Research Article

Related-Key Multiple Impossible Differential Cryptanalysis on Full-Round LiCi-2 Designed for IoT

Kai Zhang,1,2 Xuejia Lai,1 Lei Wang,1 Jie Guan,2 Bin Hu,2 Senpeng Wang,2 and Tairong Shi2

1Shanghai Jiao Tong University, Shanghai 201100, China
2PLA SSF Information Engineering University, Zhengzhou 450000, China

Correspondence should be addressed to Kai Zhang; zhkai2010@139.com and Xuejia Lai; laix@sjtu.edu.cn

Received 15 November 2021; Revised 12 March 2022; Accepted 22 March 2022; Published 25 May 2022

Abstract

LiCi-2 is an ultralightweight block cipher designed for constrained IoT devices. It is a successor of LiCi and has even better performance in both software and hardware implementation. In this paper, based on the idea of related-key multiple impossible differential cryptanalysis, a key recovery attack on full-round LiCi-2 is proposed. First, an interesting property is revealed that, with a single bit difference in the related key, a 10-round differential character with probability of 1 exists on LiCi-2. With an automatic approach, the boundaries of impossible differential distinguishers in terms of single-key setting and related-key setting are explored. Under our construction method, the longest length is 8 rounds for single-key setting and 18 rounds for related-key setting. Finally, based on these 18-round distinguishers, a 25-round key recovery attack is proposed with adding 3 rounds before and 4 rounds after the distinguisher. Our attack needs one related key. The time complexity for our attack is $O(2^{123.44})$, the memory complexity is $O(2^{94})$, and the data complexity is $O(2^{60.68})$. As far as we know, no full-round attack has previously been reported on LiCi-2.

1. Introduction

As a representative for the new information age, the Internet of things (IoT) has shown its strong vitality in many fields. It can be viewed as an expanded network based on the Internet and it combines various sensing devices to form a huge network, which enables a wider interconnection of people, machines, and things. However, the sensing devices in IoT usually have very limited resources such as computing resource, power resource, and hardware implementation resource. So, it is very essential to propose lightweight primitives and ensure the information security in IoT devices. In recent years, many good lightweight block ciphers are proposed such as PRESENT [1], GIFT [2], Midori [3], LBlock [4], RECTANGLE [5], SIMON, and SPECK [6]. In 2019, NIST (National Institute of Standards and Technology) proposed a standardization project LWC (LightWeight Cryptography) to enhance the development for lightweight ciphers. Among these lightweight block ciphers, LiCi (denoted as LiCi-1 in this paper to avoid confusion) is a lightweight block cipher proposed in 2017 [7]. The block size for LiCi-1 is 64-bit and the key size is 128-bit. It totally has 31 rounds. Up to now, the security margin on LiCi-1 is enough. To reach better performance, as a successor, LiCi-2 was proposed in 2018 [8]. For LiCi-2, it reduces the total rounds from 31 to 25, and half of the subkeys and a circular shift are omitted for better performance. LiCi-2 is designed for extremely constrained devices, such as 6LoWPAN. It shows good performance in memory requirement, power consumption, and hardware implementation.

Impossible differential cryptanalysis was originally proposed by Knudsen [9] and Biham et al. [10], respectively. It is one of the most effective cryptanalytic methods so far. The basic idea of impossible differential cryptanalysis is establishing an impossible differential distinguisher and filtering the wrong key candidates with this distinguisher until the correct key is recovered. Related-key impossible differential cryptanalysis is a variant of impossible differential
differential cryptanalysis. This attack is based on the assumption that the attacker has access to the encryption oracle and can use related keys as input. This kind of attack has been applied to many block ciphers successfully, such as AES and LBlock [11–15].

Our Contributions: The main purpose of this paper is to evaluate the security level on LiCi-2 against related-key multiple impossible differential cryptanalysis. There are three main contributions which are listed as follows:

(i) A 10-round differential distinguisher with probability of 1 is constructed. With this property, many differential-type distinguishers can be extended for better attack.

(ii) The length and number of impossible differential distinguishers on LiCi-2 in single-key setting and related-key setting are explored in an automatic approach. In the single-key setting, the longest length of the distinguisher under our construction method is 9-round and there are altogether 240 such distinguishers discovered. In the related-key setting, the longest length of the distinguishers discovered is 18-round and 65 such distinguishers are presented.

(iii) Based on twenty of these 18-round related-key distinguishers, a key recovery attack on full-round LiCi-2 is proposed with adding 3 rounds before and 4 rounds after the distinguisher. The summary of our result and prior results on LiCi-1 and LiCi-2 is presented in Table 1.

This paper is organized as follows. Section 2 introduces the notations used throughout this paper. Section 3 gives a brief description on LiCi-2. Section 4 presents some observations and properties on LiCi-2. In Section 5, a key recovery attack on LiCi-2 is proposed, and Section 6 concludes the paper.

### Table 1: Summary of previous results on LiCi-1 and LiCi-2.

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Type of attack</th>
<th>Attacked/total rounds</th>
<th>Length of the distinguisher</th>
<th>Time complexity</th>
<th>Memory complexity</th>
<th>Data complexity</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>LiCi-1</td>
<td>Linear</td>
<td>16/32</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$2^{106}$</td>
<td>[7]</td>
</tr>
<tr>
<td></td>
<td>Differential</td>
<td>16/32</td>
<td>—</td>
<td>$2^{83}$</td>
<td>$2^{41}$</td>
<td>$2^{63}$</td>
<td>[11]</td>
</tr>
<tr>
<td></td>
<td>Integral</td>
<td>13/32</td>
<td>12</td>
<td>$2^{173}$</td>
<td>$2^{19}$</td>
<td>$2^{63.6}$</td>
<td>[16]</td>
</tr>
<tr>
<td>LiCi-1</td>
<td>Impossible</td>
<td>16/32</td>
<td>10</td>
<td>$2^{83.08}$</td>
<td>$2^{76.76}$</td>
<td>$2^{59.76}$</td>
<td>[17]</td>
</tr>
<tr>
<td>Differential</td>
<td>Linear</td>
<td>20/25</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>$2^{106}$</td>
<td>[8]</td>
</tr>
<tr>
<td>LiCi-2</td>
<td>Impossible</td>
<td>25/25</td>
<td>18</td>
<td>$2^{133.44}$</td>
<td>$2^{94}$</td>
<td>$2^{60.68}$</td>
<td>This paper</td>
</tr>
</tbody>
</table>

2. Notations

Suppose that $E$ is an $n$-bit block cipher of $R$ rounds; the following notations are used throughout this paper:

(i) $K$: 128-bit master key, $K^{t}$ represents the value of the 128-bit key register at the $t$-th round and $K_{[i-j]}$ represents the $i$-th bit to the $j$-th bit of $K$

(ii) $RK^{t}$: 32-bit round key at the $i$-th round

### Table 2: Sbox in LiCi-2.

<table>
<thead>
<tr>
<th>$x$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>S($x$)</td>
<td>3</td>
<td>F</td>
<td>E</td>
<td>1</td>
<td>0</td>
<td>A</td>
<td>5</td>
<td>8</td>
<td>C</td>
<td>4</td>
<td>B</td>
<td>2</td>
<td>9</td>
<td>7</td>
<td>6</td>
</tr>
</tbody>
</table>

(iii) $(L^{t}, R^{t})$: input for the $i$-th round, $L^{t}$ represents the left branch, $R^{t}$ represents the right branch, and $L_{[i-j]}(R_{[i-j]})$ represents the $i$-th bit to the $j$-th bit of $L^{t}(R^{t})$

(iv) $S^{i}$: output of the Sbox layer for the $i$-th round

(v) "<<<i": left rotation with $i$ bits

(vi) "⊕": XOR operation

3. Brief Description on LiCi-2

LiCi-2 is a lightweight block cipher that was proposed in 2018 [8]. It adopts a Feistel-type structure. The block size for LiCi-2 is 64-bit and the key size is 128-bit. It totally has 25 rounds. Each round consists of Sbox constitution, key addition, and circular shift. The round function is depicted in Figure 1 and the 4-bit Sbox used in LiCi-2 is depicted in Table 2.
Key Schedule: The scale of master key $K$ is 128-bit. It can be denoted as $K = k_{127}, k_{126}, k_{125}, \ldots, k_2, k_1, k_0$. First, the 128-bit key register is initialized with the master key. The register is rotated by 13 bits on the left. Every four of the eight least significant bits of the key register are substituted with the Sbox. XOR the binary form of the round number to the bits $[k_{63} k_{62} k_{61} k_{60} k_{59}]$. The 32 least significant bits are extracted as the round key $R K^1$. The mathematical form of this updating progress can be illustrated as follows:

1. $[k_{127} k_{126} \ldots k_1 k_0] << < 13$
2. $[k_5 k_3 k_1 k_0] = S[k_5 k_3 k_1 k_0]$
3. $[k_7 k_5 k_3 k_1] = S[k_7 k_5 k_3 k_1]$
4. $[k_9 k_7 k_5 k_3 k_1 k_0 k_{59}] = [k_9 k_7 k_5 k_3 k_1 k_0 k_{59}] \oplus R C^2$
5. $R K^1 = [k_3 k_30 \ldots k_1 k_0]$

4. Observations on LiCi-2

In this section, some new observations and properties on LiCi-2 are introduced. First, a 10-round differential character with probability one is constructed in Property 1. The boundaries of impossible differential distinguishers on LiCi-2 in single-key setting and related-key setting are discussed in Observation 1 and Observation 2. These distinguishers will be a foundation for our later attack.

Property 1 (a 10-Round Differential Character). Given two plaintext pairs whose difference is $(\Delta L^0, \Delta R^0)$ and a related key whose difference is $\Delta K$, if $\Delta R^0_{[21]} = 1$, $\Delta K_{[21]} = 1$, and other differences are all zeros, the difference of the internal state at the beginning of the 11th round $(\Delta L^{10}, \Delta R^{10})$ will have only two-bit difference $\Delta L^{10}_{[10]} = 1$, $\Delta R^{10}_{[10]} = 1$ with probability of 1.

Proof. The difference of the first subkey $\Delta R^0$ can be derived from $\Delta K$, where $\Delta R^0_{[1]} = 1$ and other bits are all zeros. According to the round function, the difference of the input of the second round can be calculated as follows:

\[
\begin{align*}
\Delta L^1 &= \Delta R^0 \oplus \Delta R^0 \oplus (\Delta S(\Delta L^0) << << 11), \\
\Delta R^1 &= (\Delta S(\Delta L^0) << << 11) \oplus \Delta L^1.
\end{align*}
\]

As $\Delta L^0 = O$ and $\Delta R^0 = \Delta R^0$, we can derive $(\Delta L^1, \Delta R^1) = (0, 0)$. So, after several consecutive rounds, the difference of the internal state will remain zero until some $\Delta R^1$ bring in a new nonzero difference.

Let us consider the differential transfer property of the key schedule. According to the key schedule, the single difference of the master key will go through the following positions of the key register (Table 3) at the first 10 rounds.

After analyzing these positions, the nonzero difference will not appear in the subkeys until the 10th round, that is, $\Delta R^2_{[10]}$. So the differences of $(\Delta L^1, \Delta R^1)$, $(\Delta L^2, \Delta R^2)$, \ldots, $(\Delta L^9, \Delta R^9)$ will all be zeros and, for $(\Delta L^{10}, \Delta R^{10})$, $\Delta R^{10}_{[10]} = 1$, $\Delta R^{10}_{[10]} = 1$ due to the difference of $\Delta R^2_{[10]}$.

Observation 1 (Boundary of Single-Key Impossible Differential Distinguishers). For LiCi-2, in single-key setting, considering the details of the Sbox in Table 4, the maximum length of impossible differential distinguisher is 8 rounds and there are 544 such distinguishers in terms of one active bit input and output difference. If the numbers of active bits for the input and output difference are both limited to 2, the maximum length of single-key impossible differential distinguisher is 9 rounds and there are 240 such 9-round distinguishers. Summary of the statistics for these distinguishers is illustrated in Table 5.

As predicted by the designers, this new version of LiCi has better diffusion property. The result in Observation 1 validates this improvement. For the original LiCi-1, for single-bit difference, the maximum length can reach 10 rounds and, for 1-bit input and 2-bit output difference, the maximum length can reach 11 rounds.

Observation 2 (Boundary of Related-Key Impossible Differential Distinguishers). For LiCi-2, in related-key setting, considering the details of the Sbox in Table 4, the maximum length of impossible differential distinguisher is 17 rounds and there are 64 such distinguishers in terms of one active bit in input, output, and key difference. If the number of active bits is 1 for input and key and 2 for output, the maximum length of related-key impossible differential distinguisher can reach 18 rounds and there are 65 such distinguishers (all these 65 distinguishers are presented in the Appendix). The summary of all these distinguishers is illustrated in Table 6.
According to Property 1, there are some long related-key differential characters. Thus, we added the condition of related key to our automatic searching algorithm and the result coincides with Property 1. The distinguishers are stretched for about 10 rounds as expected. Through Observation 2, it is found that, due to the existence of Property 1, many long related-key impossible differential distinguishers exist for LiCi-2 and the longest length under our searching strategy can cover about 72% of the total rounds.

For LiCi-1, a single 10-round impossible differential distinguisher is reported in [11], the hamming weight for the input difference is one, and output difference is two. According to our research on LiCi-2, there are two interesting contrary observations:

1. The security level for single-key impossible differential cryptanalysis on LiCi-2 is higher than that on LiCi-1

2. In related-key setting, the security level on LiCi-2 is much worse than that on LiCi-1

Property 2 (an 18-Round Distinguisher). For LiCi-2 reduced to 18 rounds, the differences for the plaintexts, ciphertexts, and related keys are denoted as \( (\Delta L^0, \Delta R^0), (\Delta L^{18}, \Delta R^{18}) \), and \( \Delta K \) respectively. If \( \Delta R^0 = 1, \Delta K^{[45]} = 1, \Delta R^{18} = 1 \), and other differences are all zeros, it is an 18-round impossible differential distinguisher.

\begin{proof}

The difference of the master key \( \Delta K^{[45]} = 1 \) will lead to the following differences for the 18-round key registers:

\[
\begin{align*}
\Delta K_0^{[45]} &= 1, \Delta K_1^{[58]} = 1, \Delta K_2^{[71]} = 1, \Delta K_3^{[84]} = 1, \Delta K_4^{[87]} = 1, \\
\Delta K_5^{[110]} &= 1, \Delta K_6^{[123]} = 1, \Delta K_7^{[8]} = 1, \Delta K_8^{[21]} = 1, \Delta K_9^{[34]} = 1, \Delta K_{10}^{[47]} = 1, \Delta K_{11}^{[60]} = 1, \Delta K_{12}^{[73]} = 1, \Delta K_{13}^{[86]} = 1, \Delta K_{14}^{[99]} = 1, \\
\Delta K_{15}^{[112]} &= 1, \Delta K_{16}^{[125]} = 1, \Delta K_{17}^{[10]} = 1.
\end{align*}
\]

According to the key schedule, the differences for the round keys are \( \Delta R_k^{[8]} = 1, \Delta R_k^{[21]} = 1, \Delta R_k^{[10]} = 1 \), and other differences are all zeros.

As \( (\Delta L^0, \Delta R^0) = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \), according to the round function, the difference of the output of the first round can be calculated as follows:

\[
\begin{align*}
\Delta L^1 &= \Delta R^0 \oplus \Delta R^{10} \oplus (\Delta S(\Delta L^0) \ll 11), \\
\Delta R^1 &= (\Delta S(\Delta L^0) \ll 11) \oplus \Delta L^1.
\end{align*}
\]

So \( \Delta L^1 = \Delta R^1 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \).

For \( S_{3-0}^3 \), as the difference of the input is \((0,0,0,0,1)\), according to Table 4, the difference of the output is of the form \((1,*,*,*)\). After calculation according to the round function, the form of the difference for \( (\Delta L^2, \Delta R^2) \) is as follows:

\[
\begin{align*}
\Delta L^2 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta R^2 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0).
\end{align*}
\]

Similarly, the input differences for the 4th round and 5th round are as follows:

\[
\begin{align*}
\Delta L^3 &= (0, 0, 0, 0, 0, *, *, *, *, *, *, 0, 0, 0, 0, 1, *, *, *, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1), \\
\Delta R^3 &= (0, 0, 0, 0, 0, *, *, *, *, *, *, *, 0, 0, 0, 0, 0, *, *, *, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1), \\
\Delta L^4 &= (*, *, *, *, *, *, *, *, *, *, *, *, *, 0, 0, 0, 0, 1, *, *, *, 0, 0, 0, 0, *, *, *, *, *, *, *, *, *, *, *, *, *, *), \\
\Delta R^4 &= (*, *, *, *, *, *, *, *, *, *, *, *, *, 0, 0, 0, 0, 0, *, *, *, *, *, *, *, *, *, *, *, *, *, *, *, *).
\end{align*}
\]
In the direction of decryption, the output difference \((\Delta L^{18}, \Delta R^{18})\) is expressed as follows:

\[
\begin{align*}
\Delta L^{18} &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta R^{18} &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0).
\end{align*}
\]

According to the key schedule, the round key \(RK^{17}\) has a single-bit difference in \(RK^{17}_{[10]}\). After one-round decryption, the difference for the input of the 17th round is \(\Delta L^{17} = 0, \Delta R^{17} = 0\). As the differences for subkey \(RK^{16}\) to \(RK^9\) are all zeros, the difference for the input of the 9th round \(\Delta L^9 = 0, \Delta R^9 = 0\).

\[
\begin{align*}
\Delta L^7 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta R^7 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta L^6 &= (\ast, \ast, \ast, \ast, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta R^6 &= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta L^5 &= (\ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta R^5 &= (\ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0), \\
\Delta L^4 &= (\ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast), \\
\Delta R^4 &= (\ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast, \ast).
\end{align*}
\]

The subkey \(RK^8\) will bring in a new difference in \(RK^8_{[21]}\); thus the difference for the input of the 8th round is \(\Delta L^8 = 0, \Delta R^8 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0).\)

Similarly, the input differences for the 7th to 4th rounds calculated in the decryption direction are as follows:

After comparing the two differences of \((\Delta L^4, \Delta R^4)\) derived in the encryption and decryption direction, there exists a contradiction in the position of \(\Delta R^8_{[8]}\) (in the encryption direction, the difference \(\Delta R^8_{[8]}\) is zero, while in the decryption direction, the difference \(\Delta R^8_{[8]}\) is one, and the contradiction bit is marked in red in Figure 2). Thus, the constructed distinguisher is an 18-round impossible differential distinguisher. All the details for the construction of this 18-round distinguisher are depicted in Figure 2.

5. Key Recovery Attack on Full-Round LiCi-2

With the 18-round related-key impossible differential proposed in Property 2, we add three rounds before and four rounds after the distinguisher to launch a key recovery attack. There are two stages for the key recovery attack: data collection stage and key recovery stage.

Also, we use two approaches to reduce the data complexity. The first one is that more than one key is left in the candidate keys set and it will need to proceed to an exhaustive search among the remaining candidates; this strategy can be viewed as a tradeoff between the time complexity and data complexity. The second one is using multiple impossible differential distinguishers to reduce the data complexity. These two methods have been used and validated in ASIACRYPT 2014 [18] and Journal of Cryptology [19]. For simplicity, we will follow the framework proposed in [18].

5.1. Main Conclusions in [18]

5.1.1. Notations

(i) \(\Delta X (\Delta Y)\): input (output) difference of the impossible differential distinguisher
(ii) \(\Delta_{in}(\Delta_{out})\): set of all the possible input (output) differences of the cipher
(iii) \(r_{in}\): number of rounds of the differential path \((\Delta X, \Delta_{in})\)
(iv) \(r_{out}\): number of rounds of the differential path \((\Delta Y, \Delta_{out})\)
(v) \(c_{in}(c_{out})\): number of bit conditions that have to be verified to obtain \(\Delta X\) from \(\Delta_{in}\) \((\Delta Y\) from \(\Delta_{out}\))
(vi) \(k_{in}(k_{out})\): subset of subkey bits involved in the attack during the first \(r_{in}\) rounds (last \(r_{out}\) rounds)
(vii) \(|\Delta_{in}| (|\Delta_{out}|)\): the dimension of the input space (output space)
(viii) \(N\): amount of input (or output) pairs needed for the attack
(ix) \(P\): the probability that a guessed wrong key is kept in the candidate key set
The number of equivalent impossible differential distinguishers with different input differences and same output differences is denoted by $n_{\text{in}}$.

The number of equivalent impossible differential distinguishers with same input difference and different output differences is denoted by $n_{\text{out}}$.

Suppose that the block cipher we are dealing with is of block size $n$ and a key $K$ of size $|K|$.

5.1.2. Main Conclusions. The main conclusions are as follows.

(1) The relationship between $N$ and $P$ is as follows:

$$P = \left(1 - 2^{-\left(c_{\text{in}}c_{\text{out}}\right)}\right)^N < \frac{1}{2}.$$  \hfill (7)

The minimum value of $N$, denoted by $N_{\text{min}}$, is about $2^{n_{\text{in}}c_{\text{out}}}$.

(2) The time complexity of the attack is approximated by $T$:

$$T = \left(C_N + \left(2^{k_{\text{in}}+k_{\text{out}}} \cdot \frac{N}{2^{n_{\text{in}}c_{\text{out}}}} \cdot C_E + 2^{|K|} \cdot P\right)\right) \cdot C_D.$$  \hfill (8)
where $C_N = \max \left\{ \frac{\min_{x \in \{a_{in}, a_{out}\}} \sqrt{N \cdot 2^{|r| - |a_{in}|}}}{|a_{in}| \cdot N \cdot 2^{|o| - |a_{out}|}} \right\}$, $C_N$ represents the data complexity needed to construct $N$ plaintext-ciphertext pairs, with $N$ and $P$ satisfying condition (2) above, $C_E$ represents the cost for one full-round encryption, and $C_E^{C}$ represents the ratio of the cost for the partial encryption to the full encryption.

The memory complexity of the attack is determined by $N$.

(3) To apply multiple impossible differential cryptanalysis, suppose that $n_{in}, n_{out}$ impossible differential distinguishers are derived from a single distinguisher. With more distinguishers, the time complexity of the attack can also be calculated with equation (8); however, previous $|a_{in}|$ and $|a_{out}|$ are replaced with $|a_{in}^{'}|$ and $|a_{out}^{'}|$, where $|a_{in}^{'}| = |a_{in}| + \log_2 (n_{in})$ and $|a_{out}^{'}| = |a_{out}| + \log_2 (n_{out})$.

5.2. Procedures of the Attack

5.2.1. Data Collection Stage. On one hand, construct $2^m$ data sets. For each data set, to make the difference of a plaintext pair satisfy $(\Delta L^0, \Delta R^0)$ in Figure 3, there are 34 bits that can be arbitrary value and other 30 bits are fixed to a constant for a data set. On the other hand, suppose that the difference for the output of the 25th round is $(\Delta L^{25}, \Delta R^{25})$, $\Delta L^{25}_{[27,3]} = 1$, $\Delta R^{25}_{[27,3]} = 0$ (4 bits).

5.2.2. Key Recovery Stage. In the key recovery stage, we guess the subkey bits in its smallest unit to reduce the complexity. This process is depicted in Figure 3 to make it more intuitive. The details for the key recovery stage are illustrated in the following steps:

(i) Step 1. Sieve plaintext-ciphertext pairs according to the difference of the first round and last round. This process can be launched without the key.

(ii) Step 1.1. For each plaintext-ciphertext pair, sieve those pairs whose difference for the input and output of $\Delta S_{[11-8]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise. There are 3 bit conditions for this step.

(ii) Step 1.2. Sieve those pairs whose difference for the input and output of $\Delta S_{[11-8]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, 1, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (2 bit conditions).

(ii) Step 1.3. Sieve those pairs whose difference for the input and output of $\Delta S_{[15-12]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(ii) Step 1.4. Sieve those pairs whose difference for the input and output of $\Delta S_{[23-20]}^{0}$ satisfies $[\ast, \ast, \ast, 1, \ast, 0, 0, 1, 0]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iii) Step 1.5 Sieve those pairs whose difference for the input and output of $\Delta S_{[31-28]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iii) Step 1.6. Sieve the remaining pairs with $(\Delta L^1, \Delta R^1)$. If $\Delta R^1_{[15-8]} = 0$, keep the pair (8 bit conditions).

(iii) Step 1.7. Sieve those pairs whose difference for the input and output of $\Delta S_{[15-12]}^{24}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, \ast, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (1 bit condition).

(iii) Step 1.8. Sieve those pairs whose difference for the input and output of $\Delta S_{[23-20]}^{24}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, \ast, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (1 bit condition).

(iii) Step 1.9. Sieve those pairs whose difference for the input and output of $\Delta S_{[19-16]}^{24}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, \ast, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iii) Step 1.10. Sieve those pairs whose difference for the input and output of $\Delta S_{[23-20]}^{24}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, \ast, \ast, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (1 bit condition).

(iii) Step 1.11. Sieve those pairs whose difference for the input and output of $\Delta S_{[15-12]}^{24}$ satisfies $[1,0,0,0, 0, 0, 0, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iii) Step 1.12. Sieve the remaining pairs with $\Delta R^2$. If $\Delta R^2_{[26-24,19-15]} = 0$, $\Delta R^2_{[23]} = 1$, keep the pair (9 bit conditions).

(ii) Step 2. Sieve the plaintext-ciphertext pairs according to the difference of the second round.

(ii) Step 2.1. Guess $R^0_{[11-4]}$ and test whether the difference of $\Delta S_{[11-8]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 1, 0, 0, 0, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair (2 bit conditions).

(ii) Step 2.2. Guess $R^0_{[15-12]}$ and test whether the difference of $\Delta S_{[15-12]}^{0}$ satisfies $[\ast, \ast, \ast, \ast \ast, 0, 0, 0, 0, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair (3 bit conditions).

(ii) Step 2.3. Guess $R^0_{[23-20]}$ and test whether the difference of $\Delta S_{[23-20]}^{0}$ satisfies $[\ast, \ast, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, \ast]$. If the pairs can satisfy the condition, save the pair (3 bit conditions).

(ii) Step 2.4. Sieve the remaining pairs with $\Delta R^2$. If $\Delta R^2_{[23,22,20]} = 0$, keep the pair (3 bit conditions).
Figure 3: Key recovery attack on 25-round LiCi-2.

(iii) Step 3. Sieve the plaintext-ciphertext pairs according to the difference of the 23th round.

(i) Step 3.1. Guess $RK_{[14-11]}^{24}$ and test whether the difference for the input and output of $\Delta S_{[13-8]}^{2}$ satisfies $[*,*,* - *,*,*,*]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (1 bit condition).

(ii) Step 3.2. Guess $RK_{[22-19]}^{24}$ and test whether the difference for the input and output of $\Delta S_{[11-8]}^{23}$ satisfies $[0,1,0,0 - *,*,*,1]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iii) Step 3.3. Guess $RK_{[30-27]}^{24}$ and test whether the difference for the input and output of $\Delta S_{[19-16]}^{23}$ satisfies $[1,0,0,0 - *,*,*,1]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(iv) Step 3.4. Guess $RK_{[31-28]}^{24}$ and test whether the difference for the input and output of $\Delta S_{[21-20]}^{23}$ satisfies $[1,*,* - *,*,*,*]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (1 bit condition).

(v) Step 3.5. Guess $RK_{[24-21]}^{24}$ and test whether the difference for the input and output of $\Delta S_{[22-24]}^{23}$ satisfies $[1,0,0,0 - *,*,*,1]$. If the pairs can satisfy the condition, save the pair or discard it otherwise (3 bit conditions).

(vi) Step 3.6. Sieve the remaining pairs with $\Delta R^{25}$. If $\Delta R_{[14-11,9-4]}^{21} = 0$, $\Delta R_{[10]}^{24} = 1$, keep the pair. There are $2m + 3$ plaintext-ciphertext pairs left (11 bit conditions).

(iv) Step 4. Sieve the plaintext-ciphertext pairs according to the difference of the 22th round.

(i) Step 4.1. Guess $RK_{[22-19]}^{23}$ and test whether the input and output differences of $\Delta S_{[12-8]}^{22}$ satisfy the form $[0,1,0,0 - *,*,*,1]$ for the remaining pairs. If the pairs can satisfy the condition, save the pair (3 bit conditions).

(ii) Step 4.2. Guess $RK_{[30-27]}^{23}$ and test whether the input and output differences of $\Delta S_{[19-16]}^{23}$ satisfy the form $[1,0,0,0 - *,*,*,1]$ for the remaining pairs. If the pairs can satisfy the condition, save the pair (3 bit conditions).
(iii) Step 4.3 Guess $RK_{[21]}^{[31,2]}$ and test whether the input and output differences of $\Delta S_{[23–20]}^{[32]}$ satisfy the form $[1,*,*,* - *,*,*,*]$ for the remaining pairs. If the pairs can satisfy the condition, save the pair (1 bit condition). It is noted that $RK_{[21]}^{[31,0]} = RK_{[24]}^{[14,13]}$ have already been guessed in Step 3.1 and these redundant related keys will not be listed in the rest of the steps for simplicity.

(iv) Step 4.4. Sieve the remaining pairs with $\Delta R_{[2}]^{[22]}$. If $\Delta R_{[2]}^{[22]} - 0$, keep the pair (7 bit conditions).

(v) Step 5. Sieve the plaintext-ciphertext pairs according to the difference of the 3rd round. Guess extra 4 subkey bits $RK_{[22–19]}$ and calculate $\Delta S_{[23–20]}^{[32]}$. If $\Delta S_{[23–20]}^{[32]}$ satisfies the form $[1,*,1,* - 0,0,1,0]$ for the remaining pairs and if the pairs can satisfy the condition, save the pair (3 bit conditions).

(vi) Step 6. Sieve the incorrect subkey candidates according to the difference of the 21th round. Guess extra 4 subkey bits $RK_{[22–19]}$ and test whether the input and output differences of $\Delta S_{[11–8]}^{[21]}$ satisfy the form $[0,1,0,0 - *,*,*,1]$ and if $\Delta R_{[22–20]} = 0$ (6 bit conditions) if the pairs can satisfy the condition, it will lead to an impossible differential distinguisher; thus the guessed subkey candidate is false. Discard this key candidate and test another one, and iterate this process to reduce the space of key candidates.

5.3. Complexities of the Attack. According to our attack, the values are $|\Delta_{in}| = 34$, $|\Delta_{out}| = 60$, $c_{in} = 34$, and $c_{out} = 60$. As the data complexity for this attack always exceeds the full code book, it is infeasible to propose a valid impossible differential cryptanalysis with a single distinguisher. So, we will use multiple distinguishers to reduce this data complexity. Concretely speaking, twenty 18-round distinguishers with the following form are used (all the 65 18-round distinguishers can be found at the Appendix).

Input difference: $\Delta R_{[2]}^{[3]} = 1$. Key difference: $\Delta K_{[110]} = 1$

Output difference: $\Delta L_{[i]}^{[1]}$, $\Delta R_{[i]}^{[1]} = 1$, $i \in \{0 - 10, 15 - 18, 27 - 31\}$

So $n_{out} = 20$ and $|\Delta_{out}| = |\Delta_{out}| + \log_2 (n_{out}) = 60 + 4.32 = 64.32$.

As there are altogether 23 active Sboxes (50 independent subkey bits) involved in the calculation, $C'_{E} = (23/8 \cdot 24) = 2^{–1.12}$. According to the equations in Section 5.1, the data complexity, memory complexity, and time complexity for our attack are calculated as follows:

Data complexity: $C_{t} = \max \{ \min \{ \sqrt{2^{34}}, \sqrt{2^{34} + 1 - 34}, \sqrt{2^{34}}, \sqrt{2^{64} \cdot 64 - 34} \}, 2^{34} \cdot 2^{64} \cdot 1 - 34, \sqrt{2^{34}}, \sqrt{2^{64} - 64.32} \} = 2^{60.68}$.

Memory complexity: $N = 2^{35}$ plaintext pairs and corresponding ciphertext pairs.

Time complexity: $T = (2^{60.68} + (2^{94} + 2^{50} \cdot (2^{94}/2^{94})) \cdot 2^{–1.12} + 2^{128} \cdot 0.3679) \cdot 2^{–1.12} \approx 2^{123.44}$ LiCi-2 encryption.

6. Conclusion

In this paper, based on the idea of related-key multiple impossible differential cryptanalysis, a full-round attack on LiCi-2 is proposed. As a first step, several properties and observations on LiCi-2 are proposed to give an overall estimation on LiCi-2 against single-key and related-key impossible differential cryptanalysis. Some 18-round related-key impossible differential distinguishers are proposed alongside. As a second step, based on the newly constructed 18-round impossible differential distinguishers, a full-round key recovery attack is presented with adding three rounds before and four rounds after the distinguisher. In the future, some modifications on LiCi-2 are needed to avoid the reported attack.

Appendix

All the 18 related-round key possible impossible differential distinguishers are based on the one-bit input and key difference, as well as 2-bit output difference.

(i) $e[i][j]$ represents a single-bit difference at the $j$th bit; if $i = 0$, the difference appears at the left branch; if $i = 1$, the difference appears at the right branch.


No. 17: Input difference: $e[1][21]$. Output difference: $e[0][0], e[1][0]$. Key difference: $e[110]$
No. 18: Input difference: $e[1][21]$. Output difference: $e[0][1], e[1][1]$. Key difference: $e[110]$
No. 20: Input difference: $e[1][21]$. Output difference: $e[0][3], e[1][3]$. Key difference: $e[110]$
No. 25: Input difference: $e[1][21]$. Output difference: $e[0][8], e[1][8]$. Key difference: $e[110]$
No. 26: Input difference: $e[1][21]$. Output difference: $e[0][9], e[1][9]$. Key difference: $e[110]$
No. 27: Input difference: $e[1][21]$. Output difference: $e[0][10], e[1][10]$. Key difference: $e[110]$
No. 29: Input difference: $e[1][21]$. Output difference: $e[0][16], e[1][16]$. Key difference: $e[110]$
No. 30: Input difference: $e[1][21]$. Output difference: $e[0][17], e[1][17]$. Key difference: $e[110]$
No. 31: Input difference: $e[1][21]$. Output difference: $e[0][18], e[1][18]$. Key difference: $e[110]$
No. 32: Input difference: $e[1][21]$. Output difference: $e[0][27], e[1][27]$. Key difference: $e[110]$
No. 33: Input difference: $e[1][21]$. Output difference: $e[0][28], e[1][28]$. Key difference: $e[110]$
No. 34: Input difference: $e[1][21]$. Output difference: $e[0][29], e[1][29]$. Key difference: $e[110]$
No. 35: Input difference: $e[1][21]$. Output difference: $e[0][30], e[1][30]$. Key difference: $e[110]$
No. 36: Input difference: $e[1][21]$. Output difference: $e[0][31], e[1][31]$. Key difference: $e[110]$
No. 37: Input difference: $e[1][22]$. Output difference: $e[0][0], e[1][0]$. Key difference: $e[111]$
No. 38: Input difference: $e[1][22]$. Output difference: $e[0][1], e[1][1]$. Key difference: $e[111]$
No. 39: Input difference: $e[1][22]$. Output difference: $e[0][2], e[1][2]$. Key difference: $e[111]$
No. 40: Input difference: $e[1][22]$. Output difference: $e[0][3], e[1][3]$. Key difference: $e[111]$
No. 41: Input difference: $e[1][22]$. Output difference: $e[0][5], e[1][5]$. Key difference: $e[111]$
No. 42: Input difference: $e[1][22]$. Output difference: $e[0][7], e[1][7]$. Key difference: $e[111]$
No. 43: Input difference: $e[1][22]$. Output difference: $e[0][8], e[1][8]$. Key difference: $e[111]$
No. 44: Input difference: $e[1][22]$. Output difference: $e[0][9], e[1][9]$. Key difference: $e[111]$
No. 46: Input difference: $e[1][22]$. Output difference: $e[0][10], e[1][10]$. Key difference: $e[111]$
No. 48: Input difference: $e[1][22]$. Output difference: $e[0][31], e[1][31]$. Key difference: $e[111]$
No. 56: Input difference: $e[1][26]$. Output difference: $e[0][12], e[1][12]$. Key difference: $e[8]$

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grants nos. 61802437, 62102448, 61972248, and 61902428 and China Postdoctoral Science Foundation under Grant no. 2020M681314.
References


