,3} \oplus KO_{7,3}$ is of the form $a||00||a$, for some 7-bit value a, remain, and suggest the value $KI_{7,1,1} = a$).
5. Discard the values of the 105 examined key bits ($K_1, K_3, K'_4, K_6, K_7, K_8$, and the 9 least significant bits of K'_2) suggested by at least one pair. The expected number of pairs suggesting each subkey value is $2^{35.4} \cdot 2^{-9}/2^{23} = 10.56$. As the number of pairs suggesting a subkey value has a Poisson distribution, a subkey remains (i.e., is not suggested by any pairs) with probability $e^{-10.56} = 2^{-15.23}$. Hence, the expected number of remaining 105-bit subkeys is $2^{105} \cdot 2^{-15.23} = 2^{89.77}$.

6. For the remaining possibilities of the 105-bit subkey, exhaustively search all possible keys, until the right key is found.

The data complexity of the attack is $2^{50.2}$ known plaintexts. Its time complexity is mostly dominated by Step (4) and Step (6). Step (4) is repeated $2^{35.4} \cdot 2^{82} = 2^{117.4}$ times. Each such key deduction is expected to take one FI application, two memory accesses, and a few XOR operations. For sake of simplicity we assume that this is equal to 1/16 of 7-round MISTY1 encryption, and thus, Step (4) takes a total of $2^{113.4}$ encryptions. Step (6) takes $2^{128} \cdot 2^{-15.23} = 2^{112.8}$ trial encryptions. Therefore, the total time complexity of the attack is $2^{114.1}$ encryptions.

6 Summary and Conclusions

In this paper we presented several new impossible differential attacks on MISTY1. While previous attacks were applicable only up to 4 rounds of the cipher (including the FL layers), we presented a 5-round attack with time complexity of $2^{46.45}$ simple operations, and extended it to an attack on a 6-round variant faster than exhaustive key search. We also presented a 7-round attack on a variant of the cipher without FL functions. The best previously known attacks against this variant were on 6 rounds.

It seems interesting to compare between the attacks on reduced-round variants of MISTY1 including the FL functions, and the attacks on the variant without the FL functions. If the FL functions do not exist, much simpler impossible differential attacks can be mounted, and as a result, the attacks extend to one more round, compared to the case where the FL-s are present. On the other hand, when the FL functions are present, their linear structure can be exploited in order to reduce significantly the time complexity of impossible differential attacks.

Thus, we conclude that while the FL functions do contribute to the security of the full MISTY1 with respect to impossible differential attacks, they may reduce the practical security of reduced variants with a relatively small number of rounds.[10]

References

1. Babbage, S., Frisch, L.: On MISTY1 higher order differential cryptanalysis. In: Won, D. (ed.) ICISC 2000. LNCS, vol. 2015, pp. 22–36. Springer, Heidelberg (2001)
2. Biham, E., Biryukov, A., Shamir, A.: Miss in the Middle Attacks on IDEA and Khufu. In: Knudsen, L.R. (ed.) FSE 1999. LNCS, vol. 1636, pp. 124–138. Springer, Heidelberg (1999)

[10] It seems that the extremely low time complexity of the impossible differential attack on 5-round MISTY1 with the FL layers cannot be achieved if the FL layers are absent (even for a 5-round variant), due to the big amount of subkey bits affecting each FO function. As a result, the practical security of 5-round MISTY1 w.r.t. impossible differential attacks is *reduced* if the FL layers are present. A similar observation regarding a 4-round variant of MISTY1 was made in [8].

3. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack Reduced to 31 Rounds. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23. Springer, Heidelberg (1999)
4. Biham, E., Shamir, A.: Differential Cryptanalysis of the Data Encryption Standard. Springer, Heidelberg (1993)
5. Knudsen, L.R.: The Security of Feistel Ciphers with Six Rounds or Less. Journal of Cryptology 15(3), 207–222 (2002)
6. Knudsen, L.R., Wagner, D.: Integral Cryptanalysis. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 112–127. Springer, Heidelberg (2002)
7. Kühn, U.: Cryptanalysis of Reduced-Round MISTY. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 325–339. Springer, Heidelberg (2001)
8. Kühn, U.: Improved cryptanalysis of MISTY1. In: Daemen, J., Rijmen, V. (eds.) FSE 2002. LNCS, vol. 2365, pp. 61–75. Springer, Heidelberg (2002)
9. Lu, J., Kim, J., Keller, N., Dunkelman, O.: Improving the Efficiency of Impossible Differential Cryptanalysis of Reduced Camellia and MISTY1. In: Malkin, T.G. (ed.) CT-RSA 2008, vol. 4964, pp. 370–386. Springer, Heidelberg (2008)
10. Matsui, M.: Block encryption algorithm MISTY. In: Biham, E. (ed.) FSE 1997. LNCS, vol. 1267, pp. 64–74. Springer, Heidelberg (1997)
11. Tanaka, H., Hisamatsu, K., Kaneko, T.: Strength of MISTY1 without FL function for higher order differential attack. In: Fossorier, M.P.C., Imai, H., Lin, S., Poli, A. (eds.) AAECC 1999. LNCS, vol. 1719, pp. 221–230. Springer, Heidelberg (1999)