Automatic Search Model for Related-Tweakey Impossible Differential Cryptanalysis

Huiqin Chen[1,4], Yongqiang Li[1,4], Xichao Hu[1,4], Zhengbin Liu[2], Lin Jiao[3], and Mingsheng Wang[1,4]

[1] Institute of Information Engineering, Chinese Academy of Science, Beijing, China, {chenhuiqin,liyongqiang,huxichao,wangmingsheng}@iie.ac.cn
[2] Science and Technology on Communication Security Laboratory, Chengdu, China, zhengbinliu@126.com
[3] State Key Laboratory of Cryptology, Beijing, China, jiaolin_jl@126.com
[4] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. The design and analysis of dedicated tweakable block ciphers constitute a dynamic and relatively recent research field in symmetric cryptanalysis. The assessment of security in the related-tweakey model is of utmost importance owing to the existence of a public tweak. This paper proposes an automatic search model for identifying related-tweakey impossible differentials based on the propagation of states under specific constraints, which is inspired by the research of Hu et al. in ASIACRYPT 2020. Our model is universally applicable to block ciphers, but its search efficiency may be limited in some cases. To address this issue, we introduce the Locality Constraint Analysis (LCA) technique to impossible differential cryptanalysis and propose a generalized automatic search model. Technically, we transform our models into Satisfiability Modulo Theories (SMT) problems and solve them using the STP solver. We have applied our tools to several tweakable block ciphers, such as `Joltik-BC`, `SKINNY`, `QARMA`, and `CRAFT`, to evaluate their effectiveness and practicality. Specifically, we have discovered 7-round related-tweakey impossible differentials for `Joltik-BC-192`, and 12-round related-tweak impossible differentials, as well as 15-round related-tweakey impossible differentials for `CRAFT` for the first time. Based on the search results, we demonstrate that the LCA technique can be effectively performed when searching and determining the contradictory positions for the distinguisher with long trails or ciphers with large sizes in impossible differential cryptanalysis.

Keywords: Tweakable Block Cipher · Related-tweakey · Impossible differential cryptanalysis · LCA technique · SAT method

1 Introduction

Tweakable block ciphers are constructions that have an additional input called tweak compared to traditional block ciphers, which can be defined as a function $C = E(K, T, P)$ from $\mathbb{F}_2^n \times \mathbb{F}_2^\kappa \times \mathbb{F}_2^t \to \mathbb{F}_2^n$ when the tweak length is t bits. The concept of tweakable block ciphers was first introduced by Schroeppel in

the Hasty Pudding Cipher [32], and was later formalized by Liskov et al.[23,24]. They aimed to move the randomization of symmetric primitives by bringing the high-level mode operations, like ΘCB3 [18] or Counter-in-Tweak [29], directly to the design of block ciphers. Unlike the secret key, the tweak is entirely public and offers attackers more flexibility. Designers must therefore handle the tweak more carefully than the key without reducing efficiency. Responding to the high demmand, Jean et al. [13] introduced the `TWEAKEY` framework to bridge the gap between key and tweak inputs by providing a unified framework in ASIACRYPT 2014, which can be viewed as a straightforward generalization of key-alternating ciphers, where the key and tweak basically treated as a whole called *tweakey*. Based on this framwork , there are several dedicated tweakable block ciphers, such as `Joltik-BC` [14], `Deoxys-BC` [15], `SKINNY` [3]. Furthermore, with the development of tweakable block cipher, its design also becomes diversified, such as `QARMA` [1], `CARFT` [4], and some other tweakable block ciphers based on Tweak-aNd-Tweak [9] and Elastic-Tweak [6].

Impossible differential cryptanalysis was independently introduced by Biham et al. [5] and Knudsen [17] to evaluate the security of `Skipjack` and `DEAL`. In contrast to differential cryptanalysis, impossible differential cryptanalysis aims to identify a differential characteristic that have zero probability. Due to the limitations of manual derivation, various automatic methods have been developed to search for impossible differentials, including the $\mathcal{U}$-method [16], the UID-method [27], and the $\mathcal{WW}$-method [34]. Unfortunately, these method handle the underlying S-box as ideal and cannot consider its details. However, this problem was soon settled with the Mixed Integer Linear Programming (MILP) application to cryptanalysis. It was firstly proposed by Mouha et al. [28] to evaluate the lower bound on the number of the differential and linear active S-boxes and then improved by Sun et al. [33] to search for the differential characteristics of bit-oriented block ciphers. Based on this, Cui et al. [7] proposed a MILP-based tool to search the impossible differentials for lightweight block ciphers and an algorithm to verify the impossible differentials. Soon after, Sasaki and Todo [31] presented a MILP-based tool to search the impossible differential for SPN block ciphers by treating the large S-boxes as permutations so that their tool was valid to detect the contradiction in linear components.

However, the above methods are all based on the propagation of the differences and can not evaluate the effect of key schedules in the single-key setting. Hu et al. [12] solved this problem by using the equivalence between the impossible $(s+1)$-polytopic tansitions and impossible differentials. They transformed the characterization of differential propagation from the traditional sense of describing it through differential spreads, to reflecting it through the propagation of constraint values. This new approach provides a novel perspective and enables the possibility of handling large state S-boxes or value-dependent operations that are difficult to realize in the traditional sense. Additionally, this approach is applicable to all differential cryptanalysis methods, such as searching differential trail or differential active S-box, which facilitates more accurate analysis of a block cipher to resist the attacks of differential cryptanalysis.

Our Contributions. For the majority of current tweakable block ciphers, adversaries have the ability to manipulate tweak values. Drawing inspiration from Hu et al.'s contributions in [12], we present an automatic search model for related-tweakey impossible differentials. Specifically, we transform the problem of identifying an impossible differential into the Satisfiability Modulo Theories (SMT) problem by explicating the propagation of states and the tweakey update function with specific constraints, which can efficiently evaluate the resistance against impossible differential analysis for most of the block ciphers.

Unfortunately, it leads to a significant loss of efficiency with an increase in the state space and number of search rounds if considering all the details of round functions and tweakey update functions. To address this, we propose a generalized search model by introducing the Locality Constraint Analysis (LCA) technique. The optimized model has two significant advantages: improving the search efficiency for long trails and identifying the contradictory positions of impossible differentials.

In terms of practical implementation, we have employed our automatic search model in the evaluation of several tweakable block ciphers. The outcomes of these evaluations are presented below.

- For `Joltik-BC`, we have discovered several 6-round and 7-round related-tweakey impossible differentials for `Joltik-BC-128` and `Joltik-BC-192`, respectively, wherein a single nibble is active for input and output. These differentials were previously unknown.
- For `SKINNY`, we have identified related-tweakey impossible differentials for `SKINNY-64-64`, `SKINNY-64-128`, and `SKINNY-62-192`, with 12-round, 14-round, and 16-round, respectively. Notably, the majority of these differentials had not been previously reported by Sadeghi et al. in [30].
- For `QARMA-64`, we have derived several 7-round asymmetric related-tweak impossible differential distinguishers spanning from the 6th to the 12th round. Particularly, the majority of these distinguishers were not identified using Zong's method in [36].
- For `CRAFT`, we have successfully derived 12-round related-tweak impossible differentials and 15-round related-tweakey impossible differentials, assuming the condition that only one nibble is active in the tweakey differences. It is noteworthy that these differential properties have not been reported before.

Outline. In Section 2, we provide a brief overview of the necessary preliminaries utilized in the present paper. Subsequently, in Section 3, we introduce an automatic search model for related-tweakey impossible differentials based on the SAT solver. Section 4 is dedicated to the application of our tool in the search for related-tweakey impossible differentials in some tweakable block ciphers, followed by a concise evaluation of our model in Section 5. Finally, we conclude this work in Section 6. The source codes are publicly available at https://github.com/Rainy1024/ImpossibleDifferentialAnalysis.git.

2 Preliminaries

2.1 Notations

The following notations are used in the present paper. Throughout the paper, we use $\oplus$ to denote the bitwise XOR of two vectors or XOR of two bits.

- $\mathbb{F}_2 = \{0, 1\}$: the finite field with 2 elements.
- $\mathbb{F}_2^n$: the vectors space over the finite field $\mathbb{F}_2$ with dimension n.
- Δ_m^n: the set that $\{(a, a') \in \mathbb{F}_2^n \times \mathbb{F}_2^n | a \oplus a' = m, m \in \mathbb{F}_2^n \setminus \{0\}\}$.
- $BC(n, m, l)$: the set of iterated block ciphers whose block cipher is n-bit, master key size is m-bit, and round key size is l-bit.
- $TBC(n, \kappa, t)$: the set of tweakable block ciphers whose cipher size is n-bit, master key size is κ-bit and initial tweak size is t-bit.
- $TK_j^r[i]$: the i-th nibble of the j-th subtweakey of the r-th round. The difference donates as $\triangle TK_j^r[i]$.
- `DR`: the length of an impossible differential distinguisher.
- `ConR`: the round index where the contradiction occurs.
- `ConPs`: The specific location of the contradiction. For instance, S_i means the contradiction is in the S-box with the index i.

2.2 Related-tweakey Impossible Differential

Related-key impossible differential cryptanalysis is a variant of the impossible differential cryptanalysis where the attacker can control the key schedule of the cipher. In this attack, the attacker can choose two related keys and use them to generate a specific input difference that produces a target output difference with zero probability. Here, we first recall some definitions of impossible 2-polytopic transitions proposed in [12].

For an iterated block cipher $E \in BC(n, m, l)$, the tuple (x, x') with $x, x' \in \mathbb{F}_2^n$ is called a 2-polygon in $\mathbb{F}_2^n$. The 2-polygon (x_{r_b}, x'_{r_b}) propagates through round by round. If there exits an r-round related-key 2-polygonal trail

$$((x_{r_b}, x'_{r_b}), (E^1_{k_{r_b}}(x_{r_b}), E^1_{k'_{r_b}}(x'_{r_b})), \ldots, (E^r_{k_{r_b+r-1}}(x_{r_b+r-1}), E^r_{k'_{r_b+r-1}}(x'_{r_b+r-1})))$$

such that

$$(x_{r_e}, x'_{r_e}) = (E^r_{k_{r_b+r-1}}(x_{r_b+r-1}), E^r_{k'_{r_b+r-1}}(x'_{r_b+r-1})),$$

the triplets $((x_{r_b}, x'_{r_b}), (k_{r_b}, k'_{r_b}), (x_{r_e}, x'_{r_e}))$ is called an r-round dependent-key possible 2-polygons. Otherwise, it is an r-round dependent-key impossible 2-polygons of E. Based on this, we redefine the related-tweakey impossible differential for tweakable block ciphers.

Definition 1 (Related-tweakey Impossible Differential). *For a tweakable block cipher $E \in TBC(n, \kappa, t)$, if $((s_{r_b}, s'_{r_b}), (tk, tk'), (s_{r_e}, s'_{r_e}))$ is an $(r_e - r_b)$-round dependent-tweakey impossible 2-polygons, where tk is the initial tweakey, and $\forall (s_{r_b}, s'_{r_b}) \in \Delta_\alpha^n$, $\forall (s_{r_e}, s'_{r_e}) \in \Delta_\beta^n$, $\forall (tk, tk') \in \Delta_\delta^{\kappa+t}$, the triplet (α, β, δ) is called an $(r_e - r_b)$-round related-tweakey impossible differential.*

According to Definition 1, instead of describing the differential propagation, we pay attention to the propagation of values with certain constraints in the present paper. Specifically, refering to the automatic search model proposed in [12], we give an autoamtic search model for the $(r_e - r_b)$-round[5] related-tweakey impossible differentials by considering the propagation of states from the r_bth round to the r_eth round, which is shown in Algorithm 1.

Algorithm 1: The Model for related-tweakey impossible differentials

Input: $E \in TBC(n, \kappa, t)$, r_b, r_e
Output: The length of distinguisher and the values of input differentials

Generate $\Omega = \{(\alpha, \beta, \delta) | \alpha, \beta \in \mathbb{F}_2^n, \delta \in \mathbb{F}_2^{\kappa+t}, (\alpha, \beta, \delta) \neq (0, 0, 0)\}$;
Define: *distinguisher find* = True;
while *distinguisher find* **do**
    distinguisher find = False;
    foreach $(\alpha, \beta, \delta) \in \Omega$ **do**
        `// Step 1: Describe the cipher` E `in CVC format`
        Declare all variables to be used;
        Describe the propagation of $(tk_0, tk'_0) \rightarrow \cdots \rightarrow (tk_{r_e}, tk'_{r_e})$;
        Describe the propagation of $(s_{r_b}, s'_{r_b}) \rightarrow \cdots \rightarrow (s_{r_e}, s'_{r_e})$;
        Add the constraints: $s_{r_b} \oplus s'_{r_b} = \alpha$, $s_{r_e} \oplus s'_{r_e} = \beta$, $tk \oplus tk = \delta$;
        Add the statements "QUERY(FALSE);"
         "COUNTEREXAMPLE;";
        `// Step 2: Invoke the STP to solve the file`
        Start to solve the file;
        if *solver returns "Valid"* **then**
            Record the triplets (α, β, δ) and the round number (r_b, r_e);
            distinguisher find = True;
            Break;
    if *distinuisher find* **then**
        The distinguisher from r_b to r_e is found in Ω;
        Let $r_e = r_e + 1$;
    else
        The distinguisher from r_b to r_e is not found in Ω;

2.3 Boolean Satisfiability Problem

The *Boolean Satisfiability Problem* (SAT) is to find whether a set of variables, which if plugged into a boolean expression, will result in "True". Any boolean expression can be converted to normal form and the conjunctive normal form

[5] In the present paper, we use r_b and r_e to represent the beginning and ending round, respectively. With this method, we can adequately consider the influence of the number of rounds on the state propagation and accurately locate the distinguisher's position, which is more convenient for constructing a key-recovery attacks.

(CNF) is one of them. The CNF expression is a bunch of clauses consisting of variables, ORs, and NOTs, all of which are then glued together with AND into a full expression. SAT solver is merely a solver of huge boolean equations in CNF form. It just gives the answer, if there is a set of input values that can satisfy CNF expression, and what input values must be. There have been some heuristic SAT solvers. Most support CNF files as the standard input format, such as Cryptominisat [19].

The *Satisfiability Modulo Theories* (SMT) problem is an extension of the SAT problem, in which CNF formulas are enriched by binary-valued functions over a suitable set of binary and (or) non-binary variables. Many works searching for the differential and linear characteristics are based on the SMT problem, where STP[6] is a common solver for SMT problems. STP supports the CVC format and starts from an initial assignment for the literals, then builds a search tree using systematic backtracking until all conflicting clauses are resolved. An SMT problem is unsatisfiable if returning either an assignment of variables for a satisfiable set of clauses or a predicate indicates. However, when invoking STP to solve an SMT problem, the solver first interprets SMT instances in CVC format into SAT instances with CNF and then determines its satisfiability.

3 The Optimized Automatic Search Model

By utilizing Algorithm 1 to investigate related-tweakey impossible differentials, we observe that with an increase in the number of search rounds, the equation system employed to represent the state propagation expands correspondingly. This leads to an exponential escalation in both the runtime and memory requirements caused by the augmented amount of data acquired during the database query process. To overcome these impediments and enhance the efficiency of Algorithm 1, we propose an optimized automatic search model based on the LCA technique in the section.

3.1 Application of LCA in Impossible Differential Cryptanalysis

Locality Constraint Analysis (LCA) is an analytical method that uses the properties of local variables to deduce global features. In the impossible differential analysis, if $E^k_{r_1}(\Delta^n_\alpha) = D^k_{r_2}(\Delta^n_\beta)$ is never satisfied under any k for $E \in BC(n, m, l)$, the differential (α, β) is called an impossible differential. However, according to the security criterion for confusion and diffusion in the design of a block cipher, with the exception of some positions in which contradictions may occur, the value of the other positions almost reaches full diffusion after several rounds of iteration, which means that the values in those positions can traverse the entire space. Therefore, we can use the LCA technique to determine an impossible differential by considering some of the positions instead of the full state.

From the perspective of theoretical analysis, suppose $x = (x_0, x_1, \cdots, x_{n-1})$ with $x_i \in \mathbb{F}_2$, if it satisfies $x_i = 0$ for $\forall x_i \in x$, i.e. $\bigvee_{0 \le i \le n-1} x_i = 0$, we call that

[6] https://github.com/stp/

x is inactive. Otherwise, x is active. Then we can draw the following conclusion according to the definition of related-tweakey impossible differential.

Theorem 1. *Let $E(x, tk) \in TBC(n, \kappa, t)$ be a tweakable block cipher and $\mathbb{CP}$ be a tuple including the sets of possible contradictory positions that need to be constrained in the search model. For any $\alpha, \beta \in \mathbb{F}_2^n$, $\delta \in \mathbb{F}_2^{\kappa+t} \setminus \{0\}$, if there exits a set $\mathbb{P} \subset \mathbb{CP}$, such that*

$$LCA := \bigvee_{i \in \mathbb{P}} C_i(x, y, tk) \oplus C_i(x \oplus \alpha, y \oplus \beta, tk \oplus \delta)$$

is active for $\forall x, y \in \mathbb{F}_2^n$ and $\forall tk \in \mathbb{F}_2^{\kappa+t}$, where $C_i(x, y, tk) := E_{r_1}[i](x, tk) \oplus D_{r_2}[i](y, tk)$ and $D_r(E_r(x, tk), tk) = x$. Then (α, β, δ) is an $(r_1 + r_2)$-round related-tweakey impossible differential of $E(x, tk)$.

Proof. According to Definition 1, if proving (α, β, δ) is an (r_1+r_2)-round related-tweakey impossible differential for $(\alpha, \beta, \delta) \neq (0, 0, 0)$, we need to prove that

$$E_r(x, tk) \oplus E_r(x \oplus \alpha, tk \oplus \delta) = \beta$$

does not hold for any $x \in \mathbb{F}_2^n$ and $tk \in \mathbb{F}_2^{\kappa+t}$, where $r = r_1 + r_2$. Note that $E_r(x, tk) = E_{r_2}(E_{r_1}(x, tk), tk)$, so the above equation is equivalent to

$$E_{r_2}(E_{r_1}(x \oplus \alpha, tk \oplus \delta), tk \oplus \delta) = E_{r_2}(E_{r_1}(x, tk), tk) \oplus \beta$$

By composing $D_{r_2}(x, tk)$, we get that

$$E_{r_1}(x \oplus \alpha, tk \oplus \delta) = D_{r_2}(E_{r_2}(E_{r_1}(x, tk), tk) \oplus \beta, tk \oplus \delta)$$

Let $y = E_{r_2}(E_{r_1}(x, tk), tk)$. Then $E_{r_1}(x, tk) = D_{r_2}(y, tk)$ and hence we have

$$E_{r_1}(x, tk) \oplus E_{r_1}(x \oplus \alpha, tk \oplus \delta) = D_{r_2}(y, tk) \oplus D_{r_2}(y \oplus \beta, tk \oplus \delta) \tag{1}$$

However, since LCA is active for $\mathbb{P} \subseteq \mathbb{CP}$, that is

$$\bigvee_{i \in \mathbb{CP}} C_i(x, y, tk) \oplus C_i(x \oplus \alpha, y \oplus \beta, tk \oplus \delta) \neq 0$$

Thus, $\exists i \in \mathbb{P}$ such that for $\forall x, y \in \mathbb{F}_2^n$ and $\forall tk \in \mathbb{F}_2^{\kappa+t}$,

$$\begin{aligned} 1 =& C_i(x, y, tk) \oplus C_i(x \oplus \alpha, y \oplus \beta, tk \oplus \delta) \\ =& E_{r_1}[i](x, tk) \oplus D_{r_2}[i](y, tk) \oplus E_{r_1}[i](x \oplus \alpha, tk \oplus \delta) \oplus D_{r_2}[i](y \oplus \beta, tk \oplus \delta) \end{aligned}$$

This means that Equation (1) does not hold for any $x, y \in \mathbb{F}_2^n$, $tk \in \mathbb{F}_2^{\kappa+t}$. Therefore, (α, β, δ) is an $(r_1 + r_2)$-round related-tweakey impossible differential of $E(x, tk)$.

The idea of our approach. We use the "miss-in-the-middle" method to find impossible differential distinguishers of block ciphers. In contrast, we weaken the conditions of the intermediate constraints. As shown in Fig.1, we split an $(r_1 + r_2)$-round impossible differential into an r_1-round encryption and r_2-round decryption and only pay attention to the values of a few bits in the middle with the LCA technique.

In particular, suppose that $\mathbb{P} = \{i_0, i_1, \cdots, i_m\}$ is a set in which contradictions may occur. Then, if the equation

$$\bigvee_{i \in \mathbb{P}} E_{r_1}(x, tk)[i] \oplus E_{r_1}(x', tk')[i] \oplus D_{r_2}(y, tk)[i] \oplus D_{r_2}(y', tk')[i] = 0$$

is never satisfied for $\forall (x, x') \in \Delta_\alpha^n$, $\forall (y, y') \in \Delta_\beta^n$ and $\forall (tk, tk') \in \Delta_\delta^{\kappa+t}$, the triplet (α, β, δ) is an (r_1+r_2)-round related-tweakey impossible differential. However, it is worth noting that a differential triplet (α, β, δ) satisfying Theorem 1 is a related-tweakey impossible differential, not vice versa.

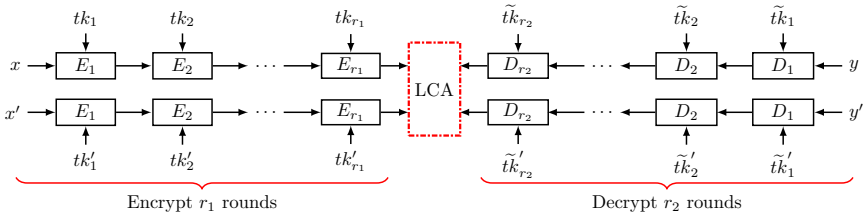

Fig. 1: The Optimization Scheme of Our Automatic Search Model

3.2 The Optimized Automatic Search Model for Related-tweakey Impossible Differentials

Based on the preceding analysis, we present an optimized automatic search model for related-tweakey impossible differentials, outlined in Algorithm 2.

Specifically, given a tweakable block cipher $E \in TBC(n, \kappa, t)$, the determination of whether a triplet (α, β, δ) is an $(r_e - r_b)$-round related-tweakey impossible differential can be accomplished through three phases: search space determination, statements generation, and STP invocation. Initially, the input parameters are the starting round number r_b, the termination round number r_e, and r_m where the constraints are added. For each triplet (α, β, δ) in the search space Ω, whether (α, β, δ) constitutes an $(r_e - r_b)$-round related-tweakey impossible differential is transformed into the corresponding SMT problem using the CVC language and solved by invocation of the STP solver. Finally, Algorithm 2 outputs the length of distinguishers and the corresponding input and output differentials. Further details of Algorithm 2 are presented below.

Specification of the search space determination phase. The efficacy of our automated search approach hinges predominantly on two factors, as demonstrated in Lines 6 and 7 of Algorithm 2: the duration needed to complete a

Algorithm 2: Optimized automatic search model using LCA technique

Input: $E \in TBC(n, \kappa, t)$, r_b, r_e, r_m
Output: (α, β, δ), (r_b, r_e, r_m), and $\mathbb{P}$

Generate $\Omega = \{(\alpha, \beta, \delta) | \alpha, \beta \in \mathbb{F}_2^n, \delta \in \mathbb{F}_2^{\kappa+t}, (\alpha, \beta, \delta) \neq (0, 0, 0)\}$;
Generate a constraint set $\mathbb{CP}$;
Define: *distinguisher find* = True;
while *distinguisher find* **do**
    distinguisher find = False;
    foreach $(\alpha, \beta, \delta) \in \Omega$ **do**
        foreach $\mathbb{P} \subseteq \mathbb{CP}$ **do**
            `// Step 1: Describe the cipher` E `in CVC format`
            Declare all variables to be used;
            Describe the propagation of $(tk_0, tk'_0) \rightarrow \cdots \rightarrow (tk_{r_e}, tk'_{r_e})$;
            Describe the propagation of $(s_{r_b}, s'_{r_b}) \rightarrow \cdots \rightarrow (s_{r_m}, s'_{r_m})$;
            Describe the propagation of $(\hat{s}_{r_m}, \hat{s}'_{r_m}) \rightarrow \cdots \rightarrow (s_{r_e}, s'_{r_e})$;
            `// Constrain the input and output difference`
            Add the constraints: $s_{r_b} \oplus s'_{r_b} = \alpha$, $s_{r_e} \oplus s'_{r_e} = \beta$, $tk \oplus tk' = \delta$.;
            `// Locality Constraint Analysis`
            foreach $i \in \mathbb{P}$ **do**
                Add the constraints: $s_{r_m}[i] \oplus s'_{r_m}[i] \oplus \hat{s}_{r_m}[i] \oplus \hat{s}'_{r_m}[i] = 0$;
            Add the statements: "QUERY(FALSE);"
            "COUNTEREXAMPLE;" ;
            `// Step 2: Invoke the STP to solve the file`
            Start to solve the file;
            if *solver returns "Valid"* **then**
                Record the triplets (α, β, δ), the round (r_b, r_e, r_m), and the set of positions $\mathbb{P}$;
                distinguisher find = True;
                Break ; `// Break out of all` *`for`* `loops`
    if *distinuisher find* **then**
        The distinguisher from r_b to r_e is found in Ω;
        Let $r_e = r_e + 1$;
    else
        The distinguisher from r_b to r_e is not found in Ω;

search and the magnitude of the search space. As the search time is restricted by the size of the cipher and the hardware used, enhancing search efficiency can be challenging under limited resources. Consequently, selecting the search space judiciously so that a minimal number of elements reflect a greater number of differential properties will be pivotal in increasing search efficiency.

The choice of Ω. The utilization of linear tweak schedules and XOR operations for the purpose of mixing subtweakeys with internal states, as observed in

numerous state-of-the-art tweakable block ciphers, can inadvertently benefit potential attackers. Specifically, under the related-tweakey setting, an attacker can manipulate certain state values by XORing the same difference of subtweakeys at corresponding positions, thereby nullifying the difference of internal states. This, in turn, enables the attacker to pass one round function without incurring any additional cost, as depicted in Fig.2.

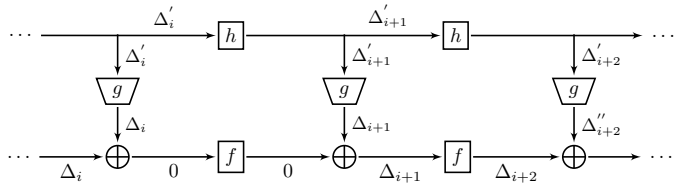

Fig. 2: The differential model under the related-tweakey setting[4].

Furthermore, Sasaki and Todo [31] have observed that all existing ciphers have the longest impossible differentials with only one active word in both input and output. In light of this, it is common practice to set the input and output difference to zero and only introduce differences to the tweakeys, that is, $\Omega = \{(\alpha, \beta, \delta) | \alpha = 0, \beta = 0, \delta \in \mathbb{F}_2^{\kappa+t} \setminus \{0\}\}$. The specific choice of δ depends on the cipher's structure, with one bit being active for bit-oriented encryption and one cell being active for cell-oriented encryption.

The choice of r_m and $\mathbb{CP}$. The parameters r_m and $\mathbb{CP}$ jointly determine the locations of the contradictions. Based on empirical observations and experimental tests, we observe that for a distinguisher of odd length, the contradictions typically manifest in the middle round; whereas for even length, they appear in the middle two rounds. As such, we derive the expression $r_m = \lceil \frac{r_b + r_e}{2} \rceil$ if $(r_e - r_b)$ is odd, and $r_m \in \{\frac{r_b + r_e}{2}, \frac{r_b + r_e}{2} + 1\}$ if $(r_e - r_b)$ is even. The selection of the constrained position tuple $\mathbb{CP}$ is also informed by empirical evidence and experimental results.

Especially, for ARX-based block ciphers, we apply a constraint tuple $\mathbb{CP} = \{[i] | 0 \leq i \leq (n-1)\}$, where we constrain one bit of the intermediate state in each search. To verify the effectiveness of this approach, we utilized Algorithm 2 on `SIMON` and `SPECK` [2], and the results are presented in Table 1, where only one branch is constrained to define $\mathbb{CP}$ for ciphers based on the Feistel structure. For SPN-based block ciphers, we consider an S-box as a constraint unit in our modified model, i.e., $\mathbb{CP} = \{S_i | 0 \leq i \leq (m-1)\}$, where $S_i = \{i | 0 \leq i \leq (m-1)\}$ for an m-bit S-box. Using this constraint, we applied Algorithm 2 to `SKINNY`, `QARMA`, and `CRAFT`. Notably, we define $\mathbb{CP} = \{\{S_{4i}, S_{4i+1}, S_{4i+2}, S_{4i+3}\} | 0 \leq i \leq 3\}$ when applying Algorithm 2 to `Joltik-BC`, since the matrix used in its `MixNibbles` operation is an MDS matrix.

[4] This is a `TWEAKEY` framework proposed by Jean et al. [13] to bridge the gap between key and tweak inputs in the design of tweakable block ciphers, which can be viewed as a straightforward generalization of key-alternating ciphers. In the model, f is the round function and h represents the tweakey update function.

Table 1: The experimental results for SIMON32/64 and SPECK32/64

Ciphers	Input Difference	Output Difference	DR	ConR	ConPs
SIMON32/64	0000 0000 0000 0000 1000 0000 0000 0000	0000 0001 0000 0000 0000 0000 0000 0000	11	6	24
SPECK32/64	0000 0000 0000 1000 0000 0000 0000 0000	1000 0000 0000 0000 1000 0000 0000 0010	6	2	5

Specification of statements generation phase. The statements generation phase is described in lines 8-16 of Algorithm 2. A detailed account of each step is then presented in the following.

- **Line 8.** Declare the variables to describe the propagation of round functions and tweakey schedules, including the variables that represent the input 2-polygon and output 2-polygons, tweakey 2-polygons, and some other intermediate variables.
- **Line 9-11.** According to the propagation rules for Copy, Xor, Modular Addition, Binary Matrix Multiplication and S-box given in [12], construct the propagation from the input 2-polygons (s_{r_b}, s'_{r_b}) to the output 2-polygons (s_{r_m}, s'_{r_m}) with the aid of the tweakey 2-polygons and intermediate variables in CVC format. Especially, the tweakey 2-polygons is constrained according to the tweakey schedule.
- **Line 12.** Generate the statements in CVC format such that the input and output 2-polygons satisfies that $s_{r_b} \oplus s'_{r_b} = \alpha$ and $s_{r_e} \oplus s'_{r_e} = \beta$, while the tweakey 2-polygons satisfies that $tk_{r_b} \oplus tk'_{r_b} = \delta$.
- **Line 13-14.** Generate the statements in CVC format such that the output 2-polygon of the first $(r_m - r_b)$ rounds and the input 2-polygon of the last $(r_e - r_m)$ rounds satisfies that $s_{r_m}[i] \oplus s'_{r_m}[i] \oplus \hat{s}_{r_m}[i] \oplus \hat{s}'_{r_m}[i] = 0$ for $\forall i \in \mathbb{P}$.
- **Line 15-16.** Add the statements "QUERY(FALSE);" and "COUNTEREXAMPLE" to the statements system, which is a common predicate in STP to determine whether an SMT problem has a solution.

Specification of the STP invocation phase. We invoke STP to tackle the file, which comprises a system of statements. If the outcome of STP is "Valid," this implies that no solution exists for the SMT problem. As such, the corresponding triplets (α, β, δ) represent an $(r_e - r_b)$-round related-tweakey impossible differential, where r_m and $\mathbb{P}$ ascertain the contradictory positions. Alternatively, if STP returns "Invalid" along with a collection of solutions, the triplets (α, β, δ) do not denote an $(r_e - r_b)$-round related-tweakey impossible differential, and these solutions constitute the corresponding differential characteristic from round r_b to round r_e for E.

4 Applications from Cryptanalysis Aspect

In this section, we apply our automatic search model to Joltik-BC, SKINNY, QARMA, and CRAFT from the cryptanalysis aspect. Especially, when searching for related-tweakey impossible differentials, only the tweakey is modified while

keeping the input and output differences at zero, that is, $\Omega = \{(0, 0, \delta) | \delta \in \mathbb{F}_2^{\kappa+t} \setminus \{0\}\}$, where κ and t are constants. Consequently, by exploiting the relationship between the tweakey and the state of a cipher, an impossible differential can be derived for the $(r+2)$-round if a r-round related-tweakey impossible differential is found within the search space Ω. Furthermore, Δ_{in} and Δ_{out} denote the input and output difference of the operation `AddRoundTweakey`, respectively.

4.1 Application to `Joltik-BC`

`Joltik-BC` is an iterative substitution-permutation network that transforms the initial plaintext through a series of round functions (that depend on the key and the tweak) to a ciphertext. The cipher exists in two variations, namely `Joltik-BC-128`, with a total key and tweak size of 128 bits, and `Joltik-BC-192`, with a combined key and tweak size of 192 bits. Additional information regarding `Joltik-BC` can be found in [14]. Notably, the construction of `Joltik-BC` is based on the Superposition `TWEAKEY` design [13], with the tweakey schedule satisfying Proposition 1. This property allows for greater differential properties when assessing differential propagation.

Proposition 1 (Cancellation of the Tweak Differences [14]). *Cancellation of differences (in general as the key schedule is linear) in the chosen nibble of TK-p cannot occur more than $(p-1)$ times. For TK-2 this means that the accumulative difference coming from the subtweakeys can be canceled only once by XOR of the subtweakeys. For TK-3, this can happen twice.*

Previous Cryptanalysis. To the best of our knowledge, the most extensive distinguisher discovered for the block cipher `Joltik-BC-128` is a 6-round related-tweak impossible differential, which was proposed in [36]. This particular impossible differential exhibits two active nibbles for both input and output differences. For `Joltik-BC-192`, no public impossible differential has been identified, apart from a meet-in-the-middle distinguisher that spans 7 rounds, which was constructed in [20].

List of 6-Round Related-tweakey Impossible Differentials for `Joltik-BC-128`. By introducing the difference to TK_1^r and TK_2^r in a single nibble, we applied Algorithm 1 to `Joltik-BC-128` and discovered a 6-round related-tweakey impossible differential with a time of 4.43 seconds. To confirm the absence of a 7-round impossible differential in the search space, we conducted a verification process by traversing the entire search space, which took approximately 23.4 hours. Based on Proposition 1, the search results can be classified into three cases. The corresponding values are presented in Table 2.

Case 1. The cancellation occurs during the second round of the distinguisher. Specifically, this cancellation is characterized by $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r$, $\Delta_{out} = \triangle TK_1^{r+6} \oplus \triangle TK_2^{r+6}$, and $\triangle TK_1^r[i] \oplus KS^1(\triangle TK_2^r[i], 2) = 0$, subject to the constraint that $\triangle TK_1^r[i] \neq \triangle TK_2^r[i] \neq 0$, where $i \in \{0, 1, \cdots, 15\}$.

Table 2: The 6-round related-tweakey impossible differentials for `Joltik-BC-128`

Cases	$(\triangle TK_1^r[i], \triangle TK_2^r[i])$	Index
Case 1	$(1,9)$,$(2,1)$,$(3,8)$,$(4,2)$,$(5,B)$,$(6,3)$,$(7,A)$,$(8,4)$ $(9,D)$,$(A,5)$,(B,C),$(C,6)$,(D,F),$(E,7)$,(F,E)	$i \in \{0,1,2,\cdots,15\}$
Case 2	$(1,7)$,$(2,E)$,$(3,9)$,$(4,F)$,$(5,8)$,$(6,1)$,$(7,6)$,$(8,D)$ $(9,A)$,$(A,3)$,$(B,4)$,$(C,2)$,$(D,5)$,(E,C),(F,B)	$i \in \{0,1,2,\cdots,15\}$
Case 3	$(9,3)$,$(D,6)$	$i \in \{0,2,4,\cdots,14\}$

Case 2. The cancellation occurs during the fifth round of the distinguisher. At this point, the input difference is denoted by $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r$, and the output difference is denoted by $\Delta_{out} = \triangle TK_1^{r+6} \oplus \triangle TK_2^{r+6}$. Additionally, it holds that $\triangle TK_1^r[i] \oplus KS^5(\triangle TK_2^r[i], 2) = 0$, where $\triangle TK_1^r[i] \neq \triangle TK_2^r[i] \neq 0$ for $i \in \{0,1,\cdots,15\}$.

Case 3. The distinguisher is not subject to cancellation. However, if the initial difference of the tweakey satisfies that $(\triangle TK_1^r[i], \triangle TK_2^r[i]) \in \{(9,3),(D,6)\}$ for $i \in \{0,2,4,\cdots,14\}$, then the input difference $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r$ cannot propagate to the output difference $\Delta_{out} = \triangle TK_1^{r+6} \oplus \triangle TK_2^{r+6}$ after 6-round encryption.

List of 7-Round Related-tweakey Impossible Differentials for `Joltik-BC-192`. By introducing differences to the same nibble of TK_1^r, TK_2^r, and TK_3^r, respectively, a 7-round related-tweakey impossible differential is obtained with a time of 2403.67 seconds. It required approximately 25 days[7] to verify the non-existence of an 8-round impossible differential in the search space. As Proposition 1 suggests, the tweakey differences can be canceled twice. The search results can be categorized into the following five cases, as shown in Table 3.

Case 1. The cancellation occurs in both the second round and the third round of the 7-round distinguisher. Specifically, if the difference between the tweakeys satisfies the conditions that $\triangle TK_1^r[i] \oplus KS^1(\triangle TK_2^r[i], 2) \oplus KS^1(\triangle TK_3^r[i], 4) = 0$ and $\triangle TK_1^r[i] \oplus KS^2(\triangle TK_2^r[i], 2) \oplus KS^2(\triangle TK_3^r[i], 4) = 0$, with $\triangle TK_1^r[i] \neq \triangle TK_2^r[i] \neq \triangle TK_3^r[i] \neq 0$ for $i \in \{0,1,\cdots,15\}$, then a 7-round related-tweakey impossible differential for `Joltik-BC-192` exists. The corresponding input and output differences are denoted by $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$ and $\Delta_{out} = \triangle TK_1^{r+7} \oplus \triangle TK_2^{r+7} \oplus \triangle TK_3^{r+7}$.

Case 2. The cancellation phenomenon is observed in the final two rounds of the distinguisher. Specifically, for $i \in \{0,1,\cdots,15\}$, if the difference of tweakey satisfies the conditions that $\triangle TK_1^r[i] \oplus KS^5(\triangle TK_2^r[i], 2) \oplus \triangle KS^5(\triangle TK_3^r, 4) = 0$ and $\triangle TK_1^r[i] \oplus KS^6(\triangle TK_2^r[i], 2) \oplus KS^6(\triangle TK_3^r, 4) = 0$, where $\triangle TK_1^r[i] \neq \triangle TK_2^r[i] \neq \triangle TK_3^r[i] \neq 0$, then a 7-round related-tweakey impossible differential for `Joltik-BC-192` exists. The input difference Δ_{in} is given by $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$, and the output difference Δ_{out} is given by $\Delta_{out} = \triangle TK_1^{r+7} \oplus \triangle TK_2^{r+7} \oplus \triangle TK_3^{r+7}$.

[7] The size of the search space is about $(16 * 15)^3 \approx 2^{23.7}$

Table 3: The 7-round related-tweakey impossible differentials for `Joltik-BC-192`

Cases	$(\triangle TK_1^r[i], \triangle TK_2^r[i], \triangle TK_3^r[i])$	Index
Case 1	$(1,4,F)$,$(2,8,D)$,$(3,C,2)$,$(4,3,9)$,$(5,7,6)$ $(6,B,4)$,$(7,F,B)$,$(8,6,1)$,$(9,2,E)$,(A,E,C) $(B,A,3)$,$(C,5,8)$,$(D,1,7)$,$(E,D,5)$,$(F,9,A)$	$i \in \{0,1,2,\cdots,15\}$
Case 2	$(1,D,3)$,$(2,9,6)$,$(3,4,5)$,$(4,1,C)$,$(5,C,F)$ $(6,8,A)$,$(7,5,9)$,$(8,2,B)$,$(9,F,8)$,(A,B,D) $(B,6,E)$,$(C,3,7)$,$(D,E,4)$,$(E,A,1)$,$(F,7,2)$	$i \in \{0,1,2,\cdots,15\}$
Case 3	$(1,3,5)$,$(2,6,A)$,$(3,5,F)$,$(4,C,7)$,$(5,F,2)$ $(6,A,D)$,$(7,9,8)$,$(8,B,E)$,$(9,8,B)$,$(A,D,4)$ $(B,E,1)$,$(C,7,9)$,$(D,4,C)$,$(E,1,3)$,$(F,2,6)$	$i \in \{0,1,2,\cdots,15\}$
Case 4	$(1,C,9)$,$(4,6,F)$,$(F,8,E)$,$(F,6,8)$	$i \in \{0,2,4,\cdots,14\}$
Case 5	$(3,6,7)$,$(4,9,C)$,$(D,6,D)$,$(F,A,2)$	$i \in \{1,3,5,\cdots,15\}$

Case 3. The cancellation phenomenon is observed in the second and sixth rounds of the distinguisher. Specifically, for $i \in \{0,1,\cdots,15\}$, if the difference between the two tweakeys satisfies that $\triangle TK_1^r[i] \oplus KS^1(\triangle TK_2^r[i],2) \oplus KS^1(\triangle TK_3^r,4) = 0$ and $\triangle TK_1^r[i] \oplus KS^6(\triangle TK_2^r[i],2) \oplus \triangle KS^6(\triangle TK_3^r,4) = 0$, with $\triangle TK_1^r[i] \neq \triangle TK_2^r[i] \neq \triangle TK_3^r[i] \neq 0$, then a 7-round related-tweakey impossible differential exists for the `Joltik-BC-192`, where $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$ and $\Delta_{out} = \triangle TK_1^{r+7} \oplus \triangle TK_2^{r+7} \oplus \triangle TK_3^{r+7}$.

Case 4. The cancellation occurs in the sixth round. Specifically, the input difference $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$ cannot propagate to the output difference $\Delta_{out} = \triangle TK_1^{r+7} \oplus \triangle TK_2^{r+7} \triangle TK_3^{r+7}$ after 7-round encryption when the initial difference of the tweakey satisfies that $(\triangle TK_1^r[i], \triangle TK_2^r[i], \triangle TK_3^r[i]) \in \{(1,C,9),(4,6,F),(F,8,E),(F,6,8)\}$ for $i \in \{0,2,4,\cdots,14\}$, where we find that $\triangle TK_1^r[i] \oplus KS^6(\triangle TK_2^r[i],2) \oplus \triangle KS^6(\triangle TK_3^r,4) = 0$.

Case 5. The cancellation occurs in the second round. Specifically, the input difference $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$ cannot propagate to the output difference $\Delta_{out} = \triangle TK_1^{r+7} \oplus \triangle TK_2^{r+7} \oplus \triangle TK_3^{r+7}$ after 7-round encryption if the input differences of tweakeys satisfies that $(\triangle TK_1^r[i], \triangle TK_2^r[i], \triangle TK_3^r[i]) \in \{(3,6,7),(4,9,C),(D,6,D),(F,A,2)\}$ for $i \in \{1,3,5,\cdots,15\}$, where we find that $\triangle TK_1^r[i] \oplus KS^1(\triangle TK_2^r[i],2) \oplus KS^1(\triangle TK_3^r,4) = 0$.

4.2 Application to `SKINNY`

`SKINNY` is a family of lightweight tweakable block ciphers designed to have the smallest hardware footprint, which was proposed at CRYPTO 2016 by Beierle et al. [3]. It has 6 main variants for `SKINNY`. Particularly, `SKINNY-`n-t is a block cipher that operates on n-bit blocks with t-bit tweakey, where $n = 64$ or 128 and $t = n, 2n$ or $3n$. More details can be found in [3]. This section will apply

our model in Algorithm 2 to search the related-tweakey impossible differential for SKINNY.

Previous Cryptanalysis. To the best of our knowledge, the longest related-tweakey impossible differentials obtained assuming a single active nibble are 12-, 14-, and 16-round for SKINNY-64-64, SKINNY-64-128, and SKINNY-64-192, respectively, as reported in [25]. Although Sadeghi et al. [30] claimed that they found 13- and 15-round related-tweakey impossible differential for SKINNY-64-64 and SKINNY-64-128, the length of distinguishers in the mode of $(0, 0, \delta)$ was the same as our results. In their results, the extra round was not eligible in our opinion because the input difference of the extra round is not certain.

The 12-Round Related-tweakey Impossible Differentials for SKINNY-64-64. By introducing the difference to one nibble of TK_1^r, we apply Algorithm 2 to find a 10-round related-tweakey impossible differential (including 10 SubCells operations) with 817.69 seconds. It took about 1.01 hours to prove that there is no 11-round impossible differential in the search space. According to the relationship between the tweakey schedule and the round function, we can further extend the 10-round related-tweakey impossible differentials to the 12-round related-tweakey impossible differentials in the mode of (α, β, δ), which is shown in Table 4.

Table 4: The related-tweakey impossible differentials for SKINNY-64-64

Num.	$\Delta_{in} = \Delta TK_1^r$	Δ_{out}	ConR	ConPs
RTK01	$a000\,0000\,0000\,0000$	$0000\,000a\,0000\,0000$		S_8
RTK02	$0a00\,0000\,0000\,0000$	$000a\,0000\,0000\,0000$		$S_{4,10,12}$
RTK03	$00a0\,0000\,0000\,0000$	$0a00\,0000\,0000\,0000$		$S_{8,9}$
RTK04	$000a\,0000\,0000\,0000$	$0000\,00a0\,0000\,0000$	7	S_7
RTK05	$0000\,0a00\,0000\,0000$	$0000\,a000\,0000\,0000$		$S_{9,15}$
RTK06	$0000\,00a0\,0000\,0000$	$00a0\,0000\,0000\,0000$		$S_{9,13}$
RTK07	$0000\,000a\,0000\,0000$	$0000\,0a00\,0000\,0000$		$S_{5,13}$

The 14-Round Related-tweakey Impossible Differentials for SKINNY-64-128. By introducing differences to the same nibble of TK_1^r and TK_2^r, we have discovered a 12-round related-tweakey impossible differential with a duration of 5.96 hours using Algorithm 2. It took approximately 26.89 hours to establish the absence of a 13-round impossible differential in the search space. Based on the relationship between the tweakey schedule and the round function, we have extended the 12-round related-tweakey impossible differentials in the $(0, 0, \delta)$ mode to 14-round related-tweakey impossible differentials in the (α, β, δ) mode. Here, $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r$, $\triangle TK_1^r \oplus L_2(\triangle TK_2^r) = 0$, and $\Delta_{out} = \triangle TK_1^{r+14} \oplus \triangle TK_2^{r+14}$. The values are presented in Table 5.

The 16-Round Related-tweakey Impossible Differentials for SKINNY-64-192. By introducing the differences to the same nibble of TK_1^r, TK_2^r, and

Table 5: The related-tweakey impossible differentials for SKINNY-64-128

i	$(\Delta TK_1^r[i], \Delta TK_2^r[i])$	Δ_{in}	Δ_{out}	ConR	ConPs
0	$(1,8),(2,1),(3,9)$ $(4,2),(5,A),(6,3)$ $(7,B),(8,C),(9,4)$ $(A,D),(B,5),(C,E)$ $(D,6),(E,F),(F,7)$	$a000\,0000\,0000\,0000$	$0b00\,0000\,0000\,0000$	8	$S_{4,8,12}$
1		$0a00\,0000\,0000\,0000$	$0000\,000b\,0000\,0000$		$S_{9,12}$
2		$00a0\,0000\,0000\,0000$	$b000\,0000\,0000\,0000$		$S_{14,15}$
3		$000a\,0000\,0000\,0000$	$0000\,0b00\,0000\,0000$		$S_{1,7,14,15}$
4		$0000\,a000\,0000\,0000$	$00b0\,0000\,0000\,0000$		$S_{11,13,14}$
5		$0000\,0a00\,0000\,0000$	$0000\,00b0\,0000\,0000$		$S_{3,8,9}$
6		$0000\,00a0\,0000\,0000$	$0000\,b000\,0000\,0000$		$S_{5,9,12}$
7		$0000\,000a\,0000\,0000$	$000b\,0000\,0000\,0000$		$S_{0,5,13}$

TK_3^r, respectively, we applied our tool to discover the 14-round related-tweakey impossible differential with 6.9 days in the search space. Moreover, we extended the 14-round related-tweakey impossible differentials in the mode of $(0,0,\delta)$ to the 16-round related-tweakey impossible differentials in the mode of (α,β,δ), where $\Delta_{in} = \triangle TK_1^r \oplus \triangle TK_2^r \oplus \triangle TK_3^r$ and $\Delta_{out} = \triangle TK_1^{r+16} \oplus \triangle TK_2^{r+16} \oplus \triangle TK_2^{r+16}$. Due to the cancellation among the differences of the tweakeys, the search results can be divided into two cases. Table 6 presents the experimental outcomes for $\Delta_{in} = (a000, 0000, 0000, 0000)$ and $\Delta_{out} = (b000, 0000, 0000, 0000)$.

Case 1. The values of $(\triangle TK_1^r, \triangle TK_2^r, \triangle TK_3^r)$ are subject to the constraint that $\triangle TK_1^r[i] \oplus L_2^1(\triangle TK_2^r[i]) \oplus L_3^1(\triangle TK_3^r[i]) = 0$ and $\triangle TK_1^r[i] \oplus L_2^2(\triangle TK_2^r[i]) \oplus L_3^2(\triangle TK_3^r[i]) = 0$, where $i \in \{0, \cdots, 7\}$. As a result, no differences are introduced for the first six rounds. The propagation of differentials in this scenario is illustrated in Fig.3 in Appendix A.

Case 2. The tuple of values $(\triangle TK_1^r, \triangle TK_2^r, \triangle TK_3^r)$ is constrained such that $\triangle TK_1^r[i] \oplus L_1^1(\triangle TK_2^r[i]) \oplus L_1^1(\triangle TK_3^r[i]) = 0$ and $\triangle TK_1^r[i] \oplus L_2^7(\triangle TK_2^r[i]) \oplus L_3^7(\triangle TK_3^r[i]) = 0$ for $i \in \{0, \cdots, 7\}$. The proof of this assertion resembles that of Case 1, where no difference is introduced to the first and last four rounds.

Table 6: The related-tweakey impossible differentials for SKINNY-64-192

Case	$(\Delta TK_1^r[0], \Delta TK_2^r[0], \Delta TK_3^r[0])$	ConR	ConPs
Case 1	$(1,7,C),(2,F,8),(3,8,4),(4,E,1),(5,9,D)$ $(6,1,9),(7,6,5),(8,9,E),(9,C,2),(A,4,6)$ $(B,3,A),(C,5,F),(D,2,3),(E,A,7),(F,D,B)$	9	S_{14}
Case 2	$(1,2,B),(2,4,7),(3,6,C),(4,9,F),(5,B,4)$ $(6,D,8),(7,F,3),(8,1,5),(9,3,E),(A,5,2)$ $(B,7,9),(C,8,A),(D,A,1),(E,C,D),(F,E,6)$	9	$S_{8,9}$

4.3 Application to QARMA

The QARMA block cipher, designed by Avanzi at ToSC'17, is a lightweight tweakable block cipher with three-round Even-Mansour construction. There are two

variants of QARMA that support block sizes of $n = 64$ and $n = 128$ bits, denoted by QARMA-64 and QARMA-128, respectively. The tweak is also n bits long and the key is always $2n$ bits long. In the present paper, we pay attention to QARMA-64.

Previous Cryptanalysis. Since the proposal of the tweakable block cipher QARMA, various attacks have been employed to assess its security, such as meet-in-the-middle attacks [22], impossible differential attacks [35,36,26] and statistical saturation attacks [21]. However, the longest related-tweak impossible differential of QARMA is 7-round proposed by Zong et al. [36] by considering the differential relationship between the tweak and a single-tweak impossible differential.

List of 7-round Related-tweakey Impossile Differentials for QARMA-64. By modifying a single nibble in the initial tweak, we apply Algorithm 2 to derive several related-tweakey impossible differentials for QARMA-64, ranging from the 7th to the 11th round, some of which were not previously discovered. By taking into account the impact of the tweak update function, we further obtain some 7-round related-tweakey impossible differentials for QARMA-64, which is covering rounds from the 6th to the 12th, as tabulated in Table 7. We take the Num.RT03 in Table 7 as an example to verify the correctness with the "miss-in-the-middle" method, which is shown in Fig.4 of Appendix A.

Table 7: The 7-round related-tweak impossible differentials for QARMA-64

Num.	$\Delta_{in} = \Delta T$	Δ_{out}	ConR	ConPs
RT01	$a000\,0000\,0000\,0000$	$0000\,0000\,0000\,00b0$	9	S_1
RT02	$000a\,0000\,0000\,0000$	$0000\,c000\,0000\,0000$		S_{13}
RT03	$0000\,0a00\,0000\,0000$	$0c00\,0000\,0000\,0000$		S_{14}
RT04	$0000\,0000\,00a0\,0000$	$0000\,0000\,0000\,b000$		S_1
RT05	$0000\,0000\,000a\,0000$	$0000\,00c0\,0000\,0000$		S_{10}
RT06	$0000\,0000\,0000\,0a00$	$0000\,0000\,b000\,0000$		S_5

$a \in \mathbb{F}_{2^4} \setminus \{0\}$, $b = \overline{\omega}(a)$ and $c = \overline{\omega}^2(a)$, where $\overline{\omega} = \omega^{-1}$.

4.4 Application to CRAFT

CRAFT is a lightweight tweakable block cipher introduced by Beierle et al. [4] at FSE 2019, which follows the SPN design with 32 rounds. The main goal of CRAFT was to efficiently protect its implementations against Differential Fault Analysis (DFA) attacks. It consists of a 64-bit block, a 128-bit key K and 64-bit tweak T, where the 128-bit key is split into two 64-bit keys K_0 and K_1. Using the permutation Q on the tweak, four 64-bit tweakeys TK_0, TK_1, TK_2 and TK_3 are derived from the tweak T and keys K_0, K_1. Then in each round, without any key update, the tweakey $TK_{i \bmod 4}$ is XORed to the cipher state. More information can be obtained in [4].

Previous Cryptanalysis. In the specification file, Hadipour et al. [4] conducted an extensive analysis of the security of CRAFT. Specifically, they identified

the 13-round impossible differential under the single-key setting as the longest one in the analysis until now. Subsequently, many studies have been conducted to evaluate the security of round-reduced CRAFT under both the single-key mode and related-key mode. However, the majority of research has been centered on differential attacks, as documented in [11,8,10]. Furthermore, Hadipour et al. [11] have reported a 14-round zero-correlation linear distinguisher under the related-tweak setting in previous research, in addition to some probability-type attacks.

List of 12-round Related-tweak Impossible Differentials for CRAFT. When searching the related-tweak impossible differentials for CRAFT, we activate a single nibble of the initial tweak while other differences remain inactive. Specifically, the active set is denoted as $\Omega = \{(0, 0, \delta) | \delta \in \mathbb{F}_2^{\kappa+t} \backslash \{0\}\}$, $\triangle K_0 = \triangle K_1 = 0$, and $\triangle T = \delta$. By utilizing Algorithm 2, we discovered several 10-round related-tweak impossible differentials for the first time in a total time of 891.34 seconds, which also can be extended to 12-round, as shown in Table 8. We take the Num.RT01 in Table 7 as an example to verify the correctness with the "miss-in-the-middle" method, which is shown in Fig.5 of Appendix A. Additionally, we have proven that there are no 13-round related-tweak impossible differentials in the search space, which required a total time of 4698.06 seconds..

Table 8: The 12-round related-tweak impossible differentials for CRAFT

Num	$\Delta_{in} = \Delta T$	Δ_{out}	ConR	ConPs
RT01	$0a00\ 0000\ 0000\ 0000$	$0a00\ 0000\ 0000\ 0000$	7	S_6
RT02	$0000\ 0a00\ 0000\ 0000$	$0000\ 0a00\ 0000\ 0000$		S_{10}
RT03	$0000\ 0000\ a000\ 0000$	$0000\ 0000\ a000\ 0000$		S_{10}

List of 15-round Related-tweakey Impossible Differentials for CRAFT. By setting the input and output differences to be zero and modify only one single nibble of K_0, K_1, and T, i.e. $\Omega = \{(0, 0, \delta) | \delta \in \mathbb{F}_2^t \setminus \{0\}\}$ and $\triangle K_0 = \triangle K_1 = \triangle T = \delta$, we apply Algorithm 2 to derive the 13-round related-tweakey impossible differentials for CRAFT for the first time within 3263.46 seconds, which also can be extended to the 15-round. The search results are summarized in Table 9. Additionally, we have proven that there are no 16-round related-tweakey impossible differentials within the search space, with a total search time of 7040.3 seconds.

Table 9: The 15-round related-tweakey impossible differentials for CRAFT

$(\Delta T, \Delta K_0, \Delta K_1)$	Δ_{in}	Δ_{out}	ConR	ConPs
$0000\ 0000\ 000a\ 0000$ $0000\ 0000\ 000a\ 0000$ $0000\ 0000\ 000a\ 0000$	$a00a\ 0000\ a00a\ 0000$	$0000\ 0000\ a00a\ 0000$	9	S_4 S_{10}

5 Evaluation of the Automatic Search Models

The LCA technique is an analysis method explicates the complete attributes by way of partial features. Consequently, compared with traditional search methods, utilizing the LCA technique can alleviate the interdependence among variables. Subsequently, we will present an assessment of Algorithm 2 compared with Algorithm 1 based on the search results.

Improving the Search Efficiency for Long Trials. The utilization of the LCA technique may enhance search efficiency and significantly reduce time costs, especially when exploring distinguishers with long trails. An illustrative example is provided in Table 10, which presents the computational time required for Algorithm 1 and Algorithm 2 to ascertain the existence of a related-tweakey impossible differential for `CRAFT`. The experimental evaluation was performed on the platform: Inter(R) Core i7-9700 CPU@3.00GHz×8, 8GB RAM, 64-bit Ubuntu VMware. As evidenced by Table 10, when the number of rounds is limited, Algorithm 2 must sequentially traverse the constraint set and intermediate rounds, resulting in a total time cost comparable to Algorithm 1. However, as the number of rounds increases, the time complexity of Algorithm 1 escalates nearly exponentially, whereas Algorithm 2 maintains a relatively constant and gradual growth trend.

Table 10: The time for the related-tweakey impossible differentials of `CRAFT`

Scenario	DR	Alg.1	Alg.2	Results
CRAFT-RT	11	425.18s	330.06s	find 11-round RTID.
	12	1752.21s	1466.48s	find 12-round RTID.
	13	84102.11s	1998.46s	find no 13-round RTID.
CRAFT-RTK	14	3528.08s	2902.99s	find 14-round RTKID.
	15	4424.64s	3263.46s	find 15-round RTKID.
	16	−−	7040.30s	find no 16-round RTKID.

RT: Related-tweak impossible differential. RTK: Related-tweakey impossible differential. **"−−":** Terminating the program because it took too long to run.

Additionally, Algorithm 2 exhibits considerably superior performance to Algorithm 1 when applied to the cipher `SKINNY`, as indicated in Table 11. However, it should be noted that Algorithm 2 does not consistently outperform Algorithm 1. Specifically, in scenarios where the length of the distinguisher is relatively short for `QARMA` and `Joltik-BC`, Algorithm 2 provides a lesser advantage over Algorithm 1 when searching for distinguishers. For instance, in the case of `QARMA`, Algorithm 1 required 1631.37 seconds to establish the absence of 8-round related-tweak impossible differentials, whereas Algorithm 2 necessitated 1624.66 seconds. In this particular case, the search efficiency was comparable. However, the discrepancy in efficiency becomes evident for `Joltik-BC-128`, where Algorithm 1 required 84447.57 seconds to prove the nonexistence of 7-round related-tweakey impossible differentials, whereas Algorithm 2 demanded 476278.89 seconds.

Table 11: The time for related-tweakey impossible differentials of SKINNY

(n,t)	DR	Alg.1		Alg.2		Results
		Single	Total	Single	Total	
$(64,64)$	12	26950s	–	6.61s	817.69s	find 12-round RTKID.
	13	−−	−−	16.23s	3643.37s	find no 13-round RTKID.
$(64,128)$	14	104.17s	−−	13.01s	21439.97s	find 14-round RTKID.
	15	−−	−−	31.00s	96791.32s	find no 15-round RTKID.
$(64,192)$	14	30.71s	1483.65s	15.34s	732.44s	find 14-round RTKID.
	15	32.36s	1727.89s	34.00s†	1744.75s	find 15-round RTKID.
	16	258709s	−−	20.24s†	599280s	find 16-round RTKID.

Single: The time it takes to complete a search, not the average time. Total: The total time it took to find the first distinguisher while traversing the search space. "†": In this case, we choose the middle round as r_m for odd-numbered rounds, while middle two rounds for even-numbered rounds. So this time is reasonable.

Determining the Contradictory Positions. In cryptanalysis, the "miss-in-the-middle" method has traditionally been employed to manually deduce the contradictory positions of an impossible differential. However, the process becomes challenging if the length of a distinguisher is too long or the cipher with sound diffusions. Therefore, there is a need for automatic tools to assist in determining the locations of contradictions. To this end, similar to the one used for verifying impossible differential distinguishers in [7] and [12], the LCA technique can be also used to derive the contradictory positions. Specifically, if there exists an impossible differential under the constraint set $\mathbb{P}$, then the contradictory occurs in the positions of $\mathbb{P}$. Here, we provide an example of SIMON128, which is obtained by Algorithm 2.

Example 1. The differential $(0x0000000000000000, 0x8000000000000000) \nrightarrow (0x4000000000000000, 0x0000000000000000)$ is a 19-round impossible differential for SIMON128, where the contradictory occurs in the second bit of the 11-th round.

6 Conclusion

This paper evaluates the security of tweakable block ciphers against the related-tweakey impossible differential analysis. The main approach involves constructing a differential propagation system using the SAT method, which describes the propagation of corresponding states under specific constraints and determines whether the transition is invalid. To achieve this goal, an automatic search model is proposed for related-tweakey impossible differentials based on the SMT problem. Subsequently, this method has been employed to identify the related-tweakey impossible differentials for QARMA-64 and Joltik-BC, respectively.

Furthermore, the paper introduces a novel analytical strategy known as Locality Constraint Analysis (LCA), which aims to improve the efficiency of searching the distinguisher with long trails or ciphers with large size. A generalized automatic search model is constructed based on LCA, and the proposed method

is applied to various ciphers such as SIMON, SPECK, QARMA, CRAFT, Joltik-BC, and SKINNY. Based on the search results, it is demonstrated that introducing the LCA technique to impossible differential cryptanalysis significantly improves the search efficiency and provides much more convenience for deriving the locations of the contradictory positions. Additionally, the LCA technique also can be used to searching the tranditional impossible differential, even though no specific example is provided in this paper.

Acknowledgements

We thank the associate editor and the anonymous reviewers for their useful feedback that improved this paper. This research was supported by the National Natural Science Foundation of China (Grant No. 12371525) and the National Key Research and Development Program of China (Grant No. 2022YFF0604702).

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 central rounds, and search heuristics for low-latency s-boxes. IACR Transactions on Symmetric Cryptology pp. 4–44 (2017). https://doi.org/10.13154/tosc.v2017.i1.4-44
2. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The simon and speck lightweight block ciphers. In: Proceedings of the 52nd annual design automation conference. pp. 1–6 (2015). https://doi.org/10.1145/2744769.2747946
3. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The skinny family of block ciphers and its low-latency variant mantis. In: Advances in Cryptology–CRYPTO 2016: 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II 36. pp. 123–153. Springer (2016). https://doi.org/10.1007/978-3-662-53008-5_5
4. Beierle, C., Leander, G., Moradi, A., Rasoolzadeh, S.: Craft: lightweight tweakable block cipher with efficient protection against dfa attacks. IACR Transactions on Symmetric Cryptology **2019**(1), 5–45 (2019). https://doi.org/10.13154/tosc.v2019.i1.5-45
5. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In: Advances in CryptologyEUROCRYPT99: International Conference on the Theory and Application of Cryptographic Techniques Prague, Czech Republic, May 2–6, 1999 Proceedings 18. pp. 12–23. Springer (1999). https://doi.org/10.1007/3-540-48910-X_2
6. Chakraborti, A., Datta, N., Jha, A., Mancillas-López, C., Nandi, M., Sasaki, Y.: Elastic-tweak: a framework for short tweak tweakable block cipher. In: Progress in Cryptology–INDOCRYPT 2021: 22nd International Conference on Cryptology in India, Jaipur, India, December 12–15, 2021, Proceedings. pp. 114–137. Springer (2021). https://doi.org/10.1007/978-3-030-92518-5_6

7. Cui, T., Chen, S., Jia, K., Fu, K., Wang, M.: New automatic search tool for impossible differentials and zero-correlation linear approximations. Sci. China Inf. Sci. **64**(2) (2021). https://doi.org/10.1007/s11432-018-1506-4
8. ElSheikh, M., Youssef, A.M.: Related-key differential cryptanalysis of full round craft. In: Security, Privacy, and Applied Cryptography Engineering: 9th International Conference, SPACE 2019, Gandhinagar, India, December 3–7, 2019, Proceedings. pp. 50–66. Springer (2019)
9. Guo, C., Guo, J., List, E., Song, L.: Towards closing the security gap of tweak-and-tweak (tnt). In: Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part I 26. pp. 567–597. Springer (2020). https://doi.org/10.1007/978-3-030-64837-4_19
10. Guo, H., Sun, S., Shi, D., Sun, L., Sun, Y., Hu, L., Wang, M.: Differential attacks on craft exploiting the involutory s-boxes and tweak additions. IACR Trans. Symmetric Cryptol. **2020**(3), 119–151 (2020). https://doi.org/10.13154/tosc.v2020.i3.119-151
11. Hadipour, H., Sadeghi, S., Niknam, M.M., Song, L., Bagheri, N.: Comprehensive security analysis of craft. IACR Transactions on Symmetric Cryptology pp. 290–317 (2019). https://doi.org/10.13154/tosc.v2019.i4.290-317
12. Hu, X., Li, Y., Jiao, L., Tian, S., Wang, M.: Mind the propagation of states: New automatic search tool for impossible differentials and impossible polytopic transitions. In: Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part I 26. pp. 415–445. Springer (2020). https://doi.org/10.1007/978-3-030-64837-4_14
13. Jean, J., Nikolić, I., Peyrin, T.: Tweaks and keys for block ciphers: the tweakey framework. In: Advances in Cryptology–ASIACRYPT 2014: 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, ROC, December 7-11, 2014, Proceedings, Part II 20. pp. 274–288. Springer (2014). https://doi.org/10.1007/978-3-662-45608-8_15
14. Jean, J., Nikolic, I., Peyrin, T.: Joltik v1.3. Submission to the CAESAR competition (2015), https://competitions.cr.yp.to/round2/joltikv13.pdf
15. Jean, J., Nikolić, I., Peyrin, T., Seurin, Y.: The deoxys aead family. Journal of Cryptology **34**(3), 31 (2021). https://doi.org/10.1007/s00145-021-09397-w
16. Kim, J., Hong, S., Lim, J.: Impossible differential cryptanalysis using matrix method. Discrete Mathematics **310**(5), 988–1002 (2010). https://doi.org/10.1016/j.disc.2009.10.019
17. Knudsen, L.: Deal - a 128-bit block cipher. NISI AES Proposal (1998)
18. Krovetz, T., Rogaway, P.: The software performance of authenticated-encryption modes. In: Fast Software Encryption: 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers 18. pp. 306–327. Springer (2011). https://doi.org/10.1007/978-3-642-21702-9_18
19. Leventi-Peetz, A.M., Zendel, O., Lennartz, W., Weber, K.: Cryptominisat switches-optimization for solving cryptographic instances. arXiv preprint arXiv:2112.11484 (2021)
20. Li, M., Chen, S.: Improved meet-in-the-middle attacks on reduced-round joltik-bc. IET Information Security **15**(3), 247–255 (2021)
21. Li, M., Hu, K., Wang, M.: Related-tweak statistical saturation cryptanalysis and its application on qarma. IACR Trans. Symmetric Cryptol. **2019**(1), 236–263 (2019). https://doi.org/10.13154/tosc.v2019.i1.236-263

22. Li, R., Jin, C.: Meet-in-the-middle attacks on reduced-round qarma-64/128. The Computer Journal **61**(8), 1158–1165 (2018)
23. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable block ciphers. In: Advances in CryptologyCRYPTO 2002: 22nd Annual International Cryptology Conference Santa Barbara, California, USA, August 18–22, 2002 Proceedings 22. pp. 31–46. Springer (2002). https://doi.org/10.1007/3-540-45708-9_3
24. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable block ciphers. Journal of cryptology **24**, 588–613 (2011). https://doi.org/10.1007/s00145-010-9073-y
25. Liu, G., Ghosh, M., Song, L.: Security analysis of skinny under related-tweakey settings. Cryptology ePrint Archive (2016)
26. Liu, Y., Zang, T., Gu, D., Zhao, F., Li, W., Liu, Z.: Improved cryptanalysis of reduced-version qarma-64/128. IEEE Access **8**, 8361–8370 (2020). https://doi.org/10.1109/ACCESS.2020.2964259
27. Luo, Y., Lai, X., Wu, Z., Gong, G.: A unified method for finding impossible differentials of block cipher structures. Information Sciences **263**, 211–220 (2014). https://doi.org/10.1016/j.ins.2013.08.051
28. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Information Security and Cryptology: 7th International Conference, Inscrypt 2011, Beijing, China, November 30–December 3, 2011. Revised Selected Papers 7. pp. 57–76. Springer (2012). https://doi.org/10.1007/978-3-642-34704-7_5
29. Peyrin, T., Seurin, Y.: Counter-in-tweak: authenticated encryption modes for tweakable block ciphers. In: Advances in Cryptology–CRYPTO 2016: 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part I. pp. 33–63. Springer (2016)
30. Sadeghi, S., Mohammadi, T., Bagheri, N.: Cryptanalysis of reduced round skinny block cipher. IACR Transactions on Symmetric Cryptology pp. 124–162 (2018). https://doi.org/10.13154/tosc.v2018.i3.124-162
31. Sasaki, Y., Todo, Y.: New impossible differential search tool from design and cryptanalysis aspects: Revealing structural properties of several ciphers. In: Advances in Cryptology–EUROCRYPT 2017: 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30–May 4, 2017, Proceedings, Part III 36. pp. 185–215. Springer (2017). https://doi.org/10.1007/978-3-319-56617-7_7
32. Schroeppel, R., Orman, H.: The hasty pudding cipher. AES candidate submitted to NIST p. M1 (1998)
33. Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L.: Automatic security evaluation and (related-key) differential characteristic search: application to simon, present, lblock, des (l) and other bit-oriented block ciphers. In: Advances in Cryptology–ASIACRYPT 2014: 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, ROC, December 7-11, 2014. Proceedings, Part I 20. pp. 158–178. Springer (2014). https://doi.org/10.1007/978-3-662-45611-8_9
34. Wu, S., Wang, M.: Automatic search of truncated impossible differentials for word-oriented block ciphers. In: Progress in Cryptology-INDOCRYPT 2012: 13th International Conference on Cryptology in India, Kolkata, India, December 9-12, 2012. Proceedings 13. pp. 283–302. Springer (2012). https://doi.org/10.1007/978-3-642-34931-7_17
35. Yang, D., Qi, W.F., Chen, H.J.: Impossible differential attack on qarma family of block ciphers. Cryptology ePrint Archive (2018)

36. Zong, R., Dong, X.: Milp-aided related-tweak/key impossible differential attack and its applications to qarma, joltik-bc. IEEE Access **7**, 153683–153693 (2019). https://doi.org/10.1109/ACCESS.2019.2946638

A Related-tweakey Impossible Differential Analysis

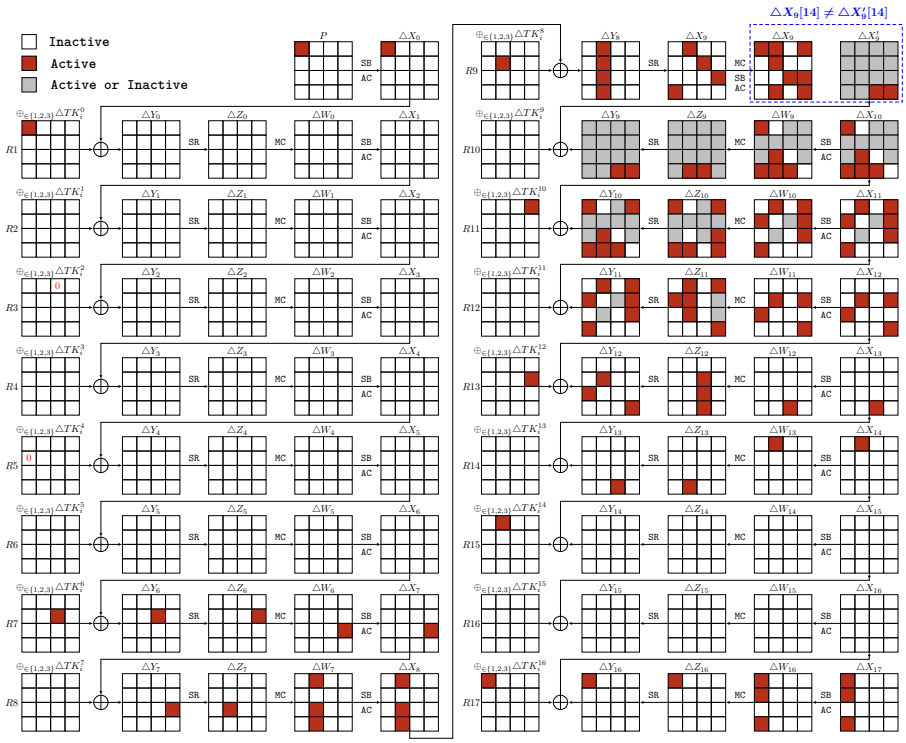

Fig. 3: The 16-round Related-tweakey impossible differential for `SKINNY-64-192`.

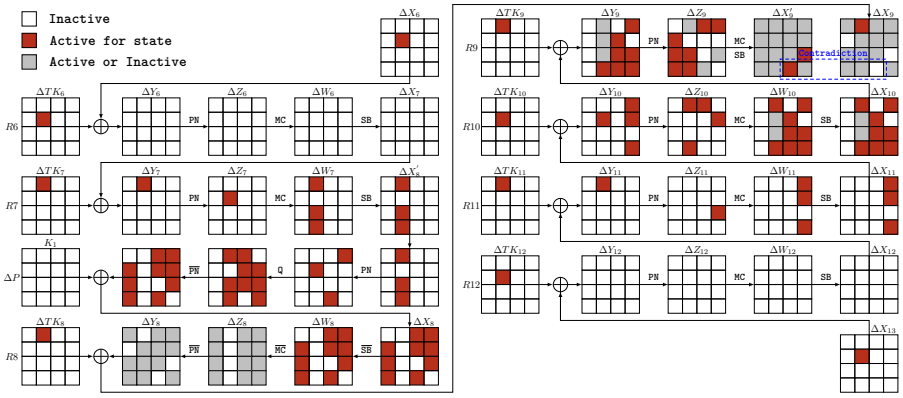

Fig. 4: The 7-round related-tweak impossible differential for QARMA-64

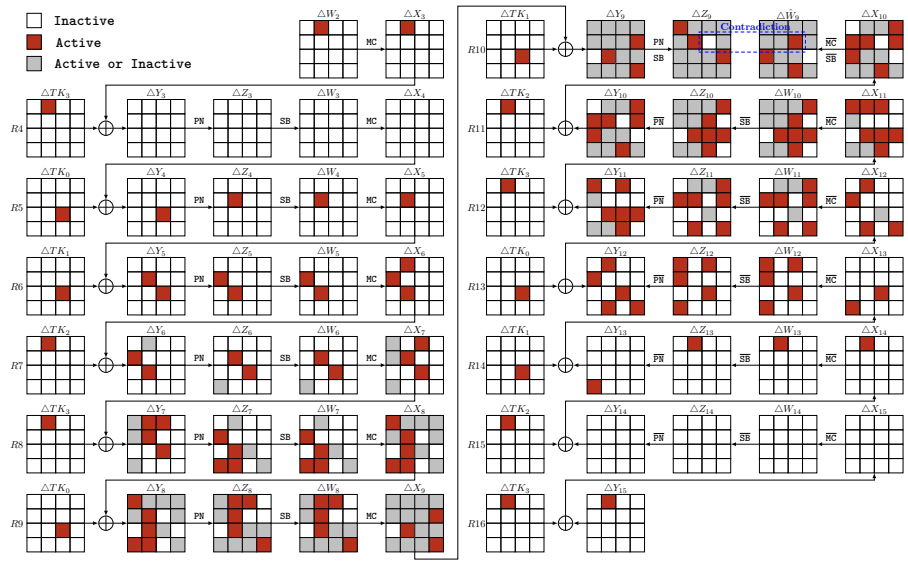

Fig. 5: The 12-round Related-tweak impossible differential for CRAFT.