Impossible Differential Cryptanalysis of Reduced-Round Tweakable TWINE

Mohamed Tolba, Muhammad ElSheikh, Amr M. Youssef

Concordia Institute for Information Systems Engineering,
Concordia University, Montréal, Québec, Canada

Abstract. Tweakable TWINE (T-TWINE) is a new lightweight tweakable block cipher family proposed by Sakamoto *et al*. at IWSEC 2019. T-TWINE is the first Tweakable Block Cipher (TBC) that is built on Generalized Feistel Structure (GFS). It is based on the TWINE block cipher in addition to a simple tweak scheduling based on SKINNY's tweakey schedule. Similar to TWINE, it has two versions, namely, T-TWINE-80 and T-TWINE-128, both have a block length of 64 bits and employ keys of length 80 and 128 bits, respectively. In this paper, we present impossible differential attacks against reduced-round versions of T-TWINE-80 and T-TWINE-128. First, we present an 18-round impossible differential distinguisher against T-TWINE. Then, using this distinguisher, we attack 25 and 27 rounds of T-TWINE-80 and T-TWINE-128, respectively.

Keywords: Cryptanalysis, Impossible differential attacks, Tweakable, Block ciphers, TWINE, T-TWINE.

1 Introduction

Tweakable Block Ciphers (TBCs) [12] differ from the conventional block ciphers since they accept an additional input called a tweak. Different specific keyed instances of the cipher can be generated by varying this tweak. TBCs allow new interesting highly-secure modes of operation and applications to become possible as they are designed to allow changing the tweak very efficiently compared to the key setup operation.

Block ciphers can be used to build TBCs through modes of operation such as LRW (Liskov, Rivest, and Wagner) and XEX (Xor–Encrypt–Xor) [15]. These modes of operations, for one TBC encryption/decryption, require few cipher calls. Therefore, they are efficient. However, their provable security guarantee, which is $2^{n/2}$ for n-bit block cipher, is not enough, in particular, for TBCs employed in modes of operation aiming to achieve "beyond-the-birthdaybound" (BBB) security. As a result, less efficient modes of operations [10,11], compared to LRW and XEX, are proposed to achieve BBB security guarantee.

Dedicated constructions is another approach to build efficient TBCs with an acceptable level of security guarantee. HPC [17], one of the submission to the AES competition, is the first proposal, where the tweak is called “spice”. Threefish [5], Deoxys-BC [8], SKINNY [2] and QARMA [1] are examples of recently proposed dedicated TBCs. Challenges such as designing efficient dedicated TBCs while having sufficient security guarantee is solved by the Tweakey flamework [7] which is based on a Substitution Permutation Network (SPN).

Tweakable TWINE (T-TWINE) [16] is the first dedicated TBC that is based on Generalized Feistel Structure (GFS) [14,23]. The only work on GFS-based TBC, before the T-TWINE proposal, is done by Goldenberg *et al.* [6] and Mitsuda and Iwata [13] who focused on studding the provable security of the round functions that are instantiated by PRFs. TWINE, which is a GFS-based block cipher, was proposed by Suzaki *et al.* [21] after a comprehensive study done by Suzaki and Minematsu [18] showing the effect of the choice of sub-block permutation on the diffusion, the number of differential/linear active S-boxes, and the maximum numbers of rounds for impossible differential characteristics and saturation characteristics. The choice of the permutation of TWINE was a result of the work done in [18], it permutes over 16 nibbles to achieve the best characteristics.

T-TWINE [16] is built with the goal of reducing the cost of design, security evaluation, and implementation. As a result, TWINE was selected to be the basic building block of T-TWINE with extremely simple tweak scheduling. This tweak schedule is based on the SKINNY’s [3] tweakey schedule. Similar to TWINE, T-TWINE has a block size of 64 bits and iterates using either 80-bit or 128-bit key over 36 rounds. It accepts an additional 64-bit tweak. It also uses independent key and tweak schedules where the tweak is mixed with the states by adding few nibble XORs to TWINE. Therefore, it has the same hardware cost of TWINE except for the additional tweak registers.

The designers of T-TWINE evaluated its security against differential, linear, impossible differential, and integral attacks in the chosen-tweak setting. However, they only presented distinguishers without converting any distinguisher to a key recovery attack. For impossible differential, they utilized the miss-in-the-middle approach to search the impossible differential characteristics that have one active nibble in the 16 tweak nibbles and one active nibble in 16 ciphertext nibbles at the decryption side. However, the 18-round impossible differential distinguisher that was proposed by the designers does not seem to be correct as we will illustrate in Section 3.

In this paper, we start by presenting the first (correct) 18-round impossible differential distinguisher. Then, we use this distinguisher to launch a 25-round attack against T-TWINE-80 by pre-appending and appending 4 and 3 rounds, respectively. Finally, we launched a 27-round attack against T-TWINE-128, using the 18-round distinguisher, by pre-appending and appending 6 and 3 rounds, respectively. The data, time, and memory complexities of the 25-round (27-round) against T-TWINE-80 (T-TWINE-128) are $2^{61.5}$ (2^{60}) chosen plaintexts, $2^{70.86}$ 25-round ($2^{120.83}$ 27-round) encryptions, 2^{66} (2^{118}) 64-bit block, respectively.

The rest of the paper is organized as follows. Section 2 provides the notations used throughout the paper and a brief description of T-TWINE. In section 3, we present the impossible differential distinguisher used in our attacks. The details of our attacks are presented in sections 4 and 5. Finally, the paper is concluded in section 6.

2 Specifications of T-TWINE

The following notation will be used throughout the rest of the paper:

- K: The 80 or 128 bits master key.
- K_j: The j^{th} nibble of K. The indices of the nibbles begin from 0.
- RK^i: The 32-bit round key used in round $i+1$.
- RK^i_j: The j^{th} nibble of RK^i. The indices of the nibbles begin from 0.
- T: The 64-bit tweak.
- T_i: The i^{th} nibble of the tweak T.
- RT^i: The 24-bit round tweak used in round $i+1$, where $RT^i \leftarrow t^i_0||t^i_1||t^i_2||t^i_3||t^i_4||t^i_5$, and t^i_j is the j^{th} nibble of RT^i.
- X^i: The 16 4-bit nibbles output of round i.
- ${X_j}^i$: j^{th} nibble of X^i.
- $\Delta X^i, \Delta X^i_j$: The difference at state X^i and nibble X^i_j, respectively.
- $\oplus$: The XOR operation.
- $||$: The concatenation operation.
- $Rotz(x)$: The z-bit left cyclic shift of x.

T-TWINE is based on TWINE [21]. T-TWINE-80/128 iterates 36 rounds over 64-bit block using 80/128-bit key, respectively, and 64-bit tweak T. The block cipher has three parts: data processing, key schedule, and tweak schedule. Except for the tweaks addition, T-TWINE-80/128 has the same data processing and key schedule of TWINE-80/128, respectively. Both T-TWINE-80 and T-TWINE-128 deploy the same generalized Feistel structure and tweak schedule where the only difference between them is the key schedule.

Data Processing Part. As depicted in Fig. 1, the round function is based on a variant of Type-2 GFS with 16 4-bit nibbles [18]. It has four operations: 4-bit S-box (S, see Table 1), round key XOR, round tweak XOR, and a 16-nibble shuffle operation (π, see Table 2). Both versions of T-TWINE have the same number of rounds (36). The nibble shuffle operation in the last round is omitted.

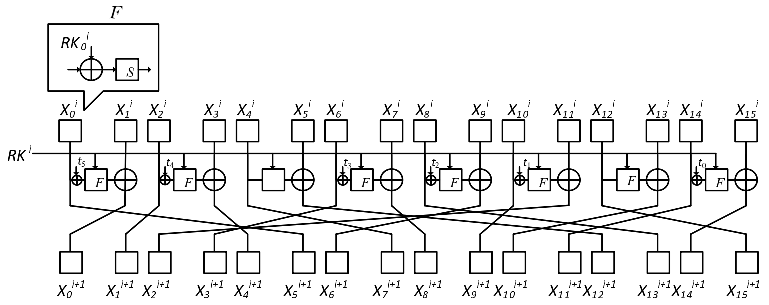

Fig. 1. The T-TWINE round function, for simplicity we use t_j instead of t_j^i. For example t_0 equivalent to t_0^i

Table 1. 4-bit S-box S in hexadecimal form

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	c	0	f	a	2	b	9	5	8	3	d	7	1	e	6	4

Key Schedule. A round key RK^i of 8 nibbles is generated from the master key K for each round i, where $0 \leq i < 35$. Each version of T-TWINE has its own key schedule. Algorithm 1 and 2 show the details of T-TWINE-80/128, respectively, where CON_H^i and CON_L^i are predefined constants. For further details, the reader is referred to [20,16].

Tweak Schedule. A round tweak RT^i of 6 nibbles is generated from the tweak T for each round i, where $0 \leq i < 35$. Both versions of T-TWINE have the same tweak schedule, shown in Algorithm 3, where π^t is a 6-nibble permutation s.t. $(0, 1, 2, 3, 4, 5) \rightarrow (1, 0, 4, 2, 3, 5)$.

3 An Impossible Differential Distinguisher of T-TWINE

Impossible differential cryptanalysis was proposed independently by Knudsen [9] and Biham, Biryukov and Shamir [4]. It exploits a (truncated) differential characteristic of probability exactly 0 and thus acts as a distinguisher. Then, this distinguisher is turned into a key-recovery attack by prepending and/or appending additional rounds, which are usually referred to as the analysis rounds. The keys involved in the analysis rounds which lead to the impossible differential are wrong keys and thus are excluded. Miss-in-the-Middle is the general technique used to construct impossible differentials, where the cipher, E, is split such that $E = E_2 \circ E_1$, and we try to find two deterministic differentials, the first one covers E_1 and has the form $\Delta\delta \rightarrow \Delta\gamma$, and the second covers E_2^{-1}, and has the form $\Delta\beta \rightarrow \Delta\zeta$. When the intermediate differences $\Delta\gamma, \Delta\zeta$ do not match, the differential $\Delta\delta \rightarrow \Delta\beta$ that covers the whole cipher E holds with zero probability.

Table 2. Nibble shuffle π

h	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$\pi[h]$	5	0	1	4	7	12	3	8	13	6	9	2	15	10	11	14
$\pi^{-1}[h]$	1	2	11	6	3	0	9	4	7	10	13	14	5	8	15	12

Algorithm 1: Key Schedule of T-TWINE-80

```
Data: The 80-bit master key $K$
Result: The round keys $RK = RK^0||RK^1||\cdots||RK^{35}$
$k_0||k_1||\cdots||k_{19} \leftarrow K$;
for $i \leftarrow 0$ to $34$ do
    $RK^i \leftarrow k_1||k_3||k_4||k_6||k_{13}||k_{14}||k_{15}||k_{16}$;
    $k_1 \leftarrow k_1 \oplus S(k_0)$;
    $k_4 \leftarrow k_4 \oplus S(k_{16})$;
    $k_7 \leftarrow k_7 \oplus (0||CON_H^i)$;
    $k_{19} \leftarrow k_{19} \oplus (0||CON_L^i)$;
    $k_0||\cdots||k_3 \leftarrow Rot4(k_0||\cdots||k_3)$;
    $k_0||\cdots||k_{19} \leftarrow Rot16(k_0||\cdots||k_{19})$;
$RK^{35} \leftarrow k_1||k_3||k_4||k_6||k_{13}||k_{14}||k_{15}||k_{16}$;
$RK \leftarrow RK^0||RK^1||\cdots||RK^{35}$;
```

The designers of T-TWINE in [16] presented an 18-round impossible differential distinguisher. They found this distinguisher using the Miss-in-the-Middle approach. The distinguisher begins at “1R” with zero differences and the tweak has a non-zero difference at the first nibble t_0. As mentioned above, this 18-round impossible differential distinguisher does not seem to be correct. In what follows, we list some of the problems (mistakes) we identified in this distinguisher (See Fig. 5): i) the numbers of rounds involved in the distinguisher is only 17 not 18 (as the plaintext is marked “1R” and the ciphertext is marked “18R”), ii) the tweaks used in the distinguisher are wrong. For example, the tweaks that are used in the seventh and ninth rounds are actually the tweaks of the sixth and seventh rounds, respectively, and iii) this distinguisher assumes that the tweak has difference at nibble “0” at the first round, then it appear again at nibble “0” at the nineteenth round, while it should appear again at the seventeenth round, after 16 rounds of the tweak schedule. Moreover, as shown in Figure 8 of [16] (See Fig. 5), the zero difference at “1R” gives, after being propagated 7 rounds in the forward direction, the difference at “8R” in the form of $(1, 1, 1, 0, 0, ?, 0, 1, 1, ?, 0, 1, ?, ?, ?, ?)$. However, the correct difference should be in the form of $(?, 1, ?, 0, 1, ?, ?, 1, 1, ?, ?, 1, ?, ?, ?, ?)$.

In this section, we present an 18-round distinguisher that begins and ends with zero difference and has a difference at t_{12} at the first round, see Fig. 2. To the best of our knowledge, this is the first valid 18-round impossible differential distinguisher. This distinguisher is found using the Miss-in-the-Middle approach, where we propagate the difference in the tweak forward 8 rounds with probability

```
Algorithm 2: Key Schedule of T-TWINE-128
  Data: The 128-bit master key $K$
  Result: The round keys $RK = RK^0||RK^1||\cdots||RK^{35}$
  $k_0||k_1||\cdots||k_{31} \leftarrow K$;
  for $i \leftarrow 0$ to 34 do
      $RK^i \leftarrow k_2||k_3||k_{12}||k_{15}||k_{17}||k_{18}||k_{28}||k_{31}$;
      $k_1 \leftarrow k_1 \oplus S(k_0)$;
      $k_4 \leftarrow k_4 \oplus S(k_{16})$;
      $k_{23} \leftarrow k_{23} \oplus S(k_{30})$;
      $k_7 \leftarrow k_7 \oplus (0||CON_H^i)$;
      $k_{19} \leftarrow k_{19} \oplus (0||CON_L^i)$;
      $k_0||\cdots||k_3 \leftarrow Rot4(k_0||\cdots||k_3)$;
      $k_0||\cdots||k_{31} \leftarrow Rot16(k_0||\cdots||k_{31})$;
  $RK^{35} \leftarrow k_2||k_3||k_{12}||k_{15}||k_{17}||k_{18}||k_{28}||k_{31}$;
  $RK \leftarrow RK^0||RK^1||\cdots||RK^{35}$;
```

```
Algorithm 3: Tweak Schedule of T-TWINE
  Data: The 64-bit tweak $T$
  Result: The round tweaks $RT = RT^0||RT^1||\cdots||RT^{35}$
  $t_0^0||t_1^0||\cdots||t_{16}^0 \leftarrow T$;
  for $i \leftarrow 0$ to 35 do
      $RT^i \leftarrow t_0^i||t_1^i||t_2^i||t_3^i||t_4^i||t_5^i$;
      for $h \leftarrow 0$ to 5 do
          $t^i_{\pi^t[h]} \leftarrow t^i_h$;
      for $h \leftarrow 0$ to 15 do
          $t^{i+1}_{(h-6)\ mod\ 16} \leftarrow t^i_h$;
  $RT \leftarrow RT^0||RT^1||\cdots||RT^{35}$;
```

1 and propagate the difference in the tweak backward 10 rounds with probability 1, then match at the middle at the end of round 8. As seen in Fig. 2, there is a contradiction at nibble "6", where in the forward path, it should have a zero difference, while in the backward path, it should have a non-zero difference.

3.1 Observations

In this section, we present some useful observations that will be utilized in our attack.

Observation 1 [19,22] *For any input difference $a(\neq 0)$ and output difference $b(\in S[a])$ of the S-box in TWINE, the average number of pairs that satisfy the differential characteristic $(a \rightarrow b)$ is $\frac{16}{7}$. Given an 8-bit pair (X^i_{2j}, X^i_{2j+1}) and $(X^i_{2j} \oplus a, X^i_{2j+1} \oplus b)$, the probability that RK^i_j leads to the S-box differential characteristic $(a \rightarrow b)$ is 7^{-1}.*

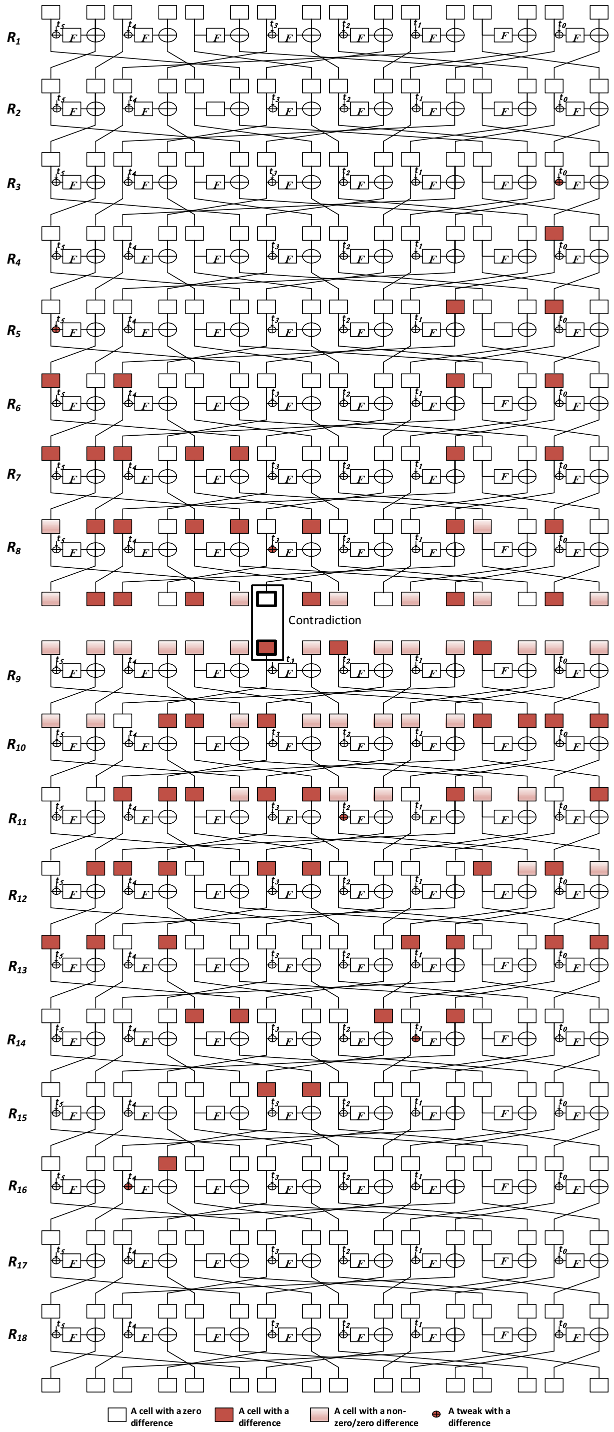

Fig. 2. An 18-round impossible differential distinguisher

Observation 2 *Given two nonzero differences Δi and Δo in $\mathbb{F}16$, the equation: $S(x) + S(x + \Delta i) = \Delta o$ has one solution on average. This property also applies to S^{-1}.*

Observation 3 *If the impossible differential illustrated in Fig. 2 is extended 6 rounds forward and 3 rounds backward, then we have the following relations, see Fig. 3: $\Delta X_3^0 \in S[\Delta X_2^0]$, $\Delta X_7^0 \in S[\Delta X_6^0]$, $\Delta X_{13}^0 \in S[\Delta X_{12}^0]$, $\Delta X_6^0 \in S[\Delta X_{11}^0]$, $\Delta X_{11}^0 \in S[\Delta X_2^0]$, $\Delta X_1^{27} \in S[\Delta T_2]$, $\Delta X_{15}^{27} \in S[\Delta X_{14}^{27}]$, $\Delta X_{14}^{27} \in S[\Delta X_{11}^{27}]$, $\Delta X_{11}^{27} \in S[\Delta T_2]$ that hold with probability $(\frac{7}{16})^9 = 2^{-10.734}$.*

Observation 4 *If the impossible differential illustrated in Fig. 2 is extended 4 rounds forward and 3 rounds backward, then we have the following relations, see Fig. 4: $\Delta X_1^0 \in S[\Delta X_0^0]$, $\Delta X_{11}^0 \in S[\Delta X_{10}^0]$, $\Delta X_{15}^0 \in S[\Delta X_{14}^0]$, $\Delta X_{14}^0 \in S[\Delta T_7]$, $\Delta X_0^0 \in S[\Delta X_3^0]$, $\Delta X_3^0 \in S[\Delta X_{10}^0]$, $\Delta X_{10}^0 \in S[\Delta T_7]$, $\Delta X_1^{25} \in S[\Delta T_7]$, $\Delta X_{15}^{25} \in S[\Delta X_{14}^{25}]$, $\Delta X_{14}^{25} \in S[\Delta X_{11}^{25}]$, $\Delta X_{11}^{25} \in S[\Delta T_7]$ that hold with probability $(\frac{7}{16})^{11} = 2^{-13.119}$.*

4 Impossible Differential Key-recovery Attack on 27-round T-TWINE-128

In this section, we present the first attack on 27-round T-TWINE-128 in the chosen-tweak model. We use the notion of data structures to generate enough pairs of messages to launch the attack. Our utilized structure takes all the possible values of the 12 nibbles X_2^0, X_3^0, X_4^0, X_5^0, X_6^0, X_7^0, X_8^0, X_9^0, X_{11}^0, X_{12}^0, X_{13}^0, X_{15}^0 while the remaining nibbles assume a fixed value. In addition, we choose the tweak T_2 such that it takes all its possible values. Thus, one structure generates $2^{4\times13} \times (2^{4\times13} - 1)/2 \approx 2^{103}$ possible pairs. Hence, we have 2^{103} possible pairs of messages satisfying the plaintext differences. In addition, we utilize the following pre-computation tables in order to efficiently extract/filter the round keys involved in the analysis rounds:

- H_1: For all the 2^{20} possible values of X_1^1, ΔX_1^1, X_4^1, t_4^0 and $RK_1^0 = K_3$, compute X_2^0, ΔX_2^0, X_3^0, and ΔX_3^0. Then, store X_1^1, ΔX_1^1, X_4^1, and $RK_1^0 = K_3$ in H_1 indexed by X_2^0, ΔX_2^0, X_3^0, ΔX_3^0, and t_4^0. ΔX_3^0 is chosen such that $\Delta X_3^0 \in S[\Delta X_2^0]$, see Observation 3. Therefore, H_1 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_2: For all the 2^{20} possible values of X_3^1, ΔX_3^1, X_8^1, t_3^0, and $RK_3^0 = K_{15}$, compute X_6^0, ΔX_6^0, X_7^0, and ΔX_7^0. Then, store X_3^1, ΔX_3^1, X_8^1, and $RK_3^0 = K_{15}$ in H_1 indexed by X_6^0, ΔX_6^0, X_7^0, ΔX_7^0, and t_3^0. ΔX_7^0 is chosen such that $\Delta X_7^0 \in S[\Delta X_6^0]$, see Observation 3. Therefore, H_2 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_3: For all the 2^{16} possible values of X_{10}^1, ΔX_{15}^1, X_{15}^1, and $RK_6^0 = K_{28}$, compute X_{12}^0, ΔX_{12}^0, X_{13}^0, and ΔX_{13}^0. Then, store X_{10}^1, ΔX_{15}^1, X_{15}^1, and $RK_6^0 = K_{28}$ in H_3 indexed by X_{12}^0, ΔX_{12}^0, X_{13}^0, and ΔX_{13}^0. ΔX_{13}^0 is chosen such that $\Delta X_{13}^0 \in S[\Delta X_{12}^0]$, see Observation 3. Therefore, H_3 has 7×2^{12} rows and on average about $2^{16}/(7 \times 2^{12}) = 16/7$ values in each row.

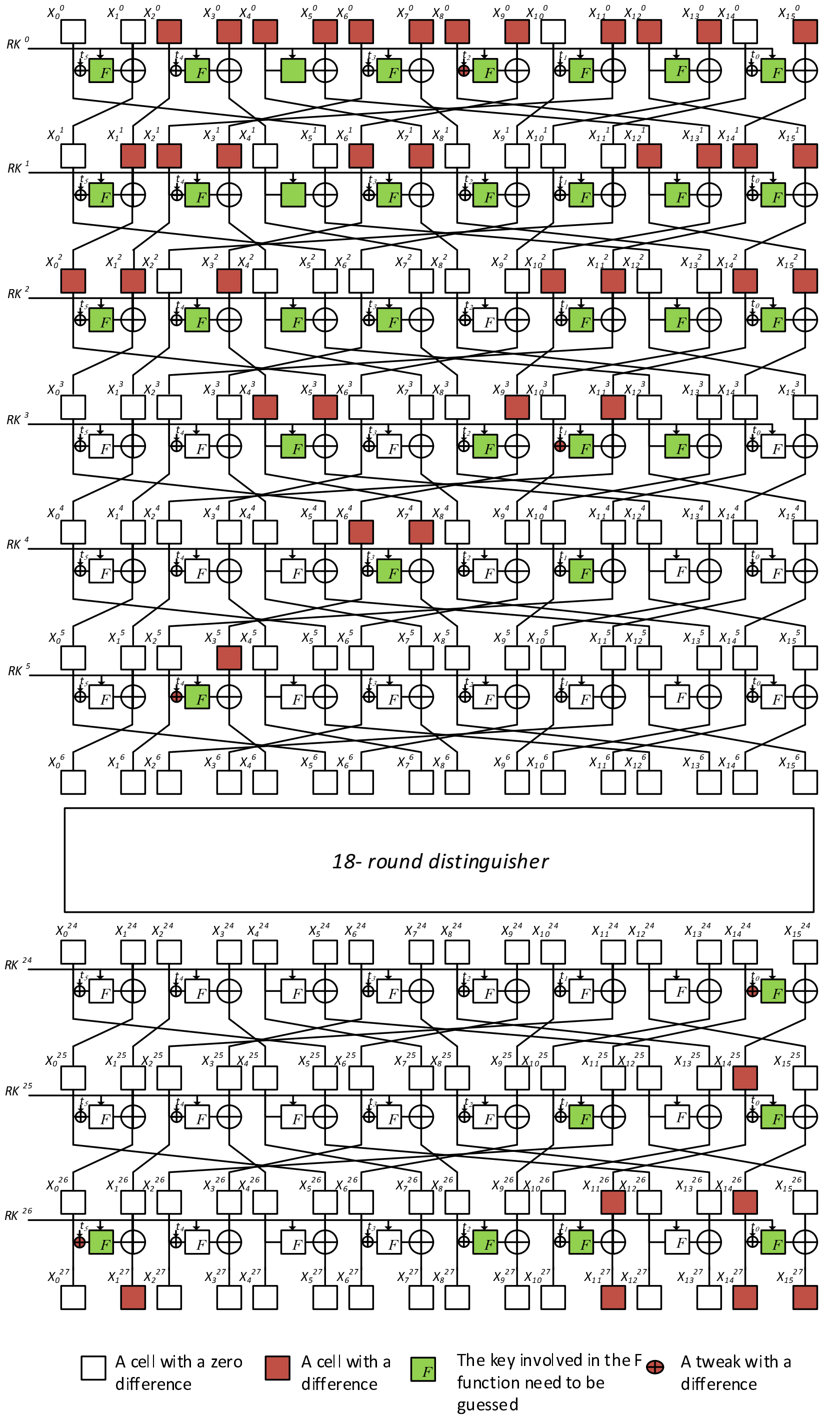
RK 0
RK 1
RK 2
RK 3
RK 4
RK 5
18- round distinguisher
RK 24
RK 25
RK 26
F
□ A cell with a zero difference
🟥 A cell with a difference
🟩 The key involved in the F function need to be guessed
⊕ A tweak with a difference

Fig. 3. Impossible differential attack on 27-round T-TWINE-128

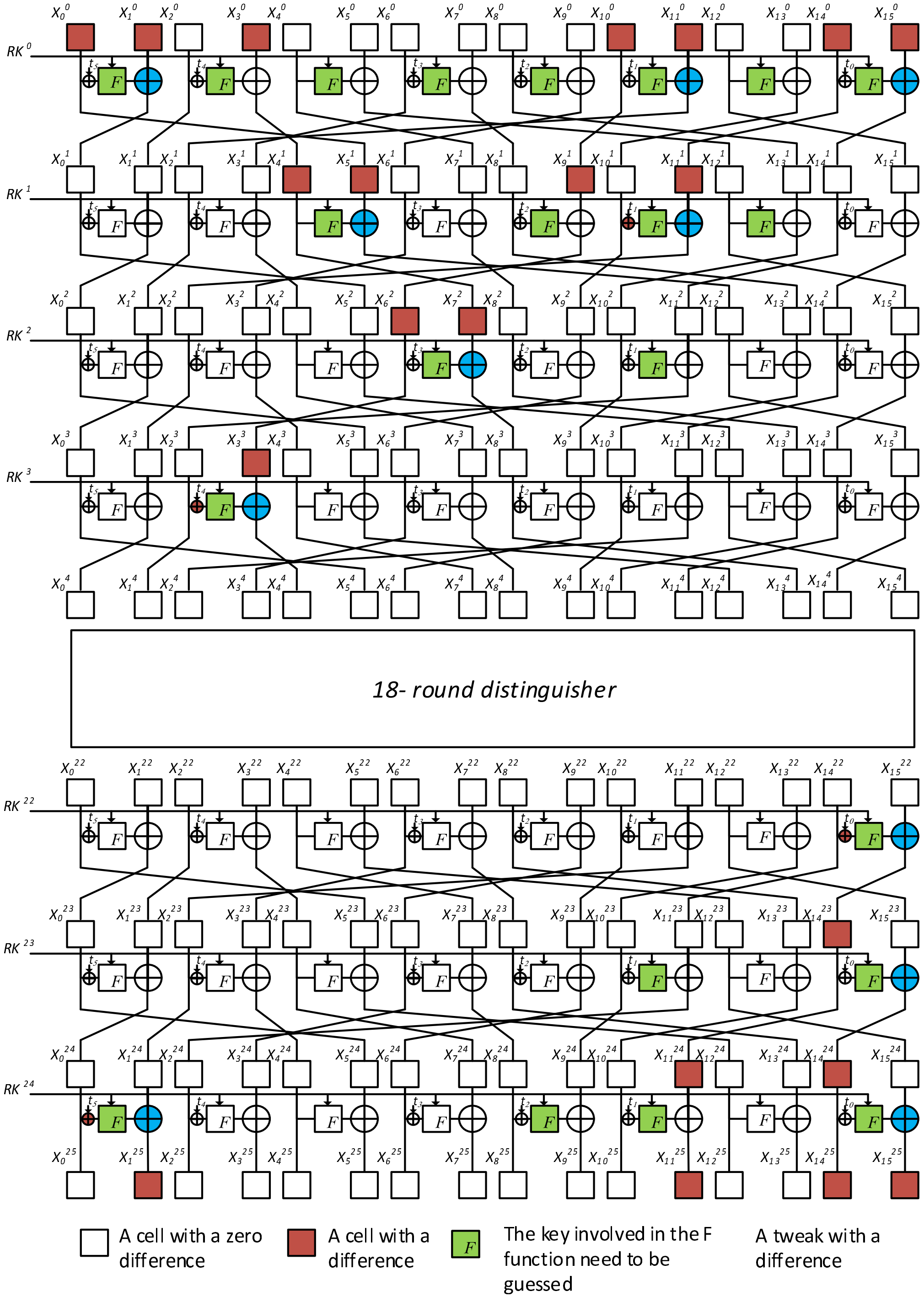
RK 0
RK 1
RK 2
RK 3
18- round distinguisher
RK 22
RK 23
RK 24
F
t0
t1
t2
t3
t4
t5
A cell with a zero difference
A cell with a difference
The key involved in the F function need to be guessed
A tweak with a difference

Fig. 4. Impossible differential attack on 25-round T-TWINE-80

- H_4: For all the 2^{32} possible values of X_1^2, ΔX_1^2, X_4^2, t_4^1, $RK_1^1 = K_7$, X_9^1, t_1^0, and $RK_5^0 = K_{18}$, compute $X_3^1 = X_6^0$, $\Delta X_3^1 = \Delta X_6^0$, X_{10}^0, X_{11}^0, and ΔX_{11}^0. Then, store X_1^2, ΔX_1^2, X_4^2, $RK_1^1 = K_7$, X_9^1, and $RK_5^0 = K_{18}$ in H_4 indexed by $X_3^1 = X_6^0$, $\Delta X_3^1 = \Delta X_6^0$, X_{10}^0, X_{11}^0, ΔX_{11}^0, t_4^1, and t_1^0. ΔX_6^0 is chosen such that $\Delta X_6^0 \in S[\Delta X_{11}^0]$, see Observation 3. Therefore, H_4 has 7×2^{24} rows and on average about $2^{32}/(7 \times 2^{24}) = (16/7) \times 2^4$ values in each row.
- H_5: For all the 2^{40} possible values of X_3^2, ΔX_3^2, X_8^2, t_3^1, $RK_3^1 = K_{19}$, X_{13}^1, ΔX_{13}^1, t_2^0, ΔT_2, and $RK_4^0 = K_{17}$, compute X_4^0, ΔX_4^0, X_8^0, ΔX_8^0, X_9^0, and ΔX_9^0. Then, store X_3^2, ΔX_3^2, X_8^2, $RK_3^1 = K_{19}$, X_{13}^1, ΔX_{13}^1, and $RK_4^0 = K_{17}$ in H_5 indexed by X_4^0, ΔX_4^0, X_8^0, ΔX_8^0, X_9^0, ΔX_9^0, t_3^1, t_2^0, and ΔT_2. H_5 has 2^{36} rows and on average about $2^{40}/2^{36} = 2^4$ values in each row.
- H_6: For all the 2^{44} possible values of X_0^3, X_5^3, ΔX_5^3, t_5^2, $RK_0^2 = K_{10}$, X_5^2, t_5^1, $RK_0^1 = K_6$, X_5^1, t_5^0, and $RK_0^0 = K_2$, compute X_1^2, $\Delta X_1^2 = \Delta X_{11}^0$, $X_1^1 = X_2^0$, $\Delta X_1^1 = \Delta X_2^0$, X_0^0, and X_1^0. Then, store X_5^3, ΔX_5^3, $RK_0^2 = K_{10}$, X_5^2, $RK_0^1 = K_6$, X_5^1, and $RK_0^0 = K_2$ in H_6 indexed by X_1^2, $\Delta X_1^2 = \Delta X_{11}^0$, $X_1^1 = X_2^0$, $\Delta X_1^1 = \Delta X_2^0$, X_0^0, X_1^0, t_5^0, t_5^1, and t_5^2. ΔX_{11}^0 is chosen such that $\Delta X_{11}^0 \in S[\Delta X_2^0]$, see Observation 3. Therefore, H_6 has 7×2^{32} rows and on average about $2^{44}/(7 \times 2^{32}) = (16/7) \times 2^8$ values in each row.
- H_7: For all the 2^{32} possible values of X_{10}^2, ΔX_{10}^2, X_{15}^2, ΔX_{15}^2, $RK_6^1 = K_1 + S(K_0)$, X_7^1, ΔX_7^1, and $RK_2^0 = K_{12}$, compute X_{13}^1, ΔX_{13}^1, X_4^0, ΔX_4^0, X_5^0, and ΔX_5^0. Then, store X_{10}^2, ΔX_{10}^2, X_{15}^2, ΔX_{15}^2, $RK_6^1 = K_1 + S(K_0)$, X_7^1, ΔX_7^1, and $RK_2^0 = K_{12}$ in H_7 indexed by X_{13}^1, ΔX_{13}^1, X_4^0, ΔX_4^0, X_5^0, and ΔX_5^0. H_7 has 2^{24} rows and on average about $2^{32}/2^{24} = 2^8$ values in each row.
- H_8: For all the 2^{36} possible values of X_{11}^2, ΔX_{11}^2, X_{14}^2, ΔX_{14}^2, t_0^1, $RK_7^1 = K_0$, X_{11}^1, t_0^0, and $RK_7^0 = K_{31}$, compute X_{15}^1, ΔX_{15}^1, X_{14}^0, X_{15}^0, and ΔX_{15}^0. Then, store X_{11}^2, ΔX_{11}^2, X_{14}^2, ΔX_{14}^2, $RK_7^1 = K_0$, X_{11}^1, and $RK_7^0 = K_{31}$ in H_8 indexed by X_{15}^1, ΔX_{15}^1, X_{14}^0, X_{15}^0, ΔX_{15}^0, t_0^1, and t_0^0. H_8 has 2^{28} rows and on average about $2^{36}/2^{28} = 2^8$ values in each row.
- H_9: For all the 2^{20} possible values of X_2^3, X_9^3, ΔX_9^3, t_1^2, and $RK_5^2 = K_{26}$, compute X_{10}^2, ΔX_{10}^2, X_{11}^2, and ΔX_{11}^2. Then, store X_9^3, ΔX_9^3, and $RK_5^2 = K_{26}$ in H_9 indexed by X_{10}^2, ΔX_{10}^2, X_{11}^2, ΔX_{11}^2, and t_1^2. H_9 has 2^{20} rows and on average about $2^{20}/2^{20} = 1$ value in each row.
- H_{10}: For all the 2^{20} possible values of X_{11}^3, ΔX_{11}^3, X_{14}^3, t_0^2, and $RK_7^2 = K_4 + S(K_{16})$, compute X_{14}^2, ΔX_{14}^2, X_{15}^2, and ΔX_{15}^2. Then, store X_{11}^3, ΔX_{11}^3, X_{14}^3, and $RK_7^2 = K_4 + S(K_{16})$ in H_{10} indexed by X_{14}^2, ΔX_{14}^2, X_{15}^2, ΔX_{15}^2, and t_0^2. H_{10} has 2^{20} rows and on average about $2^{20}/2^{20} = 1$ value in each row.
- H_{11}: For all the 2^{40} possible values of X_7^4, ΔX_7^4, X_{12}^4, $RK_2^3 = K_{24}$, X_1^3, t_4^2, $RK_1^2 = K_{11}$, X_9^2, t_1^1, and $RK_5^1 = K_{22}$, compute X_5^3, ΔX_5^3, X_3^2, ΔX_3^2, X_{10}^1, and X_{11}^1. Then, store X_7^4, ΔX_7^4, $RK_2^3 = K_{24}$, $RK_1^2 = K_{11}$, and $RK_5^1 = K_{22}$ in H_{11} indexed by X_5^3, ΔX_5^3, X_3^2, ΔX_3^2, X_{10}^1, X_{11}^1, t_4^2, and t_1^1. H_{11} has 2^{32} rows and on average about $2^{40}/2^{32} = 2^8$ values in each row.
- H_{12}: For all the 2^{12} possible values of X_7^2, X_{12}^2, and $RK_2^1 = K_{16}$, compute X_4^1, and X_5^1. Then, store X_7^2, X_{12}^2, and $RK_2^1 = K_{16}$ in H_{12} indexed by X_4^1, and X_5^1. H_{12} has 2^8 rows and on average about $2^{12}/2^8 = 2^4$ value in each row.

- H_{13}: For all the 2^{16} possible values of X_6^2, X_{13}^2, t_2^1, and $RK_4^1 = K_{21}$, compute X_8^1, and X_9^1. Then, store X_6^2, X_{13}^2, and $RK_4^1 = K_{21}$ in H_{13} indexed by X_8^1, X_9^1, and t_2^1. H_{13} has 2^{12} rows and on average about $2^{16}/2^{12} = 2^4$ value in each row.
- H_{14}: For all the 2^{28} possible values of X_2^4, X_9^4, t_1^3, ΔT_2, $RK_5^3 = K_{30}$, X_{15}^3, and $RK_6^2 = K_5 + S(K_4 + S(K_{16}))$, compute X_{11}^3, ΔX_{11}^3, X_{12}^2, and X_{13}^2. Then, store $RK_5^3 = K_{30}$, and $RK_6^2 = K_5 + S(K_4 + S(K_{16}))$ in H_{14} indexed by X_{11}^3, ΔX_{11}^3, X_{12}^2, t_1^3, ΔT_2, and X_{13}^2. H_{14} has 2^{24} rows and on average about $2^{28}/2^{24} = 2^4$ values in each row.
- H_{15}: For all the 2^{44} possible values of X_3^5, ΔX_3^5, X_8^5, t_3^4, $RK_3^4 = K_{31} + S(K_7)$, X_{13}^4, t_2^3, $RK_4^3 = K_{29}$, X_3^3, t_3^2, and $RK_3^2 = K_{23} + S(K_{30})$, compute X_7^4, ΔX_7^4, X_9^3, ΔX_9^3, X_6^2, and X_7^2. Then, store X_3^5, ΔX_3^5, $RK_4^3 = K_{29}$, and $RK_3^2 = K_{23} + S(K_{30})$ in H_{15} indexed by X_7^4, ΔX_7^4, X_9^3, ΔX_9^3, X_6^2, X_7^2, t_3^4, t_2^3, t_3^2, and $RK_3^4 = K_{31} + S(K_7)$. H_{15} has 2^{40} rows and on average about $2^{44}/2^{40} = 2^4$ values in each row.
- H_{16}: For all the 2^{48} possible values of X_1^6, X_4^6, t_4^5, ΔT_2, $RK_1^5 = K_{23} + S(K_{30})$, X_9^5, t_1^4, $RK_5^4 = K_3$, X_{15}^4, $RK_6^3 = K_9 + S(K_8 + S(K_{20}))$, X_7^3, and $RK_2^2 = K_{20}$, compute X_3^5, ΔX_3^5, X_{14}^3, X_8^2, X_4^2, and X_5^2. Then, store $RK_5^4 = K_3$, $RK_6^3 = K_9 + S(K_8 + S(K_{20}))$, and $RK_2^2 = K_{20}$ in H_{16} indexed by X_3^5, ΔX_3^5, X_{14}^3, X_8^2, X_4^2, X_5^2, $RK_1^5 = K_{23} + S(K_{30})$, $RK_5^4 = K_3$, t_4^5, ΔT_2, and t_1^4. H_{16} has 2^{44} rows and on average about $2^{48}/2^{44} = 2^4$ values in each row.
- H_{17}: For all the 2^{20} possible values of X_{14}^{26}, ΔX_{14}^{26}, X_{15}^{26}, t_0^{26}, and $RK_7^{26} = f_1(K_0, K_1, K_4, K_5, K_6, K_7, RK_6^3, K_{10}, K_{16}, K_{17}, K_{18}, K_{19}, K_{20}, K_{21}, K_{28}, K_{29}, K_{30})$, compute X_{14}^{27}, ΔX_{14}^{27}, X_{15}^{27}, and ΔX_{15}^{27}. Then, store X_{14}^{26} and ΔX_{14}^{26} in H_{17} indexed by X_{14}^{27}, ΔX_{14}^{27}, X_{15}^{27}, ΔX_{15}^{27}, RK_7^{26}, and t_0^{26}. ΔX_{15}^{27} is chosen such that $\Delta X_{15}^{27} \in S[\Delta X_{14}^{27}]$, see Observation 3. Therefore, H_{17} has 7×2^{20} rows and on average about $2^{20}/(7 \times 2^{20}) = (16/7) \times 2^{-4}$ values in each row.
- H_{18}: For all the 2^{20} possible values of X_0^{26}, X_1^{26}, t_5^{26}, ΔT_2, and $RK_0^{26} = f_2(K_0,K_1,K_3,K_{16},K_{20},K_{21},RK_6^3,K_{27},K_{28})$, compute X_0^{27}, ΔX_1^{27}, and X_1^{27}. Then, store RK_0^{26} in H_{18} indexed by X_0^{27}, ΔX_1^{27}, X_1^{27}, t_5^{26}, and ΔT_2. ΔX_1^{27} is chosen such that $\Delta X_1^{27} \in S[\Delta T_2]$, see Observation 3. Therefore, H_{18} has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_{19}: For all the 2^{28} possible values of X_{10}^{25}, X_{11}^{25}, t_1^{25}, $RK_5^{25} = f_3(K_0,K_1,K_2,K_4,K_{12},K_{13},RK_6^3,K_{15},K_{16},K_{20},K_{21},K_{24},K_{28})$, X_8^{26}, t_2^{26}, and $RK_4^{26} = f_4(K_0,K_4,K_5,K_{11},K_{16},K_{24})$, compute X_2^{27}, X_9^{27}, and X_8^{27}. Then, store X_{11}^{25} and RK_5^{25} in H_{19} indexed by X_2^{27}, X_9^{27}, X_8^{27}, RK_4^{26}, t_1^{25}, and t_2^{26}. H_{19} has 2^{24} rows and on average about $2^{28}/2^{24} = 2^4$ values in each row.
- H_{20}: For all the 2^{44} possible values of X_{14}^{24}, X_{15}^{24}, t_0^{24}, ΔT_2, $RK_7^{24} = f_5(K_0,K_1,K_2,K_{10},K_{11},K_{12},K_{13},RK_6^3,K_{20},K_{21},K_{22},K_{24},K_{28},K_{29},K_{30})$, X_{15}^{25},t_0^{25},$RK_7^{25} = f_6(K_0, K_1,K_2,K_3,K_4,K_5,K_6,K_{12},K_{13},K_{14},K_{15},K_{16},K_{17},K_{24},K_{25},K_{26},K_{28})$, X_{10}^{26}, t_1^{26},and $RK_5^{26} = f_7(K_0,K_1,K_4,K_5,K_6,K_8,K_{12},K_{13},K_{16},K_{17},K_{19},K_{20},K_{24},K_{25},K_{28})$, compute X_{11}^{25}, $X_{14}^{27} = X_{14}^{26}$, $\Delta X_{14}^{27} = \Delta X_{14}^{26}$, X_{11}^{27}, ΔX_{11}^{27}, and X_{10}^{27}. Then, store RK_7^{25},RK_5^{26} in H_{20} indexed by X_{11}^{25}, X_{14}^{26}, ΔX_{14}^{26}, X_{11}^{27}, ΔX_{11}^{27}, X_{10}^{27}, t_0^{24}, ΔT_2, RK_7^{24}, t_0^{25}, and t_1^{26}. ΔX_{14}^{27} and ΔX_{11}^{27} are chosen such that $\Delta X_{14}^{27} \in S[\Delta X_{11}^{27}]$ and $\Delta X_{11}^{27} \in S[\Delta T_2]$, receptively, see Observation 3.

Therefore, H_{19} has $7^2 \times 2^{36}$ rows and on average about $2^{44}/(7^2 \times 2^{36}) = (16/7)^2$ values in each row.

In the general approach, the round keys involved in the analysis rounds are guessed and the plaintext/ciphertext pairs are filtered to satisfy the differential path leading to the impossible differential distinguisher. Here, we use the above proposed pre-computation tables to deduce the round keys that lead a specific pair of plaintext/ciphertext to the impossible differential. Then, we exclude the deduced keys as they are wrong keys. Our attack proceeds as follows. We initialize an array H of $2^{31\times 4=124}$ entries to "0", where each entry is 1-bit and the index of the array is 31 key nibbles involved in the attack, as we will see later. Then we generate 2^m structures as described above. Therefore, we have 2^{m+103} pairs of plaintext/ciphertext pairs generated using 2^{m+48} chosen plaintexts. Then, we ask the encryption oracle for their corresponding ciphertexts. The plaintext/ciphertext pairs that satisfy Observation 3 are $2^{m+103} \times 2^{-10.734} = 2^{m+92.266}$ pairs. After the ciphertext filtration, we have only $2^{m+92.266} \times 2^{-12\times 4} = 2^{m+44.266}$ remaining pairs. For each remaining pair, we access the the pre-computation tables in sequential order from table H_1 to H_{20} one by one in order to deduce 31 key nibbles that lead each remaining pair of plaintext/ciphertext to the impossible differential. Then, we mark them in H as invalid "1". Table 3 summarize these steps by identifying which table will be used and which key nibble is involved in this step in addition to the time complexity of each step.

Remarks on the analysis steps:

1. During steps 1 - 14 and step 18, we directly access the corresponding table to obtain the values of the involved key nibbles. For example, in step 1, we determine the number of possible values of $RK_1^0 = K_3$ that satisfy the path to the impassible differential by accessing H_1. Therefore, we have (16/7) possible values for K_3.
2. During steps 15, 16, 17, 19, and 20, and because some combinations of the key nibbles determined during the previous steps are used in the indexing of the tables H_{15} to H_{20}, we firstly deduce these indices and then access the corresponding table. For example, during step 15, we deduce the value of $RK_3^4 = K_{31} + S(K_7)$ that is used in the indexing of table H_{15}, then determine the number of possible values of $RK_4^3 = K_{29}$ and $RK_3^2 = K_{23} + S(K_{30})$ that satisfy the path by accessing H_{15}. After that, the value of $RK_3^2 = K_{23} + S(K_{30})$ is used to deduce the value of K_{23} using the determined value of K_{30} from Step 14.
3. During steps 7 and 8, we determine the possible values of $RK_6^1 = K_1 + S(K_0)$ and $RK_7^1 = K_0$, respectively. Therefore, after step 8, we can deduce the values of K_1. In the same manner, we can deduce the values of K_4 and K_5 after steps 10, 12 and 14 where we determine the values of $RK_7^2 = K_4 + S(K_{16})$ and $RK_2^1 = K_{16}$, and $RK_6^2 = K_5 + S(K_4 + S(K_{16}))$, respectively.
4. During step 17, we deduce the value of $RK_7^{26} = f_1(K_0, K_1, K_4, K_5, K_6, K_7, RK_6^3, K_{10}, K_{16}, K_{17}, K_{18}, K_{19}, K_{20}, K_{21}, K_{28}, K_{29}, K_{30})$, then determine the

values of X_{14}^{26} and $\varDelta X_{14}^{26}$ that satisfy the path by accessing H_{17}. Therefore, no new key nibbles are involved during this step but there is a filtration of some keys.

5. During steps 18 and 19, we can determine the values of $RK_0^{26} = f_2(K_0, K_1, K_3, K_{16}, K_{20}, K_{21}, RK_6{}^3, K_{27}, K_{28})$ and $RK_5^{25} = f_3(K_0, K_1, K_2, K_4, K_{12}, K_{13}, RK_6^3, K_{15}, K_{16}, K_{20}, K_{21}, K_{24}, K_{28})$, respectively. Therefore, we can deduce the values of K_{27} and K_{13}, respectively, since all the other key nibbles in f_2 and f_3 are determined during the previous steps.
6. After step 20, we have $2^{60} \times (16/7)^9$ possible values for $K_0, K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_9{+}S(K_8{+}S(K_{20})), K_{10}, K_{11}, K_{12}, K_{13}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, K_{20}, K_{21}, K_{22}, K_{23}, K_{24}, K_{26}, K_{27}, K_{28}, K_{29}, K_{30}, K_{31}, RK_7^{25} = f_6(K_{14}, K_{25})$, $RK_5^{26} = f_7(K_8, K_{25})$. Hence, we marks them in H as invalid "1" in step 21.

Attack Complexity. As depicted in Fig. 3, we have 37 round keys involved in the analysis rounds. According to the key schedule, these 37 round keys take only 2^{124} possible values (see step 21 in Table 3). As mentioned in step 21, we remove on average $2^{60} \times (16/7)^9 = 2^{70.734}$ out of 2^{124} possible values of these 31 round keys involved in the attack, for each one message pair of the $2^{m+44.266}$ remaining pairs. Hence, a wrong key is not discarded with one pair with probability $1 - 2^{70.734-124} = 1 - 2^{-53.266}$. Therefore, we have $2^{124} \times (1 - 2^{-53.266})^{2^{m+44.266}} \approx 2^{124} \times (e^{-1})^{2^{m+44.266-53.266}} \approx 2^{124} \times 2^{-1.4 \times 2^{m-9}}$ remaining candidates for 124-bit of the key, after processing all the $2^{m+44.266}$ remaining pairs. We evaluated the computational complexity of the attack as a function of m, as illustrated in Table 3, to determine the optimal value of m that leads to the best computational complexity. As steps 20 and 21 dominate the time complexity of the attack, see Table 3, we choose $m = 12$ in order to optimize the time complexity of the attack. Therefore, we have $2^{124} \times 2^{-1.4 \times 2^{12-9=3}} = 2^{124-11.2} = 2^{112.8}$ remaining candidates for 124-bit of the key. The remaining key nibbles can be retrieved by guessing K_8 and exhaustively searching the $2^{112.8}$ remaining key candidates using 2 plaintext/ciphertext pairs. This step requires $2 \times 2^4 \times 2^{112.8} = 2^{117.8}$ encryptions. Therefore, the time complexity of the attack is $2^{120.245} + 2^{119.245} + 2^{117.8} \approx 2^{120.83}$ encryptions. The data complexity of the attack is $2^{m+48} = 2^{12+48} = 2^{60}$ chosen plaintexts. The memory complexity of the attack is dominated by the memory that is required to store H. Hence, the memory complexity is $2^{124} \times 2^{-6} = 2^{118}$ 64-bit blocks.

5 Impossible Differential Key-recovery Attack on 25-round T-TWINE-80

In this section, we present the first attack on 25-round T-TWINE-80 in the chosen-tweak model. We use the notion of data structures to generate enough pairs of messages to launch the attack. Our utilized structure takes all the possible values in 7 nibbles X_0^0, X_1^0, X_3^0, X_{10}^0, X_{11}^0, X_{14}^0, X_{15}^0 while the remaining nibbles take a fixed value. In addition, we choose the tweak T_7 such that it takes

Table 3. Time complexity of the different steps of the attack on 27-round T-TWINE-128, where NK denotes the number of keys to be excluded.

Step	Table	Key nibbles	Time Complexity (in 27-round encryptions)	NK	$m = 12$
1	H_1	K_3	$2^{m+44.266} \times (16/7) \times \frac{4}{8 \times 27} \approx 2^{m+39.704}$	$(16/7)$	$2^{51.704}$
2	H_2	K_{15}	$2^{m+44.266} \times (16/7)^2 \times \frac{4}{8 \times 27} \approx 2^{m+40.896}$	$(16/7)^2$	$2^{52.896}$
3	H_3	K_{28}	$2^{m+44.266} \times (16/7)^3 \times \frac{4}{8 \times 27} \approx 2^{m+42.089}$	$(16/7)^3$	$2^{54.089}$
4	H_4	K_7, K_{18}	$2^{m+44.266} \times 2^4 \times (16/7)^4 \times \frac{6}{8 \times 27} \approx 2^{m+47.867}$	$2^4 \times (16/7)^4$	$2^{59.867}$
5	H_5	K_{17}, K_{19}	$2^{m+44.266} \times 2^8 \times (16/7)^4 \times \frac{7}{8 \times 27} \approx 2^{m+52.089}$	$2^8 \times (16/7)^4$	$2^{64.089}$
6	H_6	K_2, K_6, K_{10}	$2^{m+44.266} \times 2^{16} \times (16/7)^5 \times \frac{7}{8 \times 27} \approx 2^{m+61.282}$	$2^{16} \times (16/7)^5$	$2^{73.282}$
7	H_7	$K_1 + S(K_0), K_{12}$	$2^{m+44.266} \times 2^{24} \times (16/7)^5 \times \frac{8}{8 \times 27} \approx 2^{m+69.474}$	$2^{24} \times (16/7)^5$	$2^{81.474}$
8	H_8	K_0, K_1, K_{31}	$2^{m+44.266} \times 2^{32} \times (16/7)^5 \times \frac{7}{8 \times 27} \approx 2^{m+77.282}$	$2^{32} \times (16/7)^5$	$2^{89.282}$
9	H_9	K_{26}	$2^{m+44.266} \times 2^{32} \times (16/7)^5 \times \frac{3}{8 \times 27} \approx 2^{m+76.059}$	$2^{32} \times (16/7)^5$	$2^{88.059}$
10	H_{10}	$K_4 + S(K_{16})$	$2^{m+44.266} \times 2^{32} \times (16/7)^5 \times \frac{4}{8 \times 27} \approx 2^{m+76.474}$	$2^{32} \times (16/7)^5$	$2^{88.474}$
11	H_{11}	K_{11}, K_{22}, K_{24}	$2^{m+44.266} \times 2^{40} \times (16/7)^5 \times \frac{5}{8 \times 27} \approx 2^{m+84.796}$	$2^{40} \times (16/7)^5$	$2^{96.796}$
12	H_{12}	K_4, K_{16}	$2^{m+44.266} \times 2^{44} \times (16/7)^5 \times \frac{3}{8 \times 27} \approx 2^{m+88.059}$	$2^{44} \times (16/7)^5$	$2^{100.059}$
13	H_{13}	K_{21}	$2^{m+44.266} \times 2^{48} \times (16/7)^5 \times \frac{3}{8 \times 27} \approx 2^{m+92.059}$	$2^{48} \times (16/7)^5$	$2^{104.059}$
14	H_{14}	K_5, K_{30}	$2^{m+44.266} \times 2^{52} \times (16/7)^5 \times \frac{2}{8 \times 27} \approx 2^{m+95.474}$	$2^{52} \times (16/7)^5$	$2^{107.474}$
15	H_{15}	K_{23}, K_{29}	$2^{m+44.266} \times 2^{56} \times (16/7)^5 \times \frac{4}{8 \times 27} \approx 2^{m+100.474}$	$2^{56} \times (16/7)^5$	$2^{112.474}$
16	H_{16}	$RK_6^3 = K_9 + S(K_8 + S(K_{20})), K_{20}$	$2^{m+44.266} \times 2^{60} \times (16/7)^5 \times \frac{3}{8 \times 27} \approx 2^{m+104.059}$	$2^{60} \times (16/7)^5$	$2^{116.059}$
17	H_{17}	-	$2^{m+44.266} \times 2^{56} \times (16/7)^6 \times \frac{2}{8 \times 27} \approx 2^{m+100.667}$	$2^{56} \times (16/7)^6$	$2^{112.667}$
18	H_{18}	K_{27}	$2^{m+44.266} \times 2^{56} \times (16/7)^7 \times \frac{1}{8 \times 27} \approx 2^{m+100.860}$	$2^{56} \times (16/7)^7$	$2^{112.860}$
19	H_{19}	K_{13}	$2^{m+44.266} \times 2^{60} \times (16/7)^7 \times \frac{2}{8 \times 27} \approx 2^{m+105.860}$	$2^{60} \times (16/7)^7$	$2^{117.860}$
20	H_{20}	$RK_7^{25} = f_6(K_{14}, K_{25}),$ $RK_5^{26} = f_7(K_8, K_{25})$	$2^{m+44.266} \times 2^{60} \times (16/7)^9 \times \frac{2}{8 \times 27} \approx 2^{m+108.245}$	$2^{60} \times (16/7)^9$	$2^{120.245}$
21	H	-	$2^{m+44.266} \times 2^{60} \times (16/7)^9 \times \frac{1}{8 \times 27} \approx 2^{m+107.245}$	$2^{60} \times (16/7)^9$	$2^{119.245}$

all the values. Thus, one structure generates $2^{4\times 8} \times (2^{4\times 8} - 1)/2 \approx 2^{63}$ possible pairs. Hence, we have 2^{63} possible pairs of messages satisfying the plaintext differences. In addition, we utilize the following pre-computation tables in order to efficiently extract/filter the round keys involved in the analysis rounds. Note that, for the 7 round keys that are involved in the 3 rounds below the distinguisher, we wrote them as 7 functions $f_1, f_2, f_3, f_4, f_5, f_6, f_7$ of the key nibbles that are not involved in the above analysis rounds, $K_0, K_2, K_5, K_7, K_9, K_{10}, K_{11}$, K_{12}, K_{13}, and ignored the other key nibbles as they are known.

- H_1: For all the 2^{20} possible values of X_0^1, X_5^1, ΔX_5^1, t_5^0 and $RK_0^0 = K_1$, compute X_0^0, ΔX_0^0, X_1^0, and ΔX_1^0. Then, store X_5^1, ΔX_5^1, and $RK_0^0 = K_1$ in H_1 indexed by X_0^0, ΔX_0^0, X_1^0, ΔX_1^0, and t_5^0. ΔX_1^0 is chosen such that $\Delta X_1^0 \in S[\Delta X_0^0]$, see Observation 4. Therefore, H_1 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_2: For all the 2^{20} possible values of X_2^1, X_9^1, ΔX_9^1, t_1^0, and $RK_5^0 = K_{14}$, compute X_{10}^0, ΔX_{10}^0, X_{11}^0, and ΔX_{11}^0. Then, store X_9^1, ΔX_9^1, and $RK_5^0 = K_{14}$ in H_2 indexed by X_{10}^0, ΔX_{10}^0, X_{11}^0, ΔX_{11}^0, and t_1^0. ΔX_{11}^0 is chosen such that $\Delta X_{11}^0 \in S[\Delta X_{10}^0]$, see Observation 4. Therefore, H_2 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_3: For all the 2^{20} possible values of X_{11}^1, ΔX_{11}^1, X_{14}^1, t_0^0, and $RK_7^0 = K_{16}$, compute X_{14}^0, ΔX_{14}^0, X_{15}^0, and ΔX_{15}^0. Then, store X_{11}^1, ΔX_{11}^1, X_{14}^1, and $RK_7^0 = K_{16}$ in H_3 indexed by X_{14}^0, ΔX_{14}^0, X_{15}^0, ΔX_{15}^0, and t_0^0. ΔX_{15}^0 is chosen such that $\Delta X_{15}^0 \in S[\Delta X_{14}^0]$, see Observation 4. Therefore, H_3 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_4: For all the 2^{28} possible values of X_7^2, ΔX_7^2, X_{12}^2, $RK_2^1 = K_8$, X_1^1, t_4^0, and $RK_1^0 = K_3$, compute $X_5^1 = X_0^0$, ${}_5^1 = \Delta X_0^0$, X_2^0, X_3^0, and ΔX_3^0. Then, store X_7^2, ΔX_7^2, $RK_2^1 = K_8$, and $RK_1^0 = K_3$ in H_4 indexed by $X_5^1 = X_0^0$, ${}_5^1 = \Delta X_0^0$, X_2^0, X_3^0, ΔX_3^0, and t_4^0. ΔX_0^0 is chosen such that $\Delta X_0^0 \in S[\Delta X_3^0]$, see Observation 4. Therefore, H_4 has 7×2^{20} rows and on average about $2^{28}/(7 \times 2^{20}) = (16/7) \times 2^4$ values in each row.
- H_5: For all the 2^{28} possible values of X_2^2, X_9^2, t_1^1, ΔT_7, $RK_5^1 = K_{18}$, X_{15}^1, and $RK_6^0 = K_{15}$, compute $X_{14}^0 = X_{11}^1$, $\Delta X_{14}^0 = \Delta X_{11}^1$, X_{12}^0, and X_{13}^0. Then, store $RK_5^1 = K_{18}$ and $RK_6^0 = K_{15}$ in H_5 indexed by X_{11}^1, ΔX_{11}^1, X_{12}^0, X_{13}^0, t_1^1, and ΔT_7. ΔX_{14}^0 is chosen such that $\Delta X_{14}^0 \in S[\Delta T_7]$, see Observation 4. Therefore, H_5 has 7×2^{20} rows and on average about $2^{28}/(7 \times 2^{20}) = (16/7) \times 2^4$ values in each row.
- H_6: For all the 2^{44} possible values of X_3^3, ΔX_3^3, X_8^3, t_3^2, $RK_3^2 = K_{14}$, X_{13}^2, t_2^1, $RK_4^1 = K_{17}$, X_3^1, t_3^0, and $RK_3^0 = K_6$, compute X_7^2, $\Delta X_7^2 = \Delta X_3^0$, X_9^1, $\Delta X_9^1 = \Delta X_{10}^0$, X_6^0, and X_7^0. Then, store X_3^3, ΔX_3^3, $RK_3^2 = K_{14}$, $RK_4^1 = K_{17}$, and $RK_3^0 = K_6$ in H_6 indexed by X_7^2, $\Delta X_7^2 = \Delta X_3^0$, X_9^1, $\Delta X_9^1 = \Delta X_{10}^0$, X_6^0, X_7^0, t_3^2, t_2^1, t_3^0, and $RK_3^2 = K_{14}$. ΔX_3^0 is chosen such that $\Delta X_3^0 \in S[\Delta X_{10}^0]$, see Observation 4. Therefore, H_6 has 7×2^{36} rows and on average about $2^{44}/(7 \times 2^{36}) = (16/7) \times 2^4$ values in each row.
- H_7: For all the 2^{48} possible values of X_1^4, X_4^4, t_4^3, ΔT_7, $RK_1^3 = K_{15}$, X_9^3, t_1^2, $RK_5^2 = K_3$, X_{15}^2, $RK_6^1 = K_{19}$, X_7^1, and $RK_2^0 = K_4$, compute X_3^3, $\Delta X_{10}^0 = \Delta X_3^3$, X_{14}^1, X_8^0, X_4^0, and X_5^0. Then, store $RK_1^3 = K_{15}$, $RK_5^2 = K_3$, $RK_6^1 =$

K_{19}, and $RK^0_2 = K_4$ in H_7 indexed by X^3_3, $\Delta X^0_{10} = \Delta X^3_3$, X^1_{14}, X^0_8, X^0_4, X^0_5, $RK^3_1 = K_{15}$, $RK^2_5 = K_3$, t^3_4, ΔT_7, and t^2_1. ΔX^0_{10} is chosen such that $\Delta X^0_{10} \in S[\Delta T_7]$, see Observation 4. Therefore, H_7 has 7×2^{40} rows and on average about $2^{48}/(7 \times 2^{40}) = (16/7) \times 2^4$ values in each row.

- H_8: For all the 2^{20} possible values of X^{24}_0, X^{24}_1, t^{24}_5, ΔT_7, and $RK^{24}_0 = f_1(K_0,K_2, K_5,K_9,K_{10},K_{12},K_{13})$, compute X^{25}_0, ΔX^{25}_1, and X^{25}_1. Then, store RK^{24}_0 in H_8 indexed by X^{25}_0, ΔX^{25}_1, X^{25}_1, t^{24}_5, and ΔT_7. ΔX^{25}_1 is chosen such that $\Delta X^{25}_1 \in S[\Delta T_7]$, see Observation 4. Therefore, H_8 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_9: For all the 2^{20} possible values of X^{24}_{14}, ΔX^{24}_{14}, X^{24}_{15}, t^{24}_0, and $RK^{24}_7 = f_2(K_0,K_2,K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$, compute X^{25}_{14}, ΔX^{25}_{14}, X^{25}_{15}, and ΔX^{25}_{15}. Then, store X^{24}_{14}, ΔX^{24}_{14}, and $RK^{24}_7 = f_2(K_0,K_2,K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$ in H_9 indexed by X^{25}_{14}, ΔX^{25}_{14}, X^{25}_{15}, ΔX^{25}_{15}, and t^{24}_0. ΔX^{25}_{15} is chosen such that $\Delta X^{25}_{15} \in S[\Delta X^{25}_{14}]$, see Observation 4. Therefore, H_9 has 7×2^{16} rows and on average about $2^{20}/(7 \times 2^{16}) = 16/7$ values in each row.
- H_{10}: For all the 2^{32} possible values of X^{23}_{14}, ΔX^{23}_{14}, X^{23}_{15}, t^{23}_0, $RK^{23}_7 = f_3(K_0,K_2, K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$, X^{24}_{10}, t^{24}_1, and $RK^{24}_5 = f_4(K_0,K_2,K_5,K_7,K_9, K_{10},K_{11},K_{12},K_{13})$, compute $X^{25}_{14} = X^{24}_{14}$, $\Delta X^{25}_{14} = \Delta X^{24}_{14}$, X^{25}_{11}, ΔX^{25}_{11}, and X^{25}_{10}. Then, store X^{23}_{14}, ΔX^{23}_{14}, $RK^{23}_7 = f_3(K_0,K_2,K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$, and $RK^{24}_5 = f_4(K_0,K_2,K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$ in H_{10} indexed by $X^{25}_{14} = X^{24}_{14}$, $\Delta X^{25}_{14} = \Delta X^{24}_{14}$, X^{25}_{11}, ΔX^{25}_{11}, X^{25}_{10}, t^{23}_0, and t^{24}_1. ΔX^{25}_{14} is chosen such that $\Delta X^{25}_{14} \in S[\Delta X^{25}_{11}]$, see Observation 4. Therefore, H_{10} has 7×2^{24} rows and on average about $2^{32}/(7 \times 2^{24}) = (16/7) \times 2^4$ values in each row.
- H_{11}: For all the 2^{44} possible values of X^{22}_{14}, X^{22}_{15}, t^{22}_0, ΔT_7, $RK^{22}_7 = f_5(K_0,K_2, K_5,K_7,K_9,K_{10},K_{11},K_{12},K_{13})$, X^{23}_{10}, t^{23}_1,$RK^{23}_5 = f_6(K_0,K_2,K_5,K_7,K_9,K_{10},K_{11}, K_{12},K_{13})$, X^{24}_8, t^{24}_2, and $RK^{24}_4 = f_7(K_0,K_2,K_5,K_9,K_{10},K_{12},K_{13})$, compute X^{23}_{14}, $\Delta X^{25}_{11} =^{23}_{14}$, X^{25}_2, X^{25}_9, and X^{25}_8. Then, store $RK^{22}_7 = f_5(K_0,K_2,K_5,K_7,K_9, K_{10},K_{11},K_{12},K_{13})$, $RK^{23}_5 = f_6(K_0,K_2,K_5,K_7,K_9,K_{10}, K_{11},K_{12},K_{13})$, and $RK^{24}_4 = f_7(K_0,K_2,K_5,K_9,K_{10},K_{12},K_{13})$ in H_{11} indexed by X^{23}_{14}, $\Delta X^{25}_{11} =^{23}_{14}$, X^{25}_2, X^{25}_9, X^{25}_8, t^{22}_0, ΔT_7, t^{23}_1, and t^{24}_2. ΔX^{25}_{11} is chosen such that $\Delta X^{25}_{11} \in S[\Delta T_7]$, see Observation 4. Therefore, H_{11} has 7×2^{32} rows and on average about $2^{44}/(7 \times 2^{32}) = (16/7) \times 2^8$ values in each row.

Our attack proceeds as follows. We initialize an array H of $2^{18\times4=72}$ entries to "0", where each entry is 1-bit and the index of the array is 18 key nibbles involved in the attack, as we will see later. Then, we generate 2^m structures as described above. Therefore, we have 2^{m+63} pairs of plaintext/ciphertext pairs generated using 2^{m+28} chosen plaintexts. Next, we ask the encryption oracle for their corresponding ciphertexts. The plaintext/ciphertext pairs that satisfy Observation 4 are $2^{m+63} \times 2^{-13.119} = 2^{m+49.881}$ pairs; and after the ciphertext filtration, we have only $2^{m+49.881} \times 2^{-12\times4} = 2^{m+1.881}$ remaining pairs. For each remaining pair, we perform the following steps:

1. Determine the number of possible values of $RK^0_0 = K_1$ that satisfy the path by accessing H_1. Therefore, we have $(16/7)$ possible values for K_1.
2. Determine the number of possible values of $RK^0_5 = K_{14}$ that satisfy the path by accessing H_2. Therefore, we have $(16/7)^2$ possible values for K_1, K_{14}.

3. Determine the number of possible values of $RK_7^0 = K_{16}$ that satisfy the path by accessing H_3. Therefore, we have $(16/7)^3$ possible values for K_1, K_{14}, K_{16}.
4. Determine the number of possible values of $RK_2^1 = K_8, RK_1^0 = K_3$ that satisfy the path by accessing H_4. Therefore, we have $2^4 \times (16/7)^4$ possible values for $K_1, K_3, K_8, K_{14}, K_{16}$.
5. Determine the number of possible values of $RK_5^1 = K_{18}, RK_6^0 = K_{15}$ that satisfy the path by accessing H_5. Therefore, we have $2^8 \times (16/7)^5$ possible values for $K_1, K_3, K_8, K_{14}, K_{15}, K_{16}, K_{18}$.
6. Determine the number of possible values of $RK_3^2 = K_{14}, RK_4^1 = K_{17}, RK_3^0 = K_6$ that satisfy the path by accessing H_6. Therefore, we have $2^{12} \times (16/7)^6$ possible values for $K_1, K_3, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}$.
7. Determine the number of possible values of $RK_1^3 = K_{15}, RK_5^2 = K_3, RK_6^1 = K_{19}, RK_2^0 = K_4$ that satisfy the path by accessing H_7. Therefore, we have $2^{16} \times (16/7)^7$ possible values for $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}$.
8. Determine the number of possible values of RK_0^{24} that satisfy the path by accessing H_8. Therefore, we have $2^{16} \times (16/7)^8$ possible values for $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, RK_0^{24}$.
9. Determine the number of possible values of RK_7^{24} that satisfy the path by accessing H_9. Therefore, we have $2^{16} \times (16/7)^9$ possible values for $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, RK_0^{24}, RK_7^{24}$.
10. Determine the number of possible values of RK_7^{23}, RK_5^{24} that satisfy the path by accessing H_{10}. Therefore, we have $2^{20} \times (16/7)^{10}$ possible values for $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, RK_0^{24}, RK_7^{24}, RK_7^{23}, RK_5^{24}$.
11. Determine the number of possible values of $RK_7^{22}, RK_5^{23}, RK_4^{24}$ that satisfy the path by accessing H_{11}. Therefore, we have $2^{28} \times (16/7)^{11}$ possible values for $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, RK_0^{24}, RK_7^{24}, RK_7^{23}, RK_5^{24}, RK_7^{22}, RK_5^{23}, RK_4^{24}$.
12. The deduced $2^{28} \times (16/7)^{11}$ values for 18 key nibbles, $K_1, K_3, K_4, K_6, K_8, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}, K_{19}, RK_0^{24}, RK_7^{24}, RK_7^{23}, RK_5^{24}, RK_7^{22}, RK_5^{23}, RK_4^{24}$, involved in the attack are wrong keys. Hence, mark them in H invalid "1".

Attack Complexity. As depicted in Fig. 4, we have 22 round keys involved in the analysis rounds. According to the key schedule, these 22 round keys take only 2^{72} possible values (see step 12 in Table 4). As mentioned in step 12, we remove on average $2^{28} \times (16/7)^{11} = 2^{41.119}$ out of 2^{72} possible values of these 22 round keys involved in the attack, for each one message pair of the $2^{m+1.881}$ remaining pairs. Hence, a wrong key is not discarded with one pair with probability $1 - 2^{41.119-72} = 1 - 2^{-30.881}$. Therefore, we have $2^{72} \times (1 - 2^{-30.881})^{2^{m+1.881}} \approx 2^{72} \times (e^{-1})^{2^{m+1.881-30.881}} \approx 2^{72} \times 2^{-1.4 \times 2^{m-29}}$ remaining candidates for 72-bit of the key, after processing all the $2^{m+1.881}$ remaining pairs. We evaluated the computational complexity of the attack as a function of m, as illustrated in Table 4, to determine the optimal value of m that leads to the best computational complexity. As steps 11 and 12 dominate the time complexity of the attack, see Table 4, we choose $m = 33.5$ in order to optimize the time complexity of the attack. Therefore, we have $2^{72} \times 2^{-1.4 \times 2^{33.5-29=4.5}} = 2^{72-31.678} = 2^{40.322}$

remaining candidates for 72-bit of the key. These 72-bit of the key include 11 master key nibbles and 7 round key nibbles. To, retrieve the whole master key, perform the following steps:

1. Retrieve K_{10} from RK_4^{24} by guessing the 6 key nibbles $K_0, K_2, K_5, K_9, K_{12}$, K_{13}. Since this step includes 18 S-box operations, it requires $2^{40.322+24=64.322} \times \frac{18}{8\times 25} \approx 2^{60.848}$ encryptions. Since RK_4^{24} and RK_0^{24} are functions in the same nibbles of the master key, we can compute RK_0^{24} using the retrieved K_{10} and then match the computed value with its value in the remaining candidate key. As a result, we have 4-bit filtration. Hence, we have only $2^{40.322+24-4=60.322}$ 72-bit remaining key candidates. This step requires $2^{40.322+24=64.322} \times \frac{37}{8\times 25} \approx 2^{61.888}$ encryptions.
2. Using the same technique, retrieve K_7 from RK_5^{23} by guessing K_{11}. This step requires $2^{60.322+4=64.322} \times \frac{90}{8\times 25} \approx 2^{63.167}$ encryptions. Since RK_5^{24} is also a function in the same nibbles of the master key, we can compute it using the retrieved K_7 and compare it with its value in the remaining candidate. As a result, we have 4-bit filtration. Hence, we have only $2^{60.322+4-4=60.322}$ 80-bit remaining key candidates. This step requires $2^{60.322+4=64.322} \times \frac{112}{8\times 25} \approx 2^{63.485}$. Then, we perform the previous filtration to the following round key nibbles RK_7^{22}, RK_7^{23}, and RK_7^{24}. Finally, we have another 3 4-bit filtrations. Therefore, we have only $2^{60.322-12} = 2^{48.322}$ remaining candidates for the whole master key. The time complexity of this step is dominated by $2^{64.335}$ encryptions.

The right master key can be retrieved by exhaustively searching the $2^{48.322}$ remaining key candidates using 2 plaintext/ciphertext pairs. This step requires $2 \times 2^{48.322} = 2^{49.322}$ encryptions. Therefore, the time complexity of the attack is dominated by steps 11 and 12 in Table 4 which requires $2^{70.441} + 2^{68.856} \approx 2^{70.86}$ encryptions, see Table 4. The data complexity of the attack is $2^{m+28} = 2^{33.5+28} = 2^{61.5}$ chosen plaintexts. The memory complexity of the attack is dominated by the memory that is required to store H. Hence, the memory complexity is $2^{72} \times 2^{-6} = 2^{66}$ 64-bit blocks.

6 Conclusion

In this work, we presented two impossible differential attacks against reduced-round versions of T-TWINE. Both attacks use our proposed 18-round impossible differential distinguisher. To the best of our knowledge, this distinguisher is the first valid 18-round distinguisher. Utilizing this distinguisher, we launched 25-round and 27-round attacks on T-WINE-80 and T-TWINE-128, respectively. The presented attacks are the first published attacks against both versions of T-TWINE.

References

1. Avanzi, R. The qarma block cipher family. almost mds matrices over rings with zero divisors, nearly symmetric even-mansour constructions with non-involutory

Table 4. Time complexity of the different steps of the attack on 25-round T-TWINE-80, where NK denotes the number of keys to be excluded.

Step	Time Complexity (in 25-round encryptions)	NK	$m = 33.5$
1	$2^{m+1.881} \times (16/7) \times \frac{3}{8 \times 25} \approx 2^{m-2.985}$	$(16/7)$	$2^{30.515}$
2	$2^{m+1.881} \times (16/7)^2 \times \frac{3}{8 \times 25} \approx 2^{m-1.793}$	$(16/7)^2$	$2^{31.707}$
3	$2^{m+1.881} \times (16/7)^3 \times \frac{4}{8 \times 25} \approx 2^{m-0.185}$	$(16/7)^3$	$2^{33.315}$
4	$2^{m+1.881} \times 2^4 \times (16/7)^4 \times \frac{4}{8 \times 25} \approx 2^{m+5.008}$	$2^4 \times (16/7)^4$	$2^{38.508}$
5	$2^{m+1.881} \times 2^8 \times (16/7)^5 \times \frac{2}{8 \times 25} \approx 2^{m+9.200}$	$2^8 \times (16/7)^5$	$2^{42.700}$
6	$2^{m+1.881} \times 2^{12} \times (16/7)^6 \times \frac{5}{8 \times 25} \approx 2^{m+15.715}$	$2^{12} \times (16/7)^6$	$2^{49.215}$
7	$2^{m+1.881} \times 2^{16} \times (16/7)^7 \times \frac{4}{8 \times 25} \approx 2^{m+20.586}$	$2^{16} \times (16/7)^7$	$2^{54.086}$
8	$2^{m+1.881} \times 2^{16} \times (16/7)^8 \times \frac{1}{8 \times 25} \approx 2^{m+19.778}$	$2^{16} \times (16/7)^8$	$2^{53.278}$
9	$2^{m+1.881} \times 2^{16} \times (16/7)^9 \times \frac{3}{8 \times 25} \approx 2^{m+22.556}$	$2^{16} \times (16/7)^9$	$2^{56.056}$
10	$2^{m+1.881} \times 2^{20} \times (16/7)^{10} \times \frac{4}{8 \times 25} \approx 2^{m+28.164}$	$2^{20} \times (16/7)^{11}$	$2^{61.664}$
11	$2^{m+1.881} \times 2^{28} \times (16/7)^{11} \times \frac{3}{8 \times 25} \approx 2^{m+36.941}$	$2^{28} \times (16/7)^{11}$	$2^{70.441}$
12	$2^{m+1.881} \times 2^{28} \times (16/7)^{11} \times \frac{1}{8 \times 25} \approx 2^{m+35.356}$	$2^{28} \times (16/7)^{11}$	$2^{68.856}$

central rounds, and search heuristics for low-latency s-boxes. *IACR Transactions on Symmetric Cryptology 2017*, 1 (Mar. 2017), 4–44.

2. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., and Sim, S. M. The skinny family of block ciphers and its low-latency variant mantis. In *Advances in Cryptology – CRYPTO 2016* (Berlin, Heidelberg, 2016), M. Robshaw and J. Katz, Eds., Springer Berlin Heidelberg, pp. 123–153.
3. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., and Sim, S. M. The skinny family of block ciphers and its low-latency variant mantis. In *Advances in Cryptology – CRYPTO 2016*, M. Robshaw and J. Katz, Eds. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016, pp. 123–153.
4. Biham, E., Biryukov, A., and Shamir, A. Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In *Advances in Cryptology —- EUROCRYPT '99*, J. Stern, Ed., vol. 1592 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1999, pp. 12–23.
5. Ferguson, N., Lucks, S., Schneier, B., Whiting, D., Bellare, M., Kohno, T., Callas, J., and Walker, J. The SKEIN hash function family , 2010. http://www.skeinhash.info.
6. Goldenberg, D., Hohenberger, S., Liskov, M., Schwartz, E. C., and Seyalioglu, H. On tweaking luby-rackoff blockciphers. In *Advances in Cryptology – ASIACRYPT 2007* (Berlin, Heidelberg, 2007), K. Kurosawa, Ed., Springer Berlin Heidelberg, pp. 342–356.

7. Jean, J., Nikolić, I., and Peyrin, T. Tweaks and keys for block ciphers: The tweakey framework. In *Advances in Cryptology – ASIACRYPT 2014* (Berlin, Heidelberg, 2014), P. Sarkar and T. Iwata, Eds., Springer Berlin Heidelberg, pp. 274–288.
8. Jean, J., Nikolić, I., Peyrin, T., and Seurin, Y. Deoxys v1.41. Submitted to CAESAR Competition, 2016. https://competitions.cr.yp.to/round3/deoxysv141.pdf.
9. Knudsen, L. DEAL: A 128-bit block cipher. *Complexity 258*, 2 (1998). NIST AES Proposal.
10. Lampe, R., and Seurin, Y. Tweakable blockciphers with asymptotically optimal security. In *Fast Software Encryption* (Berlin, Heidelberg, 2014), S. Moriai, Ed., Springer Berlin Heidelberg, pp. 133–151.
11. Landecker, W., Shrimpton, T., and Terashima, R. S. Tweakable blockciphers with beyond birthday-bound security. In *Advances in Cryptology – CRYPTO 2012* (Berlin, Heidelberg, 2012), R. Safavi-Naini and R. Canetti, Eds., Springer Berlin Heidelberg, pp. 14–30.
12. Liskov, M., Rivest, R. L., and Wagner, D. Tweakable block ciphers. *Journal of Cryptology 24*, 3 (2010), 588–613.
13. Mitsuda, A., and Iwata, T. Tweakable pseudorandom permutation from generalized feistel structure. In *Provable Security* (Berlin, Heidelberg, 2008), J. Baek, F. Bao, K. Chen, and X. Lai, Eds., Springer Berlin Heidelberg, pp. 22–37.
14. Nyberg, K. Generalized feistel networks. In *Advances in Cryptology — ASIACRYPT '96* (Berlin, Heidelberg, 1996), K. Kim and T. Matsumoto, Eds., Springer Berlin Heidelberg, pp. 91–104.
15. Rogaway, P. Efficient instantiations of tweakable blockciphers and refinements to modes ocb and pmac. In *Advances in Cryptology - ASIACRYPT 2004* (Berlin, Heidelberg, 2004), P. J. Lee, Ed., Springer Berlin Heidelberg, pp. 16–31.
16. Sakamoto, K., Minematsu, K., Shibata, N., Shigeri, M., Kubo, H., Funabiki, Y., Bogdanov, A., Morioka, S., and Isobe, T. Tweakable twine: Building a tweakable block cipher on generalized feistel structure. In *Advances in Information and Computer Security* (Cham, 2019), N. Attrapadung and T. Yagi, Eds., Springer International Publishing, pp. 129–145.
17. Schroeppel, R. An overview of the hasty pudding cipher , 1998. http://www.cs.arizona.edu/rcs/hpc.
18. Suzaki, T., and Minematsu, K. Improving the generalized feistel. In *Fast Software Encryption* (Berlin, Heidelberg, 2010), S. Hong and T. Iwata, Eds., Springer Berlin Heidelberg, pp. 19–39.
19. Suzaki, T., Minematsu, K., Morioka, S., , and Kobayashi, E. TWINE: A Lightweight, Versatile Block Cipher. In *ECRYPT Workshop on Lightweight Cryptography* (Belgium, 2011), pp. 28–29.
20. Suzaki, T., Minematsu, K., Morioka, S., and Kobayashi, E. twine: A lightweight block cipher for multiple platforms. In *Selected Areas in Cryptography*, L. Knudsen and H. Wu, Eds., vol. 7707 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 339–354.
21. Suzaki, T., Minematsu, K., Morioka, S., and Kobayashi, E. TWINE: A Lightweight Block Cipher for Multiple Platforms. In *Selected Areas in Cryptography* (Berlin, Heidelberg, 2013), L. R. Knudsen and H. Wu, Eds., Springer Berlin Heidelberg, pp. 339–354.
22. Zheng, X., and Jia, K. Impossible differential attack on reduced-round twine. In *Information Security and Cryptology – ICISC 2013* (Cham, 2014), H.-S. Lee and D.-G. Han, Eds., Springer International Publishing, pp. 123–143.

23. ZHENG, Y., MATSUMOTO, T., AND IMAI, H. Impossibility and optimality results on constructing pseudorandom permutations. In *Advances in Cryptology — EUROCRYPT '89* (Berlin, Heidelberg, 1990), J.-J. Quisquater and J. Vandewalle, Eds., Springer Berlin Heidelberg, pp. 412–422.

A 18-round impossible differential characteristic as depicted in Figure 8 of [16]

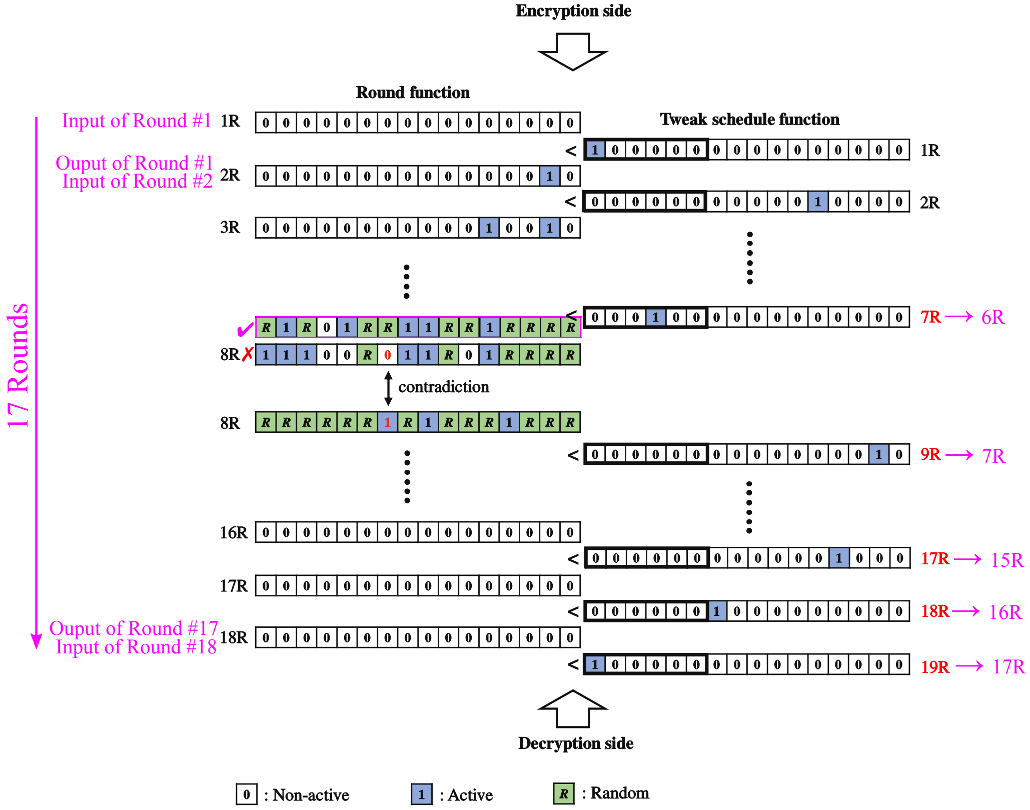
Encryption side
Round function
Tweak schedule function
Input of Round #1
1R 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
< 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1R
Ouput of Round #1
Input of Round #2
2R 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
< 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 2R
3R 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0
✓ R 1 R 0 1 R R 1 1 R R 1 R R R R
< 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 7R ⟶ 6R
8R ✗ 1 1 1 0 0 R 0 1 1 R 0 1 R R R R
contradiction
8R R R R R R R 1 R 1 R R R 1 R R R
< 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 9R ⟶ 7R
17 Rounds
16R 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
< 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 17R ⟶ 15R
17R 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
< 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 18R ⟶ 16R
Ouput of Round #17
Input of Round #18
18R 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
< 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 19R ⟶ 17R
Decryption side
0 : Non-active
1 : Active
R : Random

Fig. 5. 18-round impossible differential characteristic as depicted in Figure 8 of [16] with our comments.