IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2021, No. 1, pp. 269–315. DOI:10.46586/tosc.v2021.i1.269-315

Accelerating the Search of Differential and Linear Characteristics with the SAT Method

Ling Sun[1,2], Wei Wang[1,2] and Meiqin Wang[1,2](✉)

[1] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China
[2] School of Cyber Science and Technology, Shandong University, Qingdao, China
lingsun@sdu.edu.cn, weiwangsdu@sdu.edu.cn, mqwang@sdu.edu.cn

Abstract. The introduction of the automatic search boosts the cryptanalysis of symmetric-key primitives to some degree. However, the performance of the automatic search is not always satisfactory for the search of long trails or ciphers with large state sizes. Compared with the extensive attention on the enhancement for the search with the mixed integer linear programming (MILP) method, few works care for the acceleration of the automatic search with the Boolean satisfiability problem (SAT) or satisfiability modulo theories (SMT) method. This paper intends to fill this vacancy. Firstly, with the additional encoding variables of the sequential counter circuit for the original objective function in the standard SAT method, we put forward a new encoding method to convert the Matsui's bounding conditions into Boolean formulas. This approach does not rely on new auxiliary variables and significantly reduces the consumption of clauses for integrating multiple bounding conditions into one SAT problem. Then, we evaluate the accelerating effect of the novel encoding method under different sets of bounding conditions. With the observations and experience in the tests, a strategy on how to create the sets of bounding conditions that probably achieve extraordinary advances is proposed. The new idea is applied to search for optimal differential and linear characteristics for multiple ciphers. For `PRESENT`, `GIFT-64`, `RECTANGLE`, `LBlock`, `TWINE`, and some versions in `SIMON` and `SPECK` families of block ciphers, we obtain the complete bounds (full rounds) on the number of active S-boxes, the differential probability, as well as the linear bias. The acceleration method is also employed to speed up the search of related-key differential trails for `GIFT-64`. Based on the newly identified 18-round distinguisher with probability 2^{-58}, we launch a 26-round key-recovery attack with $2^{60.96}$ chosen plaintexts. To our knowledge, this is the longest attack on `GIFT-64`. Lastly, we note that the attack result is far from threatening the security of `GIFT-64` since the designers recommended users to double the number of rounds under the related-key attack setting.

Keywords: Automatic search · SAT method · Differential cryptanalysis · Linear cryptanalysis · Matsui's bounding condition

1 Introduction

Differential [BS90] and linear [Mat93] cryptanalyses can be seen as the cornerstone of modern cryptanalysis techniques for symmetric-key ciphers. Resistance against these two attacks is regarded as the baseline in the design of new primitives. The first step for the evaluation of the security against these attack is to find differential and linear trails with non-random behaviours. Shortly after the introduction of linear cryptanalysis, Matsui [Mat94] proposed a branch-and-bound depth-first searching algorithm that can be used to identify the optimal differentials with the maximum probability of symmetric-key primitives.

Received: 2020-11-23 Accepted: 2021-01-23 Published: 2021-03-19

The advantage of this algorithm is enhanced by taking in the customised optimisation for the specific cipher, which puts a high demand for sophisticated programming skills.

At the beginning of the last decade, the automatic method came on the stage and showed incredible performances in search of various distinguishers in cryptanalysis. The first category of the automatic search is based on the mixed integer linear programming (MILP) method, which was firstly introduced by Mouha et al. [MWGP11] to estimate the lower bound on the number of differential and linear active S-boxes. Later, this method was refined by Sun et al. [SHW+14] to search for (related-key) differential characteristics concerning bit-oriented block ciphers. Following that, the MILP method is further applied to accomplish tasks in search of multiple sorts of distinguishers, such as differential and linear characteristics for ARX ciphers [FWG+16], integral distinguishers [XZBL16], zero-correlation distinguishers [CJF+16], impossible differential distinguishers [ST17b], and non-blackbox polynomials manipulated in the cube attack [TIHM17].

Another important branch of the automatic search is based on the Boolean satisfiability problem (SAT) or the more general extension called satisfiability modulo theories (SMT) method. The initial work considering the usage of the SAT/SMT method in search of differential characteristics for ARX ciphers was proposed by Mouha and Preneel [MP13]. Also, this method is generalised to find various cryptanalytic distinguishers, including differential and linear characteristics for the `SIMON`-like round function [KLT15], linear trails for ARX ciphers [LWR16], and division properties for ARX ciphers [SWW17].

The automatic method enables users to write relatively simple codes to convert the distinguisher searching problem into the underlying mathematical problem, which can be handled by some openly available solvers. However, since the performance of the automatic search is tied to the power of the mathematical problem solver, the efficiency is not always satisfactory for the search of long trails or ciphers with large state sizes.

Many works aimed at an improvement in the efficiency of the MILP method, and we only name a few. Sasaki and Todo [ST17a] put forward a new algorithm that ensures the minimum number of inequalities for modelling S-boxes in search of differential characteristics. Furthermore, the relation between the number of inequalities and the runtime was studied, and they experimentally showed that minimising the number of inequalities does not always minimise the runtime. At ISC 2018, Zhang et al. [ZSCH18] incorporated the Matsui's bounding conditions into the MILP model and observed acceleration in search of differential trails for `PRESENT` [BKL+07] and `SIMON` [BSS+13]. Later, Li et al. [LWZZ19] investigated the relationship between the construction of the MILP model and the runtime. The results for `PRESENT` and `GIFT` [BPP+17] were updated by carefully elaborating the MILP model. With the central observation that high-probability differential/linear characteristics are likely to have a lower number of active S-boxes at a certain round, Zhou et al. [ZZDX19] came up with a divide-and-conquer approach to optimise the search with MILP. The whole searching space was split into several subspaces, and the MILP model was separately implemented on every subspace. At the same conference, Boura and Coggia [BC20] created efficient MILP models for S-boxes and linear layers of SPN ciphers and showed an impact on `AES` [DR02] and `SKINNY-128` [BJK+16].

Compared with the extensive attention regarding the improvement of the MILP method, few works consider the acceleration of the automatic search with the SAT/SMT method. As far as we know, the unique work related to this topic is proposed by Song et al. [SHY16]. They practised a splicing heuristic method to find better differential trails for ARX ciphers. Consequently, this paper is motivated by this vacancy and endeavours to speed up the search with the SAT method.

1.1 Our Contributions

In this paper, we study how to accelerate the search of differential and linear characteristics with the SAT method. In light of the enhanced performance of the MILP method [ZSCH18]

with Matsui's bounding conditions, we wonder the feasibility of integrating the bounding condition into the SAT method. Centred with this issue, the contributions of this paper can be classified into four parts.

Novel method to encode Matsui's bounding conditions. The standard SAT method applies the sequential encoding method to realise the transformation of the Boolean cardinality constraint $\sum_{j=0}^{n-1} x_j \leqslant k$ with $\mathcal{O}(n \cdot k)$ variables and clauses. Thus, for the constraint $\sum_{j=e_1}^{e_2} x_j \leqslant m$ corresponding to Matsui's bounding condition, the direct conversion by reusing the previous method consumes $\mathcal{O}((e_2 - e_1) \cdot m)$ variables and clauses. Nevertheless, when multiple bounding conditions are considered, this direct approach will notably raise the number of variables and clauses in the SAT problem, which may result in a negative influence on the efficiency of the searching phase. To overcome this shortcoming, we put forward a new method that manipulates the additional encoding variables of the sequential counter circuit for the original objective function. Without introducing any new variables, the number of clauses is reduced from $\mathcal{O}((e_2 - e_1) \cdot m)$ to $e_2 - e_1$ or $k - m$ depending on the concrete values of e_1 and e_2.

Direction for the selection of the bounding condition. With the novel encoding method, multiple bounding conditions can be integrated into the standard SAT method, conveniently. However, whether the searching phase regarding the modified SAT problem can be accelerated is the actual problem. We take the distinguisher searching problem of `GIFT-64` as an illustration and compare the runtime for solving SAT problems involving different sets of bound conditions. With the observations in the tests, we experimentally show the accelerating effect of the encoding method. Further, a strategy on how to select the sets of bounding conditions that potentially achieve extraordinary advances is proposed. We hope it may be helpful for both designers and attackers in search of differential and linear characteristics.

Complete bounds about differential and linear characteristics of multiple ciphers. The new idea is exploited to search for various trails of multiple primitives. For `PRESENT`, `GIFT-64`, `RECTANGLE` [ZBL+15], `LBlock` [WZ11], `TWINE` [SMMK12], and some versions in `SIMON` and `SPECK` [BSS+13] families of block ciphers, we obtain the complete bounds (full rounds) on the number of active S-boxes, the differential probability, as well as the linear bias. To our knowledge, we are the first one to offer complete information about the optimal differential and linear characteristics. For `GIFT-128`, we obtain the full picture regarding the number of differential and linear active S-boxes. Beyond that, the optimal differential trails with the maximum probability of `GIFT-128` for up to 29 rounds and the optimal linear characteristics with the maximum correlation for up to 25 rounds are discovered. Although Li et al. [LWZZ19] also found a 20-round differential trail with probability $2^{-121.415}$, their searching method did not ensure the optimality. All the searches in this paper guarantee the optimality. A comparison of the maximum length of differential and linear trails with different approaches for `SPECK` is provided in Table 1. For all versions in the `SPECK` family of block ciphers, our results reach the maximum length of differential and linear trails among all methods targeting the optimal trail.

Related-key differential attack on 26-round `GIFT-64`. The acceleration method also can be employed to speed up the search of related-key differential characteristics. In this way, for `GIFT-64`, we get an 18-round related-key differential distinguisher with probability 2^{-58}. This distinguisher is utilised to launch a 26-round key-recovery attack. The data complexity is $2^{60.96}$ chosen plaintexts, the time complexity is $2^{123.23}$ 26-round of encryptions, and

Table 1: The maximum length of trails with different approaches for SPECK.

Trail	Ref.	Optimal	SPECK32	SPECK48	SPECK64	SPECK96	SPECK128
Differential	[BVC16]	✓	10	9	8	7	6
	[FWG+16]	-	9	11	15	16	19
	[LLJW19]	✓	10	12	16	8	8
	Sect. 5.3	✓	**22**	**18**	**27**	**10**	**9**
Linear	[BVC16]	✓	6	-	-	-	-
	[FWG+16]	-	9	10	13	15	16
	[LWR16]	✓	22	11	13	9	9
	[LLJW19]	✓	22	13	15	9	9
	Sect. 5.3	✓	**22**	**23**	**27**	**14**	**10**

the memory complexity is about $2^{102.86}$. As far as we know, this is the longest attack on GIFT-64. A summary of cryptanalytic results on GIFT-64 to date is provided in Table 2. We note that our result is far from threatening the security of GIFT-64 since the authors recommended users to double the number of rounds under the related-key attack setting.

Outline. The relevant contents on the automatic search with the SAT method are introduced in Sect. 2. In Sect. 3, we propose a method to encode Matsui's bounding conditions into Boolean formulas with a minor increment on the number of clauses. To figure out the accelerating effect of the bounding condition, we investigate the performances regarding different sets of bounding conditions in Sect. 4. Also, a strategy for the selection of the bounding condition is presented. The novel searching method is applied to several block ciphers and derives many new findings in Sect. 5. We conclude the paper in Sect. 6. The source codes are publicly available at https://github.com/SunLing134340/Accelerating_Automatic_Search.

2 Automatic Searches with the SAT Method

2.1 Preliminaries about SAT and SMT Problems

A formula is named as a *Boolean formula* if it is formulated with Boolean variables, operators AND ($\wedge$), OR ($\vee$), NOT ($\bar{\cdot}$), and parentheses. Every Boolean formula can be converted into an equivalent formula that is in *conjunctive normal form* (CNF) [RN10, Sob10], which

Table 2: Symmary of cryptanalytic results on GIFT-64.

Round	Method	Setting	Time	Data	Memory	Ref.
20	Differential	SK	$2^{112.68}$	$2^{62.00}$	$2^{112.00}$	[CZD19]
21	Differential	SK	$2^{107.61}$	$2^{64.00}$	$2^{96.00}$	[CZD19]
23	Boomerang	RK	$2^{126.60}$	$2^{63.30}$	-	[LS19]
24	Rectangle	RK	$2^{106.00}$	$2^{63.78}$	$2^{64.10}$	[JZZD20]
25	Rectangle	RK	$2^{120.92}$	$2^{63.78}$	$2^{64.10}$	[JZZD20]
26	**Differential**	**RK**	$\mathbf{2^{123.23}}$	$\mathbf{2^{60.96}}$	$\mathbf{2^{102.86}}$	**Sect. 5.4**

is a propositional formula of the form $\bigwedge_{i=0}^{n} \bigvee_{j=0}^{m_i} C_{ij}$, where each C_{ij} $(0 \leqslant i \leqslant n,\ 0 \leqslant j \leqslant m_i)$ is either an atomic formula, i.e., a variable or constant, or the negation of an atomic formula, and each disjunction $\bigvee_{j=0}^{m_i} C_{ij}$ is called a *clause*.

The *Boolean satisfiability problem* (SAT) is the problem of determining whether there exists an evaluation for the binary variables such that the value of the given Boolean formula equals one. Although the SAT problem is the first problem that was proven to be NP-complete [Coo71], modern SAT solvers can solve problem instances comprising tens of thousands of variables and millions of clauses.

An extension of the SAT problem is *satisfiability modulo theories* (SMT) problem, in which some of the Boolean variables are replaced by predicates over a suitable set of binary and (or) non-binary variables. The *predicates* are binary-valued functions, such as linear inequalities, arrays, and all-different constraints. This kind of extension typically remains NP-complete, and a great deal of SMT solvers available to date follow the *eager approach*, which interprets SMT instances into SAT instances first and then transfers the CNF formulas to a SAT solver.

2.2 Related Works about the Search with the SAT/SMT Method

We investigate all literature involving the search of differential and (or) linear characteristics in cryptanalysis with the SAT/SMT method and find that most of these works [MP13, AJN14, Ste, KLT15, SHY16, AK18, LLL+19, RLA20, ARS+20] rely on the SMT method and utilise the SMT solver STP [GD07]. The remaining two works [LWR16, SWW18] that claim to be SAT-based methods tie to the generalised SAT problem with XOR clauses in the CNF formula since the employed SAT solver, which is called Cryptominisat [SNC09], is specially designed to be compatible with XOR operations. Thus, none of the existing automatic tools is realised with the real SAT problem that only admits AND, OR, and NOT operations.

In this work, we aim at accelerating the search for optimal differential and linear characteristics with the real SAT method. The SAT solver we use is CaDiCaL [Bie19], which is based on the *conflict-driven clause learning* (CDCL) algorithm [SS96, JS97]. The internal functioning of the CDCL algorithm is inspired by the *Davis-Putnam-Logemann-Loveland* (DPLL) algorithm [DP60, DLL62], which is the kernel of some frequently-used SAT solvers, including Cryptominisat as mentioned above. Nevertheless, the CDCL algorithm can learn new clauses with conflict analysis. Another notable distinction between the CDCL and DPLL algorithms is that the back jumping in the CDCL algorithm is non-chronological. Both the clause learning and the modified backtracking phases do not alter the soundness and completeness of the algorithm. We observe that CaDiCaL is faster than Cryptominisat regarding differential and linear trails searching problems, and this is the main reason that we choose this SAT solver.

To discover useful distinguishers with the off-the-shelf SAT solver, we should specify the distinguisher searching problem with CNF formulas. The clauses in a CNF formula regarding the search of the optimal differential or linear trail are classified into two groups. The first group represents the propagations of differences or linear masks inside the cipher, and the second one measures the non-random feature of the trail, which can be set as the number of active S-boxes, the differential probability, or the linear bias, optionally. In the remaining of this section, we first recall SAT models demonstrating the differential and linear propagations of some necessary operations, which act as components of the primitives analysed in this paper. Then, the second group of clauses constructed with the sequential encoding method is introduced.

2.3 SAT Models of Some Necessary Operations

We start with the non-probabilistic models of two linear operations, which are branching and XOR operations. The differential and linear propagations of these operations are deterministic. After that, probabilistic models of some non-linear operations are presented.

2.3.1 Non-probabilistic Models

In the following, α_i $(0 \leqslant i \leqslant n-1)$ denotes the i-th bit of the n-bit vector α. We always use α_0 to stand for the most-significant bit.

Differential Model 1 (Branching)**.** *For the n-bit branching operation shown in Figure 1 (a), denote α the input difference, β and γ the two output differences. The differential holds if and only if the values of α, β, and γ validate all the assertions in the following.*

$$\left.\begin{aligned} \alpha_i \vee \overline{\beta_i} &= 1 \\ \overline{\alpha_i} \vee \beta_i &= 1 \\ \alpha_i \vee \overline{\gamma_i} &= 1 \\ \overline{\alpha_i} \vee \gamma_i &= 1 \end{aligned}\right\} 0 \leqslant i \leqslant n-1$$

Differential Model 2 (XOR)**.** *For the n-bit XOR operation illustrated in Figure 1 (b), we use α and β to represent the two input differences and denote the output difference as γ. The differential holds if and only if the values of α, β, and γ validate all the assertions in the following.*

$$\left.\begin{aligned} \alpha_i \vee \beta_i \vee \overline{\gamma_i} &= 1 \\ \alpha_i \vee \overline{\beta_i} \vee \gamma_i &= 1 \\ \overline{\alpha_i} \vee \beta_i \vee \gamma_i &= 1 \\ \overline{\alpha_i} \vee \overline{\beta_i} \vee \overline{\gamma_i} &= 1 \end{aligned}\right\} 0 \leqslant i \leqslant n-1$$

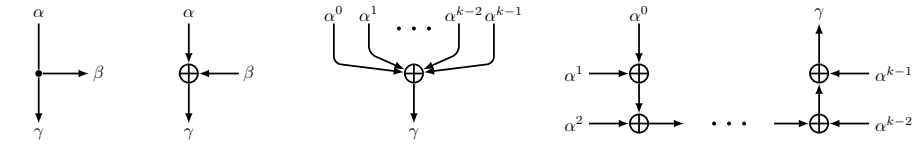
Generally, for the n-bit XOR operation with k inputs as in Figure 1 (c), we denote the k input differences as α^0, α^1, …, α^{k-1} and the output difference as γ. There is a trade-off between the number of variables and the number of clauses when we construct the differential model of this operation. On the one hand, we can decompose the k-input XOR operation into $(k-1)$ 2-input XOR operations as in Figure 1 (d) and introduce $(k-2)\cdot n$ auxiliary Boolean variables to keep track of the differences of the $k-2$ intermediate states. After sequentially applying **Differential Model 1** to the $(k-1)$ 2-input XOR operations, the differential propagation of the k-input XOR operation can be expressed with $4\cdot(k-1)\cdot n$ clauses. On the other hand, the propagation can be established with $n\cdot 2^k$ clauses without using any auxiliary variables. To be explicit, for each of the 2^k $(k+1)$-tuple $(a_0, a_1, \ldots, a_k)$ of Boolean variables with $a_0 \oplus a_1 \oplus \cdots \oplus a_k = 1$, we generate n equations as follows

$$(\alpha_i^0 \oplus a_0) \vee (\alpha_i^1 \oplus a_1) \vee \cdots \vee (\alpha_i^{k-1} \oplus a_{k-1}) \vee (\gamma_i \oplus a_k) = 1,\ 0 \leqslant i \leqslant n-1.$$

Note that these equations are clauses in CNF formulas since $\alpha_i^j \oplus a_j$ equals α_i^j if a_j is zero and equals $\overline{\alpha_i^j}$ otherwise. At the same time, the valid differential propagation $(\alpha^0, \alpha^1, \ldots, \alpha^{k-1}) \to (\gamma)$ fulfils these clauses, simultaneously. As the values of k for the XOR operations with more than two inputs in the subsequent probabilistic models are relatively small, we always pick the second option, which maintains the minimum number of variables.

Besides, to create the differential model of the matrix multiplication operation, which is exploited in multiple ciphers to provide the diffusion property, we note that this operation can be written as a sequence of branching and XOR operations [SLR+15]. Hence, the model can be generated with **Differential Model 1** and **2**.

Since the propagations of differences and linear masks concerning the branching and XOR operations are dual [SLR+15], the linear model of the branching (resp. XOR) operation is the same as the differential model of the XOR (resp. branching) operation. Thus, we do not restate the non-probabilistic linear models.



(a) Branching. (b) XOR. (c) k-input XOR. (d) Sequential k-input XOR.

Figure 1: Linear operations.

2.3.2 Probabilistic Models

The propagations of differences and linear masks for non-linear operations are probabilistic. Here, we consider three non-linear operations, which are S-box, modular addition operation, and `SIMON`-like round function.

S-box. We implement the method in [SWW18] to create differential and linear models of S-boxes. For primitives with S-boxes as building blocks, the automatic searches of distinguishers in the field of differential and linear cryptanalyses accomplish two goals. One is finding optimal trails with the minimum number of active S-boxes, and the other one is discovering optimal trails with the maximum differential probability or linear correlation. We take the construction of the differential model concerning the number of active S-boxes as an instance. Likewise, we can generate remaining differential and linear models of S-boxes regarding different searching purposes.

Denote $(\alpha_0, \alpha_1, \ldots, \alpha_{s-1})$ and $(\beta_0, \beta_1, \ldots, \beta_{s-1})$ the input and output differences of the s-bit S-box as in Figure 2 (a). An extra binary variable w is required to characterise whether the S-box is active or not. With the differential distribution table (DDT), if $(a_0, a_1, \ldots, a_{s-1}) \to (b_0, b_1, \ldots, b_{s-1})$ is a possible differential propagation with a nonzero probability, w is set as one. w equals zero if the differential propagation $(a_0, a_1, \ldots, a_{s-1}) \to (b_0, b_1, \ldots, b_{s-1})$ is deterministic. Then, we enumerate all η $(2 \cdot s + 1)$-bit negative combinations $(a_0^{(i)}, \ldots, a_{s-1}^{(i)}, b_0^{(i)}, \ldots, b_{s-1}^{(i)}, w^{(i)})$ $(0 \leqslant i \leqslant \eta - 1)$ such that neither of the two assignment rules is satisfied. The following η clauses constitute a primary differential model of the given S-box,

$$\bigvee_{j=0}^{s-1} \left(\alpha_j \oplus a_j^{(i)}\right) \vee \bigvee_{j=0}^{s-1} \left(\beta_j \oplus b_j^{(i)}\right) \vee \left(w \oplus w^{(i)}\right) = 1, \ 0 \leqslant i \leqslant \eta - 1.$$

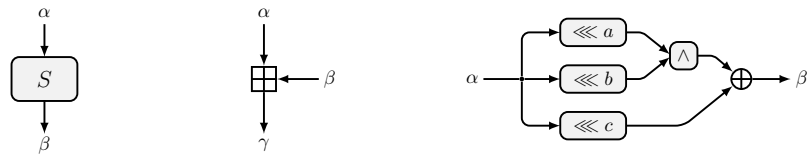
To generate a model with fewer clauses, we first define a function f over the $(2 \cdot s + 1)$-bit vector $\boldsymbol{x} = (x_0, x_1, \ldots, x_{2 \cdot s})$ as

$$f(\boldsymbol{x}) = \begin{cases} 0, & \text{if } \boldsymbol{x} \text{ is a negative combination} \\ 1, & \text{otherwise} \end{cases}.$$

Equivalently, f can be reformulated as the *product-of-sum representation*

$$f(\boldsymbol{x}) = \bigwedge_{\boldsymbol{c}\in\mathbb{F}_2^{2\cdot s+1}} \left(f(\boldsymbol{c}) \vee \bigvee_{i=0}^{2\cdot s} (x_i \oplus c_i) \right),$$

where $\boldsymbol{c} = (c_0, c_1, \ldots, c_{2\cdot s})$. After simplifying this representation with some openly available programs such as Logic Friday[1] and Espresso[2], a smaller set of clauses is yielded, which is the differential model we adopt in the implementation.



(a) S-box. (b) Modular addition operation. (c) SIMON-like round function.

Figure 2: Non-linear operations.

Modular addition operation. The modular addition operation is a crucial ingredient for ARX ciphers. The differential and linear models of the modular addition operation with CNF formulas are accommodated from the models in [MP13] and [LWR16], respectively. Note that the XOR operations signified by '$\oplus$' in the following models are symbolic representations, which ensure compact descriptions of the models. In the implementation, these XOR operations are converted into CNF formulas with the method in Sect. 2.3.1.

Differential Model 3 (Modular Addition, [MP13])**.** *For the n-bit modular addition operation as in Figure 2 (b), we use α and β to stand for the two input differences and denote the output difference as γ. The differential is valid if and only if the values of α, β, and γ validate all the assertions listed below.*

$$\left.\begin{aligned}
&\alpha_i \vee \beta_i \vee \overline{\gamma_i} \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\
&\alpha_i \vee \overline{\beta_i} \vee \gamma_i \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\
&\overline{\alpha_i} \vee \beta_i \vee \gamma_i \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\
&\overline{\alpha_i} \vee \overline{\beta_i} \vee \overline{\gamma_i} \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\
&\alpha_i \vee \beta_i \vee \gamma_i \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\
&\alpha_i \vee \overline{\beta_i} \vee \overline{\gamma_i} \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\
&\overline{\alpha_i} \vee \beta_i \vee \overline{\gamma_i} \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\
&\overline{\alpha_i} \vee \overline{\beta_i} \vee \gamma_i \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1
\end{aligned}\right\} 0 \leqslant i \leqslant n-2$$

$$\alpha_{n-1} \oplus \beta_{n-1} \oplus \gamma_{n-1} = 0$$

The weight, which is the negative value of the binary logarithm of the differential probability, of the valid differential is $\sum_{i=0}^{n-2} w_i$, where w_i's are binary values satisfying the following equations.

[1] https://web.archive.org/web/20131022021257/http://www.sontrak.com/
[2] https://code.google.com/archive/p/eqntott/

$$\left.\begin{array}{l}\overline{\alpha_{i+1}} \vee \gamma_{i+1} \vee w_i = 1\\ \beta_{i+1} \vee \overline{\gamma_{i+1}} \vee w_i = 1\\ \alpha_{i+1} \vee \overline{\beta_{i+1}} \vee w_i = 1\\ \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} \vee \overline{w_i} = 1\\ \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} \vee \overline{w_i} = 1\end{array}\right\} 0 \leqslant i \leqslant n-2$$

Linear Model 1 (Modular Addition, [LWR16])**.** *For the n-bit modular addition operation as in Figure 2 (b), we use α and β to represent the two input linear masks and denote the output mask as γ. Additionally, we introduce an n-bit vector z to assist us in evaluating the correlation. The correlation of the linear approximation is nonzero if the values of α, β, γ, and z fulfil all the constraints in the following.*

$$\begin{array}{l}\overline{z_0} = 1\\ \alpha_0 \oplus \beta_0 \oplus \gamma_0 \oplus z_1 = 0\\ \alpha_{j+1} \oplus \beta_{j+1} \oplus \gamma_{j+1} \oplus z_{j+1} \oplus z_{j+2} = 0,\ 0 \leqslant j \leqslant n-3\end{array}$$
$$\left.\begin{array}{l}\alpha_i \vee \overline{\gamma_i} \vee z_i = 1\\ \overline{\alpha_i} \vee \gamma_i \vee z_i = 1\\ \beta_i \vee \overline{\gamma_i} \vee z_i = 1\\ \overline{\beta_i} \vee \gamma_i \vee z_i = 1\end{array}\right\} 0 \leqslant i \leqslant n-1$$

Similarly, the binary logarithm of the absolute value of the correlation reflects the performance of the linear approximation in the attack. The opposite number of this feature is calculated as $\sum_{i=0}^{n-1} z_i$.

`SIMON`-like round function. As in Figure 2 (c), the n-bit `SIMON`-like round function is defined as $f(x) = (x \lll a) \wedge (x \lll b) \oplus (x \lll c)$, where $a > b$, n is even and $\gcd(n, a-b) = 1$. This function serves as the round function of the `SIMON` block cipher family [BSS+13]. The differential model originates from [KLT15].

Differential Model 4 (`SIMON`-like Round Function, [KLT15])**.** *For the n-bit `SIMON`-like round function, we denote α and β the input and output differences, respectively. Additionally, three n-bit variables* `varibits`, `doublebits`, *and z are incorporated so that we can evaluate the differential probability. If α is not an all-ones vector, the differential is valid if and only if the values of α, β,* `varibits`, `doublebits`, *and z validate all the constraints listed below.*

$$\left.\begin{array}{l}\overline{\alpha_{(i+a) \bmod n}} \vee \texttt{varibits}_i = 1\\ \overline{\alpha_{(i+b) \bmod n}} \vee \texttt{varibits}_i = 1\\ \alpha_{(i+a) \bmod n} \vee \alpha_{(i+b) \bmod n} \vee \overline{\texttt{varibits}_i} = 1\\ \alpha_{(i+b) \bmod n} \vee \overline{\texttt{doublebits}_i} = 1\\ \overline{\alpha_{(i+b) \bmod n}} \vee \alpha_{(i+2\cdot a-b) \bmod n} \vee \overline{\texttt{doublebits}_i} = 1\\ \overline{\alpha_{(i+a) \bmod n}} \vee \overline{\alpha_{(i+b) \bmod n}} \vee \overline{\texttt{doublebits}_i} = 1\\ \alpha_{(i+a) \bmod n} \vee \overline{\alpha_{(i+b) \bmod n}} \vee \overline{\alpha_{(i+2\cdot a-b) \bmod n}} \vee \texttt{doublebits}_i = 1\\ \alpha_{(i+c) \bmod n} \oplus \beta_i \oplus z_i = 0\\ \texttt{varibits}_i \vee \overline{z_i} = 1\\ \overline{\texttt{doublebits}_i} \vee z_i \vee \overline{z_{(i+a-b) \bmod n}} = 1\\ \overline{\texttt{doublebits}_i} \vee \overline{z_i} \vee z_{(i+a-b) \bmod n} = 1\end{array}\right\} 0 \leqslant i \leqslant n-1$$

The weight of the possible differential is $\sum_{i=0}^{n-1} (\texttt{varibits}_i \oplus \texttt{doublebits}_i)$.

The linear model (cf. **Theorem 5**) in [KLT15] is an elegant model that perfectly handles the dependency and thus results in a precise evaluation for the linear property of SIMON. However, for the difficulty of encoding this model with Boolean equations, we do not apply it. Instead, we regard the AND operations in the round function as independent S-boxes and claim the linear approximations of SIMON found in this paper are heuristic. Specifically, for the linear model, we consider the AND operation with two input bits as $(x_{(i+a) \bmod n}, x_{(i+b) \bmod n})$ and view it as an S-box. After computing its linear approximation table (LAT), we exploit the model generating method for S-boxes to complete the formation of the linear model.

Linear Model 2 (SIMON-like Round Function)**.** *For the n-bit SIMON-like round function, we denote the input and output linear masks as α and β, respectively. Two auxiliary n-bit variables γ^0 and γ^1 are employed to record the two input masks of the AND operation. To estimate the linear correlation, we also import an n-bit variable z. The correlation of the linear approximation is nonzero if the values of α, β, γ^0, γ^1, and z validate all the constraints listed in the following.*

$$\left.\begin{array}{l} \beta_i \vee \overline{z_i} = 1 \\ \overline{\beta_i} \vee z_i = 1 \\ \overline{\gamma_i^0} \vee z_i = 1 \\ \overline{\gamma_i^1} \vee z_i = 1 \\ \alpha_i \oplus \beta_{(i-c) \bmod n} \oplus \gamma^0_{(i-a) \bmod n} \oplus \gamma^1_{(i-b) \bmod n} = 0 \end{array}\right\} 0 \leqslant i \leqslant n-1$$

The value of $\sum_{i=0}^{n-1} z_i$ *equals the opposite number of the binary logarithm of the absolute value of the correlation.*

In the application, to characterise the differential or linear propagation inside the cipher, we first decompose the round function into a sequence of basic operations and generate SAT models for these operations. Then, the basic models are interlinked with each other by using some common variables expressing the differences or linear masks of the internal states.

2.4 Sequential Encoding Method

Since we always aim at trails with significant non-trivial features, according to the specific goal, we should restrict the number of active S-boxes, the differential probability, or the linear correlation in the distinguisher searching problem. All of these kinds of constraints can be abstracted as the Boolean cardinality constraint $\sum_{j=0}^{n-1} x_j \leqslant k$, where x_j's are Boolean variables, and k is a non-negative integer. Following the approaches in [LWR16, SWW18], we take the *sequential encoding method* [Sin05] to convert this constraint into CNF formulas.

The sequential encoding method is based on the sequential counter circuit as shown in Figure 3. The circuit computes the partial sum $s_i = \sum_{j=0}^{i} x_j$ for increasing the value of i from 0 to $n-2$. To express this circuit with CNF formulas, we first introduce $(n-1) \cdot k$ auxiliary variables $s_{i,j}$ $(0 \leqslant i \leqslant n-2, 0 \leqslant j \leqslant k-1)$. The partial sum s_i is represented as $s_{i,k-1} \| s_{i,k-2} \| \cdots \| s_{i,1} \| s_{i,0}$ under the *unary numeral system*. That is, $s_i = m$ $(0 \leqslant m \leqslant k)$

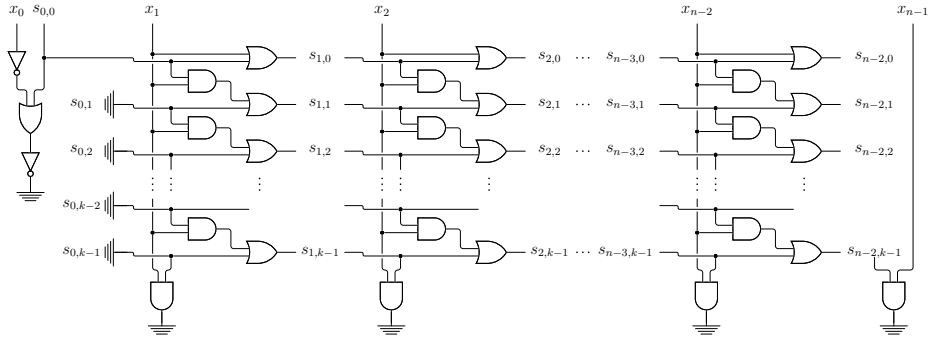


Figure 3: Sequential counter circuit.

is equivalent to the following equations,

$$s_{i,k-1} = \cdots = s_{i,m+1} = s_{i,m} = 0,\ s_{i,m-1} = \cdots = s_{i,1} = s_{i,0} = 1.$$

Thus, the equality $\sum_{j=0}^{i} x_j = s_i = \sum_{j=0}^{k-1} s_{i,j}$ holds. Accompanied by the notification that the sequence of the partial sum $\left\{ s_i = \sum_{j=0}^{k-1} s_{i,j} \;\middle|\; 0 \leqslant i \leqslant n-2 \right\}$ is non-decreasing, the constraint $\sum_{j=0}^{n-1} x_j \leqslant k$ is satisfied if the following implication predicates hold, simultaneously.

$$\begin{array}{l}
\texttt{if } x_0 = 1 \texttt{ then } s_{0,0} = 1 \texttt{ endif} \\
s_{0,j} = 0,\ 1 \leqslant j \leqslant k-1 \\
\left.\begin{array}{l}
\texttt{if } x_i = 1 \texttt{ then } s_{i,0} = 1 \texttt{ endif} \\
\texttt{if } s_{i-1,0} = 1 \texttt{ then } s_{i,0} = 1 \texttt{ endif} \\
\left.\begin{array}{l}
\texttt{if } x_i = 1 \texttt{ and } s_{i-1,j-1} = 1 \texttt{ then } s_{i,j} = 1 \texttt{ endif} \\
\texttt{if } s_{i-1,j} = 1 \texttt{ then } s_{i,j} = 1 \texttt{ endif}
\end{array}\right\} 1 \leqslant j \leqslant k-1 \\
\texttt{if } x_i = 1 \texttt{ then } s_{i-1,k-1} = 0 \texttt{ endif}
\end{array}\right\} 1 \leqslant i \leqslant n-2 \\
\texttt{if } x_{n-1} = 1 \texttt{ then } s_{n-2,k-1} = 0 \texttt{ endif}
\end{array}$$

These predicates are interpreted as the following $2 \cdot k \cdot n + n - 3 \cdot k - 1$ clauses, which formulate the SAT model of the Boolean cardinality constraint $\sum_{j=0}^{n-1} x_j \leqslant k$.

$$\begin{array}{l}
\overline{x_0} \vee s_{0,0} = 1 \\
\overline{s_{0,j}} = 1,\ 1 \leqslant j \leqslant k-1 \\
\left.\begin{array}{l}
\overline{x_i} \vee s_{i,0} = 1 \\
\overline{s_{i-1,0}} \vee s_{i,0} = 1 \\
\left.\begin{array}{l}
\overline{x_i} \vee \overline{s_{i-1,j-1}} \vee s_{i,j} = 1 \\
\overline{s_{i-1,j}} \vee s_{i,j} = 1
\end{array}\right\} 1 \leqslant j \leqslant k-1 \\
\overline{x_i} \vee \overline{s_{i-1,k-1}} = 1
\end{array}\right\} 1 \leqslant i \leqslant n-2 \\
\overline{x_{n-1}} \vee \overline{s_{n-2,k-1}} = 1
\end{array} \tag{1}$$

3 Integrating Bounding Conditions into the SAT Method

At EUROCRYPT 1994, Matsui [Mat94] proposed a branch-and-bound depth-first searching algorithm that can be used to identify the optimal differential trails with the maximum probability of a symmetric-key primitive. The efficiency of Matsui's algorithm comes from the manipulation of the known upper bounds on probabilities of short trails. Denote $\mathrm{Pr}_{\mathtt{Opt}}(i)$ the maximum probability achieved by i-round differential trails for $1 \leqslant i \leqslant R-1$. With these messages, we aim at searching for R-round optimal trails. Let $\mathrm{Pr}_{\mathtt{Ini}}(R)$ be the initial estimation for the probability bound achieved by R-round trails. Now, suppose that we obtain a partial trail $(\alpha^0, \alpha^1, \ldots, \alpha^r)$ covering the first r rounds ($1 \leqslant r < R$), which is a child node located at the r-th level of the search tree created with Matsui's algorithm. The subtree originating from this node will not be explored if the following *bounding condition* is violated

$$\prod_{i=0}^{r-1} \mathrm{Pr}\left(\alpha^i \to \alpha^{i+1}\right) \cdot \mathrm{Pr}_{\mathtt{Opt}}\left(R-r\right) \geqslant \mathrm{Pr}_{\mathtt{Ini}}(R), \tag{2}$$

where $\mathrm{Pr}(\alpha^i \to \alpha^{i+1})$ is the probability of the differential propagation in the i-th round.

Note that the bounding condition has been incorporated in the automatic search of differential characteristics with the MILP method by Zhang et al. [ZSCH18]. In this section, we show how to integrate the bounding condition into the SAT method without introducing new auxiliary variables so that the search for optimal differential and linear trails can be accelerated. Although the description in this section proceeds with the optimal differential trail possessing the maximum probability, we remind the readers that the method can be applied to the search of optimal linear characteristics as well.

3.1 Extracting the Essential of the Problem

Typically, to check the existence of R-round differential trail $(\alpha^0, \alpha^1, \ldots, \alpha^R)$ with the probability being $\mathrm{Pr}_{\mathtt{Ini}}(R)$, the SAT method tries to find instantiations for α^i's such that

$$\sum_{i=0}^{R-1} \left(-\log_2 \left(\mathrm{Pr}\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) \leqslant -\log_2\left(\mathrm{Pr}_{\mathtt{Ini}}(R)\right). \tag{3}$$

We import Boolean variables $w_j^{(i)}$ $(0 \leqslant j \leqslant \varpi)$ to calculate the weight of the differential propagation $\alpha^i \to \alpha^{i+1}$ in the i-th round, i.e., $-\log_2\left(\mathrm{Pr}(\alpha^i \to \alpha^{i+1})\right) = \sum_{j=0}^{\varpi-1} w_j^{(i)}$. For simplicity, we define the symbols $n \triangleq R \cdot \varpi$, $k \triangleq -\log_2\left(\mathrm{Pr}_{\mathtt{Ini}}(R)\right)$, and $x_{\varpi \cdot i + j} \triangleq w_j^{(i)}$. Then, Eq. (3) is rewritten as follows

$$\sum_{i=0}^{n-1} x_i = \sum_{i=0}^{R-1} \sum_{j=0}^{\varpi-1} w_j^{(i)} \leqslant k. \tag{4}$$

On the other hand, the bounding condition in Eq. (2) is equivalent to

$$\sum_{i=0}^{r-1} \left(-\log_2 \left(\mathrm{Pr}\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) \leqslant \log_2\left(\mathrm{Pr}_{\mathtt{Opt}}(R-r)\right) - \log_2\left(\mathrm{Pr}_{\mathtt{Ini}}(R)\right). \tag{5}$$

Note that the right-hand side of Eq. (5) is a constant no more than k, and the left-hand side of Eq. (5) matches the weight of the trail covering the first r rounds. Generally, with

the previously defined symbols, all bounding conditions can be replaced with an inequality constraint of the following form

$$\sum_{i=e_1}^{e_2} x_i \leqslant m, \tag{6}$$

where $e_1 \geqslant 0$, $e_2 \leqslant n-1$, and $m \leqslant k$.

Of course, we can reapply the sequential encoding method to Eq. (6) by introducing a new group of $(e_2 - e_1 + 1) \cdot m$ auxiliary variables and generating $(2 \cdot m + 1) \cdot (e_2 - e_1) - m$ clauses. However, when multiple bounding conditions are considered, this direct approach will significantly expand the number of variables and clauses in the SAT problem. Since increasing the number of variables and clauses in the SAT problem may result in a negative impact on the efficiency of the SAT solver, which is a conjecture resulting from the experience and observations in the numerous tests, we attempt to find another way to encode the bounding condition. The new approach is motivated by the circuit of the sequential encoding method and reuses variables in the sequential counter circuit of the original objective function in Eq. (4). Without claiming any new variables, the number of clauses is reduced from $\mathcal{O}((e_2 - e_1) \cdot m)$ to $e_2 - e_1$ or $k - m$ depending on the concrete values of e_1 and e_2.

3.2 Clausal Encoding of the Bounding Condition

Formally, we target a *clausal encoding*, whose definition is supplied in the following, of the two Boolean cardinality constraints $\sum_{i=0}^{n-1} x_i \leqslant k$ and $\sum_{j=e_1}^{e_2} x_j \leqslant m$. Note that the following definition is adjusted from the one in [Sin05].

Definition 1 (Clausal Encoding of Two Boolean Cardinality Constraints)**.** Denote $\mathbb{X} = \{x_0, x_1, \ldots, x_{n-1}\}$ the set of variables in the constraints and $\{\varsigma_0, \varsigma_1, \ldots, \varsigma_{\ell-1}\}$ the set of additional encoding variables. A set $\mathbb{C}$ of clauses over the set of variables $\mathbb{V} = \{x_0, \ldots, x_{n-1}, \varsigma_0 \ldots \varsigma_{\ell-1}\}$ is a clausal encoding of the two Boolean cardinality constraints $\sum_{i=0}^{n-1} x_i \leqslant k$ and $\sum_{j=e_1}^{e_2} x_j \leqslant m$ if for all assignments $\Psi_{\mathbb{X}} \in \mathbb{F}_2^n$ of the variables in $\mathbb{X}$ that validate the two constraints the following holds: $\Psi_{\mathbb{X}}$ validates the two constraints if and only if there is an extended assignment $\Psi_{\mathbb{V}}$ of all variables in $\mathbb{V}$ such that the restricted value of $\Psi_{\mathbb{V}}$ on $\mathbb{X}$ coincides with the value of $\Psi_{\mathbb{X}}$.

For the first constraint, we apply the normal sequential encoding method in Sect. 2.4 with additional encoding variables $s_{i,j}$ $(0 \leqslant i \leqslant n-2,\ 0 \leqslant j \leqslant k-1)$. The corresponding sequential counter circuit accomplishes the computation of the partial sum $\sum_{j=0}^{i} x_j$ with $\sum_{j=0}^{k-1} s_{i,j}$ for $0 \leqslant i \leqslant n-2$. Note that the second constraint focuses on the value of a consecutive partial sum of variables belonging to the first constraint, and it can be inferred from the values of $\sum_{j=0}^{e_1-1} x_j$ and $\sum_{j=0}^{e_2} x_j$, which have been evaluated in the encoding of the first constraint. This observation reminds us to explore the possibility of reusing variables in the sequential counter circuit to realise the encoding of the second constraint. According to the values of e_1 and e_2, we split the encoding problem regarding the second constraint into three different cases and construct SAT models, separately.

Case 1. $\sum_{j=e_1}^{e_2} x_j \leqslant m$ with $e_1 = 0$ and $e_2 < n-1$

The bounding condition is rewritten as $\sum_{i=0}^{e_2} x_i \leqslant m$. Apart from the equivalence relation between the values of $\sum_{j=0}^{i} x_j$ and $\sum_{j=0}^{k-1} s_{i,j}$, we also find that $s_{i,j} = 0$ automatically implies $s_{i,j'} = 0$ for all $j < j' \leqslant k-1$ by the intrinsic property of the unary numeral system. With these properties, we derive the following $e_2 - e_1$ predicates, which guarantee the satisfiability of the second constraint.

$$\texttt{if } x_i = 1 \texttt{ then } s_{i-1,m-1} = 0 \texttt{ endif for } 1 \leqslant i \leqslant e_2$$

These predicates are converted into the following Boolean expressions.

$$\overline{x_i} \vee \overline{s_{i-1,m-1}} = 1,\ 1 \leqslant i \leqslant e_2 \tag{7}$$

Thus, the combination of clauses in Eq. (1) and (7) constitutes a clausal encoding of the two Boolean cardinality constraints in this case.

Case 2. $\sum_{j=e_1}^{e_2} x_j \leqslant m$ with $e_1 > 0$ and $e_2 < n-1$

With a similar consideration as in *Case* 1, we create the following $k-m$ predicates so that the least upper bound for the value of $\sum_{j=e_1}^{e_2} x_j$ is fixed as m.

$$\texttt{if } s_{e_1-1,j} = 0 \texttt{ then } s_{e_2,j+m} = 0 \texttt{ endif for } 0 \leqslant j \leqslant k-m-1$$

These predicates are substituted with clauses listed below.

$$s_{e_1-1,j} \vee \overline{s_{e_2,j+m}} = 1,\ 0 \leqslant j \leqslant k-m-1 \tag{8}$$

The combination of clauses in Eq. (1) and (8) can operate as a clausal encoding of the two Boolean cardinality constraints with $e_1 > 0$ and $e_2 < n-1$.

Case 3. $\sum_{j=e_1}^{e_2} x_j \leqslant m$ with $e_1 > 0$ and $e_2 = n-1$

The constraint is adjusted as $\sum_{j=e_1}^{n-1} x_j \leqslant m$. By incorporating the property of the sequential counter circuit, we come up with the following predicates, which enable us to restrict the value of the summation $\sum_{j=e_1}^{n-1} x_j$ in this case.

$$\texttt{if } s_{e_1-1,j} = 0 \texttt{ then } s_{n-2,j+m} = 0 \texttt{ endif for } 0 \leqslant j \leqslant k-m-1$$
$$\texttt{if } s_{e_1-1,j} = 0 \texttt{ and } x_{n-1} = 1 \texttt{ then } s_{n-2,j+m-1} = 0 \texttt{ endif for } 0 \leqslant j \leqslant k-m$$

The corresponding clauses regarding these implication predicates are listed as follows.

$$\begin{aligned} &s_{e_1-1,j} \vee \overline{s_{n-2,j+m}} = 1,\ 0 \leqslant j \leqslant k-m-1 \\ &s_{e_1-1,j} \vee \overline{x_{n-1}} \vee \overline{s_{n-2,j+m-1}},\ 0 \leqslant j \leqslant k-m \end{aligned} \tag{9}$$

The clauses in Eq. (1) and (9) make up a clausal encoding of the two Boolean cardinality constraints.

Now, we finish the construction of SAT model for the bounding condition. This new process allows us to intermix multiple Matsui's bounding conditions into one SAT problem with a minor increment on the number of clauses. At the same time, the number of variables remains the same as the standard SAT method. Note that numerous bounding

conditions are available. In the next section, we discuss which sets of bounding conditions produce better accelerating effect.

4 Accelerating Effect of the Bounding Condition

Suppose that we aim at the R-round differential trail with the weight being no more than k, the global constraint should be $\sum_{i=0}^{R-1}\sum_{j=0}^{\varpi-1} w_j^{(i)} \leqslant k$, where we reuse the symbols in Sect. 3.1. Given the probability bounds $\mathrm{Pr}_{\mathtt{Opt}}(i)$ for $1 \leqslant i \leqslant R-1$, in theory, we can generate $C_R^2 - 1$ bounding conditions of the following form

$$\sum_{i=r_1}^{r_2}\sum_{j=0}^{\varpi-1} w_j^{(i)} \leqslant k + \log_2\left(\mathrm{Pr}_{\mathtt{Opt}}(r_1)\right) + \log_2\left(\mathrm{Pr}_{\mathtt{Opt}}(R-r_2-1)\right),$$

where $0 \leqslant r_1 \leqslant r_2 \leqslant R-1$, and r_1 and r_2 cannot reach the two endpoints, simultaneously. For simplicity, we denote the bounding condition starting from the r_1-th round and terminating with the r_2-th round as $\mathcal{C}_{(r_1,r_2)}$. Many queries should be answered.

- Whether the automatic search with the SAT method can be accelerated after integrating some of these bounding conditions?
- If we add all the bounding conditions into the SAT problem, does it result in the best performance of the search with the SAT solver?
- If this is not the case, which sets of bounding conditions potentially result in extraordinary advances?

In this section, we take the distinguisher searching problem of `GIFT-64` [BPP+17], which is a 28-round SPN cipher with the 64-bit block size, as an illustration, and compare the runtime for solving SAT problems with different sets of bounding conditions. By taking into account the observations in the test as well as our experience, we try to find answers for the above problems. At the end of this section, we provide a strategy concerning the selection of the sets of bounding conditions, which may be helpful for designers and attackers in search of differential and linear characteristics. All the tests in this section are implemented on a PC with Intel® Core™ i5-9400F CPU @ 2.90GHz × 6, and we only use one core.

We set the goal as searching for the optimal differential trails with the minimum number of active S-boxes for `GIFT-64` from 1-round to 28-round. After initialising both the number of rounds R and the number of active S-boxes τ as one, we invoke the SAT solver to determine the existence of the R-round trail with no more than τ active S-boxes. If this prediction is satisfiable, we obtain an R-round trail with τ active S-boxes, and the searching phase proceeds after respectively increasing the values of R and τ[3] by one. Otherwise, we update the value of τ with $\tau + 1$ and ask the SAT solver to verify the satisfiability. This procedure is terminated until we get the 28-round trail with the minimum number of active S-boxes. Denote $\#\mathtt{S}_{\mathtt{D}}(i)$ the minimum number of active S-boxes achieved by i-round differential trails. Thus, in this procedure, we solve $\#\mathtt{S}_{\mathtt{D}}(28)$ SAT problems in total. In the following, we view the runtime for solving the $\#\mathtt{S}_{\mathtt{D}}(28)$ problems as a criterion and compare the runtime under different settings that integrate different sets of bounding conditions. The runtime for the standard SAT method with no bounding condition is 4306.9s, which is a benchmark for the accelerating effect.

[3] For SPN ciphers with nonlinear layers composed of parallel S-boxes, the minimum number of active S-boxes strictly rises with the increment of the number of rounds.

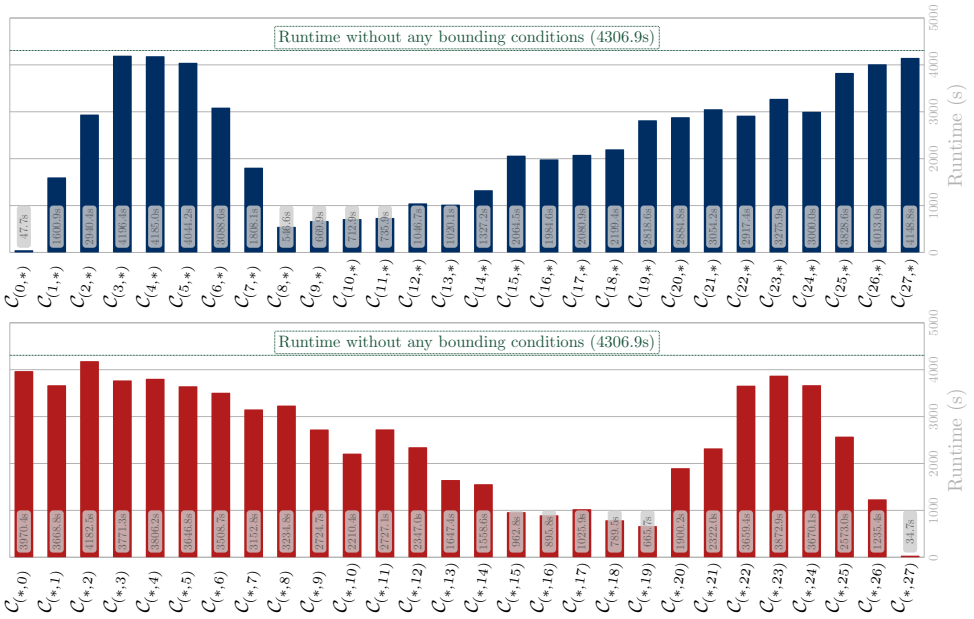


Figure 4: Runtime regarding the sets $\mathcal{C}_{(r_1,*)}$ and $\mathcal{C}_{(*,r_2)}$ for `GIFT-64`.

4.1 Sets of Bounding Conditions with the Same Initial/Terminal Round

In [ZSCH18], by incorporating the Matsui's bounding conditions originating from the first round and (or) terminating with the last round, Zhang et al. realised a speedup on the search with the MILP method. Inspired by this work, we wonder the accelerating outcome of the set comprising bounding conditions with the same initial or terminal round on the SAT method. Denote $\mathcal{C}_{(r_1,*)}$ the set of $28 - r_1$ bounding conditions starting from the r_1-th round and $\mathcal{C}_{(*,r_2)}$ the set of r_2 bounding conditions terminating with the r_2-th round, $0 \leqslant r_1, r_2 \leqslant 27$. After encoding the 56 sets of conditions into SAT problems, we conduct the test, separately, and present an intuitive comparison of the runtime in Figure 4.

From the results illustrated in Figure 4, we note that all the 56 sets $\mathcal{C}_{(r_1,*)}$ and $\mathcal{C}_{(*,r_2)}$ indeed shorten the runtime. This observation allows us to provide a positive answer for the first issue, that is, the automatic search with the SAT method can be accelerated after integrating some of these bounding conditions. Besides, it also can be notified that the degrees of improvements for different sets exhibit apparent variation. The set $\mathcal{C}_{(*,27)}$ results in the best performance in the test, and it only takes about $34.7/4306.9 \approx 0.8\%$ of the runtime for the standard SAT method to get precisely the same result. Also, the performance regarding the set $\mathcal{C}_{(0,*)}$ is good, although the corresponding runtime is slightly longer than that of $\mathcal{C}_{(*,27)}$. Lastly, the runtime has a sharp decline at the two points $\mathcal{C}_{(8,*)}$ and $\mathcal{C}_{(*,19)}$ in Figure 4. In the same test regarding `PRESENT`, a similar decline occurs at the two points $\mathcal{C}_{(5,*)}$ and $\mathcal{C}_{(*,25)}$, which is shown in Figure 5. We conjecture this circumstance relates to the structure of the cipher as well as the optimising technique in the SAT solver and leave this issue as future work.

4.2 Unions of Multiple Sets Defined in Sect. 4.1

Now, we study whether the performance of the automatic search can be further improved by taking multiple sets defined in Sect. 4.1 into account. Since the two sets $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,27)}$ obtain overwhelming advantages over the remaining ones, we generate the union

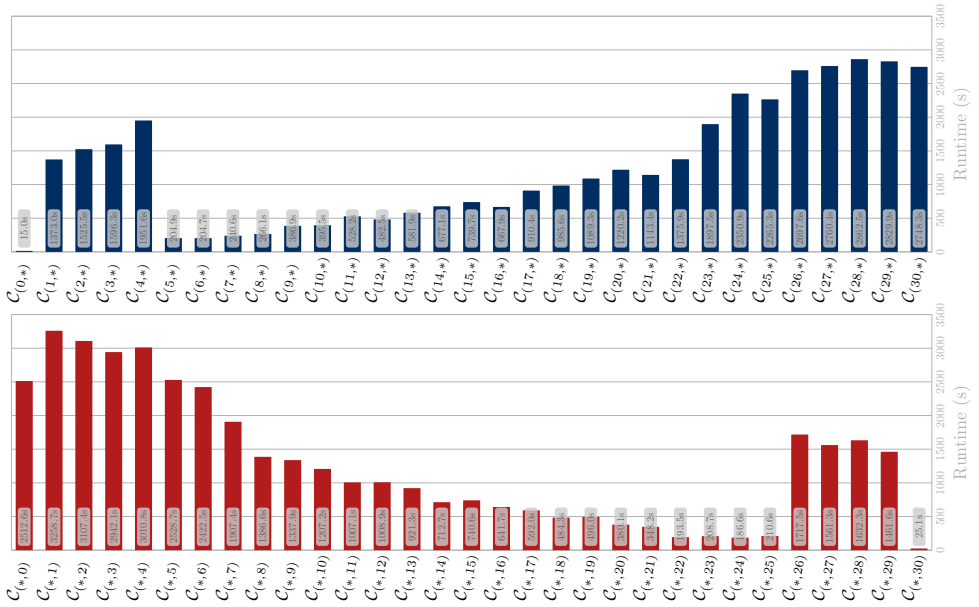


Figure 5: Runtime regarding the sets $\mathcal{C}_{(r,*)}$ and $\mathcal{C}_{(*,r)}$ for PRESENT.

sets $\mathcal{U}_{(r,*)} = \bigcup_{r_1=0}^{r-1} \mathcal{C}_{(r_1,*)}$ and $\mathcal{U}_{(*,r)} = \bigcup_{r_2=28-r}^{27} \mathcal{C}_{(*,r_2)}$ $(2 \leqslant r \leqslant 28)$ by accumulating multiple sets based on $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,27)}$, respectively. Both $\mathcal{U}_{(r,*)}$ and $\mathcal{U}_{(*,r)}$ are composed of r sets defined in Sect. 4.1. A comparison on the runtimes under the 54 union sets can be found in Figure 6. All the 54 sets achieve improvements on the runtime in almost equal measure. Thus, probably, we cannot significantly improve the runtime with $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,27)}$ by combining multiple sets like $\mathcal{C}_{(r_1,*)}$ or $\mathcal{C}_{(*,r_2)}$ into the SAT problem. For the two sets $\mathcal{U}_{(28,*)}$ and $\mathcal{U}_{(*,28)}$ containing all the bounding conditions, the runtime does not attain the minimum value. This observation indicates that adding all the bounding conditions into the SAT problem does not always give the best performance. Furthermore, at the two points $\mathcal{U}_{(*,2)}$ and $\mathcal{U}_{(*,4)}$ in Figure 6, the tests get minor acceleration over the result with $\mathcal{C}_{(*,27)}$, which is the origin of the following conjecture. When the tests with $C_{(0,*)}$ and $C_{(*,\mathcal{R}-1)}$ do not meet the requirement in the application for a primitive with $\mathcal{R}$ rounds of encryption, the union sets $\mathcal{U}_{(r,*)}$ and $\mathcal{U}_{(*,r)}$ with r being a small integer might be the last hope for better returns under the searching framework in this paper.

4.3 Sets of Conditions Covering the Same Number of Rounds

We also analyse the accelerating effect of the set of bounding conditions covering the same number of rounds. Denote $\mathcal{C}_{|r|}$ the set of $28 - r$ bounding conditions covering r-round of encryption, i.e., $\mathcal{C}_{|r|} = \{\mathcal{C}_{(x,x+r-1)} \mid 0 \leqslant x \leqslant 28 - r\}$, $1 \leqslant r \leqslant 27$. The runtime is illustrated in Figure 7. It can be notified that this kind of set speeds up the search when the value of r is relatively small. Nevertheless, the performance is getting worse with the increasing value of r, since r-round bounding conditions cannot be united into the R-round optimal trail searching problem with $R < r$. Moreover, we should remind that the automatic search with all the sets $\mathcal{C}_{|r|}$ cannot get better performance than those achieved by $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,27)}$.

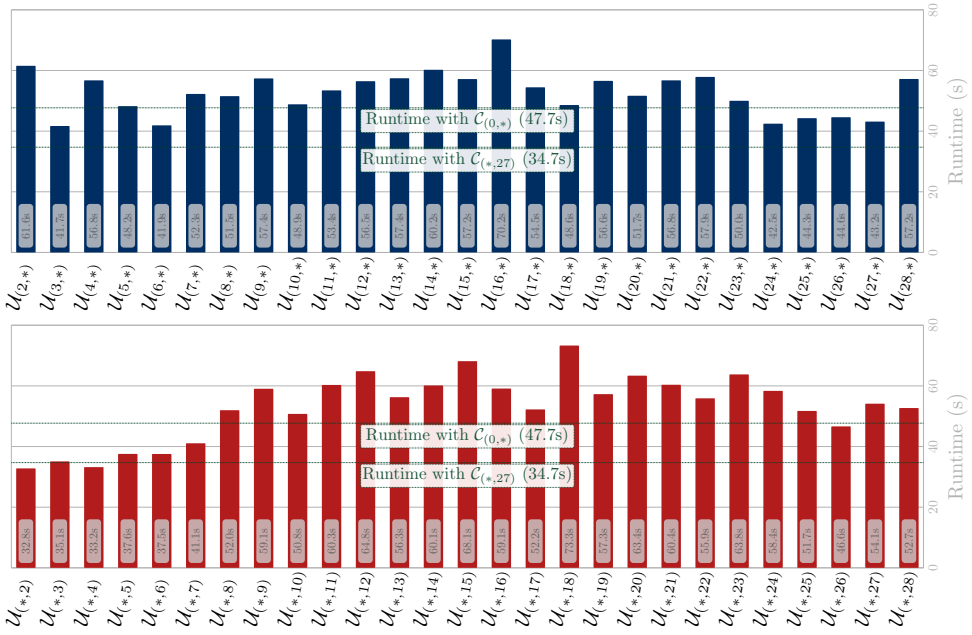


Figure 6: Runtime regarding the sets $\mathcal{U}_{(r,*)}$ and $\mathcal{U}_{(*,r)}$ for `GIFT-64`.

4.4 How to Select the Sets of Bounding Conditions

Now, we sum up strategies concerning the selection of the sets of bounding conditions in the automatic search for $\mathcal{R}$-round primitives with the SAT method. According to a considerable amount of experiments for different primitives and our experience, these strategies can be generally applied to various block ciphers, even though these ideas are explained with the tests on `GIFT-64`.

First of all, we think the two sets $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,\mathcal{R}-1)}$ are the first choice and are more likely to show remarkable improvements in the runtime over the standard method with no bounding condition. Secondly, if the performances with $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,\mathcal{R}-1)}$ do not meet the requirement, the union sets $\mathcal{U}_{(r,*)}$ and $\mathcal{U}_{(*,r)}$ with r being a small integer worth a shot.

The last thing we want to mention is that we also study the efficiency of sets with randomly drawn bounding conditions and evaluate the outcome, correspondingly. The accelerating effect is not visible when the number of conditions in the set is not adequate. Also, the improvements cannot outperform those of $\mathcal{C}_{(0,*)}$ and $\mathcal{C}_{(*,\mathcal{R}-1)}$. Therefore, we do not recommend using random sets.

Remark 1. Since we adopt CaDiCaL instead of Cryptominisat as the SAT solver, it is natural to question whether the acceleration is just achieved by using a different solver. To clearly illustrate the gain by the new encoding approach, we provide a comprehensive comparison of the runtime for various primitives regarding distinct searching targets. The comparison takes the two solvers as mentioned above and different sets of bounding conditions into account. It can be notified from the results that altering the solver is not the essential reason for the acceleration, and the significant improvement mainly benefits from the new encoding approach. Please refer to Appendix E for more details.

5 Applications to Several Block Ciphers

In this section, we apply the ideas in Sect. 3 and Sect. 4 to several block ciphers and gain many new results. All the tests in this section are performed by integrating the

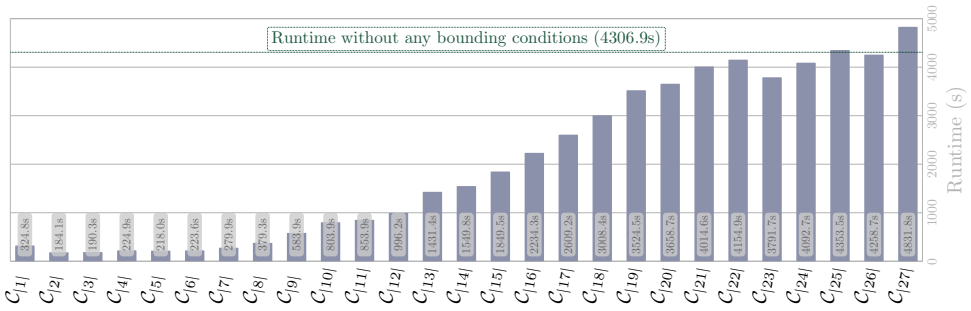


Figure 7: Runtime regarding the sets $\mathcal{C}_{|r|}$ for `GIFT-64`.

set $\mathcal{C}_{(*,\mathcal{R}-1)}$, which is composed of bounding conditions terminating with the last round, into the SAT method. For the primitives studied in [ZZDX19] with the MILP method, we give a comparison on the runtime. The comparisons are not fair since the tests are implemented on different platforms. However, in all comparisons, our tests with the SAT method operate much faster than those with the MILP method. The source codes are publicly available at https://github.com/SunLing134340/Accelerating_Automatic_Search. For simplicity, we introduce the following notations.

- $\#\mathtt{S}_{\mathtt{D}}$: the minimum number of differential active S-boxes.
- $\#\mathtt{S}_{\mathtt{L}}$: the minimum number of linear active S-boxes.
- $\mathrm{Pr}_{\mathtt{Opt}}$: the maximum probability of differential trails.
- $\mathrm{Cor}_{\mathtt{Opt}}$: the maximum correlation of linear trails.
- $T_{\mathtt{SAT}}$: runtime in our tests on a PC with Intel® Core™ i5-9400F CPU @ 2.90GHz.
- $T_{\mathtt{MILP}}$: runtime in [ZZDX19] on a PC with Intel® Core™ i7-4790 CPU @ 3.60GHz.

5.1 Applications to Three SPN Ciphers

Applications to `PRESENT`. `PRESENT` [BKL+07] is a lightweight SPN cipher proposed by Bogdanov et al. at CHES 2007. It consists of 31 rounds, and the block size is 64-bit. `PRESENT` is probably one of the first candidates that take lightweight hardware implementations into account and has a profound effect on the design of lightweight block ciphers. For `PRESENT`, we obtain full information about $\#\mathtt{S}_{\mathtt{D}}$, $\mathrm{Pr}_{\mathtt{Opt}}$, $\#\mathtt{S}_{\mathtt{L}}$ and $\mathrm{Cor}_{\mathtt{Opt}}$ from 1-round to 31-round. As far as we know, we are the first one to provide all these results. The experimental results are covered in Table 3.

Applications to `GIFT`. As an improved version of `PRESENT`, `GIFT` [BPP+17] attains a much-increased efficiency in all domains. Meanwhile, it corrects the well-known weakness of `PRESENT` with regards to linear hulls. `GIFT` is composed of two versions. `GIFT-64` is a 28-round SPN cipher with the 64-bit block size, and `GIFT-128` is a 40-round SPN cipher with the 128-bit block size.

- For `GIFT-64`, full information about $\#\mathtt{S}_{\mathtt{D}}$, $\mathrm{Pr}_{\mathtt{Opt}}$, $\#\mathtt{S}_{\mathtt{L}}$ and $\mathrm{Cor}_{\mathtt{Opt}}$ is known, and the test results can be found in Table 4.
- For `GIFT-128`, we get complete knowledge of $\#\mathtt{S}_{\mathtt{D}}$ and $\#\mathtt{S}_{\mathtt{L}}$ from 1-round to 40-round. Moreover, we discover the optimal differential trails for up to 29 rounds and the optimal linear characteristics for up to 25 rounds. Please check Table 5 for more details.

Applications to RECTANGLE. RECTANGLE [ZBL+15] is a 25-round SPN cipher with the 64-bit block size. It facilitates fast implementations for multiple platforms by using bit-slice techniques. Full knowledge about $\#S_D$, Pr_{Opt}, $\#S_L$ and Cor_{Opt} is explicit with the acceleration method. The test results are provided in Table 6.

5.2 Applications to Two Feistel Ciphers

For the two ciphers LBlock [WZ11] and TWINE [SMMK12] with Feistel structures, the entire messages about $\#S_D$, Pr_{Opt}, $\#S_L$ and Cor_{Opt} are clear. Please find in Table 7 and Table 8 the results of LBlock and TWINE, respectively.

5.3 Applications to SIMON and SPECK Families of Block Ciphers

Applications to SIMON family of block ciphers. We obtain the full learning of Pr_{Opt} and Cor_{Opt} for all versions in the SIMON family of block ciphers [BSS+13]. As mentioned in Sect. 2.3.2, we claim that the result of the value Cor_{Opt} for SIMON is heuristic. Please find the test results about SIMON in Table 9.

Applications to SPECK family of block ciphers. The acceleration method is also practised on all versions of SPECK family of block ciphers. Especially for the two versions SPECK32 and SPECK64, we get complete pictures of Pr_{Opt} and Cor_{Opt}. In the test for SPECK, we notice that adding bound conditions cannot significantly improve the automatic search with the SAT method. This circumstance coincides with the observation raised by Zhang et al. [ZSCH18]. That is, adding Matsui's bounding conditions into MILP models of ARX ciphers is not a good choice.

5.4 Related-Key Differential Attack on 26-Round GIFT-64

Note that the acceleration method also can be utilised to speed up the search of related-key differential characteristics if we view the n-bit cipher with k-bit master key as a function with an $(n+k)$-bit input and an n-bit output. With this method, for GIFT-64, we discover an 18-round related-key differential trail with probability 2^{-58}, which is presented in Figure 8. The 128-bit master key of this trail is 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0028 0x0000. Since it is decoded from the experimental result of the search concerning the minimum number of active S-boxes, we do not claim it is an optimal 18-round related-key differential characteristic.

With the 18-round distinguisher, we launch a related-key differential attack on 26-round GIFT-64 by appending three and five rounds before and after the distinguisher, respectively. Please find in Appendix D.2 for a brief description of GIFT-64. The key-recovery attack is demonstrated in Figure 9, where X^i and Y^i denote the 64-bit input and output of the **SubCells** operation in the i-th round ($0 \leqslant i \leqslant 25$), and RK^i stands for the i-th round key. We employ $X^i[j]$ to represent the j-th bit of X^i.

Since there is no whitening key at the input of GIFT-64, we can construct structures at the position of Y^0. In each structure, we fix the value of the eight bits $Y^0[16, 20, 21, 25, 33, 40, 44, 45]$ marked with 'Δ' in Figure 9 and traverse all the values of the remaining 56 bits. Then, one pair is generated by respectively drawing one element from two structures with the fixed 8-bit value being opposite with each other. Thus, 2^{112} pairs can be created with two structures composed of 2^{57} elements.

In the attack, we prepare $\mathcal{S}$ twin structures and obtain $N_1 = \mathcal{S} \cdot 2^{112}$ pairs. So, the data complexity is $\mathcal{S} \cdot 2^{57}$. For each pair (Y^0, Y'^0), we compute the pair of plaintexts (P, P') by applying GS^{-1} to every nibble of the two states (Y^0, Y'^0). By querying the oracle, we obtain the corresponding pair of ciphertexts (C, C'). To reduce the time complexity in the

subsequent round key recovery phase, we apply the partial sum technique and take the property of the key schedule into account.

The 32-bit state of RK^0 is partitioned into 16 parts. We guess the value of $RK^0[0,1]$ and check whether the 4-bit difference $\Delta Y^1[0-3]$ fills the condition $\Delta Y^1[0] = \Delta Y^1[1] = \Delta Y^1[2] = 0$. The remaining $N_1 \cdot 2^{-3}$ pairs will participate in the following processes. This guess-and-check procedure is repeated for all the 16 parts until all the 32-bit value of RK^0 is traversed. The time complexity and the number of remaining pairs in each step are detailed in Table 11. After the enumeration of RK^0, about $N_1 \cdot 2^{-46}$ pairs are left. Then, we proceed with the enumeration of the 8-bit value $RK^1[16-19, 28-31]$. Similarly, this procedure is split into four parts. For the first part regarding $RK^1[16,17]$, we guess the 2-bit value and check the validity of the condition $\Delta Y^2[32] = \overline{\Delta Y^2[33]} = \Delta Y^2[34] = \Delta Y^2[35] = 0$. According to the DDT of the S-box GS, if the input difference is '$11**$', the prediction that the output difference equals `0x4` holds with probability 2^{-2}. Thus, $N_1 \cdot 2^{-46} \cdot 2^{-2}$ pairs fulfiling this constraint will receive further consideration. We repeat this procedure for the remaining three parts of RK^1. The detailed analysis can be found in Table 11. After the enumeration of RK^1, we obtain $N \triangleq N_1 \cdot 2^{-56}$ pairs that match the input difference of the 18-round distinguisher.

Now, we turn to the tail of the distinguisher. With the property of the key schedule, we find that the 8-bit value of $RK^{25}[1,3,8,10,13,15,20,22]$ is known since it corresponds to the guessed 8-bit value of RK^1. Also, the 32-bit value of RK^{24} is known as it is the output of a bit permutation on RK^0. Hence, to calculate the values of the pairs (X^{24}, X'^{24}), we only need to guess the 24-bit unknown value of RK^{25}. The time complexity of this step is $2 \cdot N_1 \cdot 2^{-56} \cdot 2^{40} \cdot 2^{24} \cdot \frac{2}{26} \approx N_1 \cdot 2^{5.30}$ 26-round of encryptions. The following steps concerning the enumerations of RK^{23} and RK^{22} are similar to those performed on RK^0 and RK^1. The evaluation for the complexity is listed in Table 11.

We set a counter to record the number of right pairs that validate the input and output differences of the 18-round distinguisher. At last, for random key guesses, the number of right pairs is about $N_1 \cdot 2^{-120}$. For the right key guess, the number of right pairs is expected to be $N_1 \cdot 2^{-56} \cdot 2^{-58} = N_1 \cdot 2^{-114}$. Thus, the number of right pairs follows a binomial distribution with parameters $(N, p_0 = 2^{-58})$ in the case of the good key and $(N, p_1 = 2^{-64})$ otherwise. We fix the threshold as Θ, and the key guess will be accepted as a candidate if the counter of right pairs is no less than Θ. Note that we already guess the value of the 112-bit in the master key. For all surviving key candidates, we exhaustively search for the value of the remaining 16-bit with at most two plaintext-ciphertext pairs.

Complexity Analysis We apply the method in [BGT11] to estimate the complexity. Let α stand for the non-detection error probability and β be the false alarm error probability. Then, we have the following approximations for the values of α and β,

$$\begin{aligned} \beta &\overset{N\to\infty}{\sim} \frac{(1-p_1)\sqrt{\Theta/N}}{(\Theta/N-p_1)\sqrt{2\pi N(1-\Theta/N)}} \exp\left[-N \cdot D\left(\frac{\Theta}{N}\middle\| p_1\right)\right], \\ \alpha &\overset{N\to\infty}{\sim} \frac{p_0\sqrt{1-(\Theta-1)/N}}{(p_0-(\Theta-1)/N)\sqrt{2\pi(\Theta-1)}} \exp\left[-N \cdot D\left(\frac{\Theta-1}{N}\middle\| p_0\right)\right], \end{aligned} \tag{10}$$

where $D(p\|q) \triangleq p \cdot \ln\left(\frac{p}{q}\right) + (1-p) \cdot \ln\left(\frac{1-p}{1-q}\right)$ is the Kullback-Leibler divergence between two Bernoulli probability distributions with parameters being p and q, respectively.

The time complexity T_1 in the subkey enumeration phase is about $N_1 \cdot 2^{15.90} \cdot \frac{1}{16 \cdot 26} \approx N_1 \cdot 2^{7.20}$ 26-round of encryptions. The time complexity T_2 to exhaustively check the remaining 16-bit value in the master key is $2^{128} \cdot \beta \cdot (1-2^{-64})$ 26-round of encryptions. We set the threshold Θ as $\Theta = 2$ and try to find the minimum value of $N = N_1 \cdot 2^{-56}$ such

that the success probability $P_S = 1 - \alpha$ of the attack is not less than 90%. With Eq. (10), we compute the values $N \approx 2^{59.96}$ and $P_S \approx 90\%$. Accordingly, we have $\beta \approx 2^{-9.14}$, and the time complexity of this attack is $T_1 + T_2 \approx 2^{123.23}$. The data complexity is $\mathcal{S} \cdot 2^{57} = N \cdot 2 = 2^{60.96}$ chosen plaintexts. The memory complexity is $2^{112} \cdot \beta \approx 2^{102.86}$ for memorising the right key candidates with $\Theta \geqslant 2$.

6 Conclusion

In this paper, we try to accelerate the search of differential and linear characteristics with the SAT method. The main idea is to encode Matsui's bounding conditions by reusing the sequential counter circuit for the objective function in the standard SAT method. The novel encoding method does not rely on new auxiliary variables. It enables us to incorporate multiple bounding conditions into one SAT problem with a minor increment on the number of clauses. With the observations and experience in a considerable amount of experiments, we come up with a strategy on how to organise the sets of bounding conditions that potentially achieve better performance. This new idea is applied to various primitives and obtains many updated cryptanalytic results.

As we mentioned in the paper, we observe a striking drop in the runtime regarding the tests with the sets $\mathcal{C}_{(r_1,*)}$ and $\mathcal{C}_{(*,r_2)}$ for `GIFT-64` and `PRESENT`. We think that figuring out the reason for this circumstance is an interesting future work. Probably, the reason may result in new ideas to further accelerate the automatic search with the SAT method.

Acknowledgments

The authors would like to thank the shepherd Stefan Kölbl and the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper. The research leading to these results has received funding from the National Natural Science Foundation of China (Grant No. 62002201, Grant No. 62032014), the National Key Research and Development Program of China (Grant No. 2018YFA0704702), the Major Scientific and Technological Innovation Project of Shandong Province, China (Grant No. 2019JZZY010133), the Major Basic Research Project of Natural Science Foundation of Shandong Province, China (Grant No. ZR202010220025), and the Qingdao Postdoctor Application Research Project (Grant No. 61580070311101).

References

[AJN14] Jean-Philippe Aumasson, Philipp Jovanovic, and Samuel Neves. Analysis of NORX: Investigating differential and rotational properties. In *Progress in Cryptology - LATINCRYPT 2014 - Third International Conference on Cryptology and Information Security in Latin America, Florianópolis, Brazil, September 17-19, 2014, Revised Selected Papers*, pages 306–324, 2014.

[AK18] Ralph Ankele and Stefan Kölbl. Mind the gap - A closer look at the security of block ciphers against differential cryptanalysis. In *Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada, August 15-17, 2018, Revised Selected Papers*, pages 163–190, 2018.

[ARS+20] Seyyed Arash Azimi, Adrián Ranea, Mahmoud Salmasizadeh, Javad Mohajeri, Mohammad Reza Aref, and Vincent Rijmen. A bit-vector differential model for the modular addition by a constant. *IACR Cryptol. ePrint Arch.*, 2020:1025, 2020.

[BC20] Christina Boura and Daniel Coggia. Efficient MILP modelings for Sboxes and linear layers of SPN ciphers. *IACR Trans. Symmetric Cryptol.*, 2020(3):327–361, 2020.

[BGT11] Céline Blondeau, Benoît Gérard, and Jean-Pierre Tillich. Accurate estimates of the data complexity and success probability for various cryptanalyses. *Des. Codes Cryptogr.*, 59(1-3):3–34, 2011.

[Bie19] Armin Biere. CaDiCaL at the SAT Race 2019. In Marijn Heule, Matti Järvisalo, and Martin Suda, editors, *Proc. of SAT Race 2019 – Solver and Benchmark Descriptions*, volume B-2019-1 of *Department of Computer Science Series of Publications B*, pages 8–9. University of Helsinki, 2019.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. 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*, pages 123–153, 2016.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: An ultra-lightweight block cipher. In *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, pages 450–466, 2007.

[BPP+17] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching the limit of lightweight encryption. In *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, pages 321–345, 2017.

[BS90] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. In *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, pages 2–21, 1990.

[BSS+13] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK families of lightweight block ciphers. *IACR Cryptol. ePrint Arch.*, 2013:404, 2013.

[BVC16] Alex Biryukov, Vesselin Velichkov, and Yann Le Corre. Automatic search for the best trails in ARX: application to block cipher Speck. In *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, pages 289–310, 2016.

[CJF+16] Tingting Cui, Keting Jia, Kai Fu, Shiyao Chen, and Meiqin Wang. New automatic search tool for impossible differentials and zero-correlation linear approximations. *IACR Cryptol. ePrint Arch.*, 2016:689, 2016.

[Coo71] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*, pages 151–158, 1971.

[CZD19] Huaifeng Chen, Rui Zong, and Xiaoyang Dong. Improved differential attacks on GIFT-64. In *Information and Communications Security - 21st International Conference, ICICS 2019, Beijing, China, December 15-17, 2019, Revised Selected Papers*, pages 447–462, 2019.

[DLL62] Martin Davis, George Logemann, and Donald W. Loveland. A machine program for theorem-proving. *Commun. ACM*, 5(7):394–397, 1962.

[DP60] Martin Davis and Hilary Putnam. A computing procedure for quantification theory. *J. ACM*, 7(3):201–215, 1960.

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[FWG+16] Kai Fu, Meiqin Wang, Yinghua Guo, Siwei Sun, and Lei Hu. MILP-based automatic search algorithms for differential and linear trails for Speck. In *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, pages 268–288, 2016.

[GD07] Vijay Ganesh and David L. Dill. A decision procedure for bit-vectors and arrays. In Werner Damm and Holger Hermanns, editors, *Computer Aided Verification, 19th International Conference, CAV 2007, Berlin, Germany, July 3-7, 2007, Proceedings*, volume 4590 of *Lecture Notes in Computer Science*, pages 519–531. Springer, 2007.

[JS97] Roberto J. Bayardo Jr. and Robert Schrag. Using CSP look-back techniques to solve real-world SAT instances. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Innovative Applications of Artificial Intelligence Conference, AAAI 97, IAAI 97, July 27-31, 1997, Providence, Rhode Island, USA*, pages 203–208, 1997.

[JZZD20] Fulei Ji, Wentao Zhang, Chunning Zhou, and Tianyou Ding. Improved (related-key) differential cryptanalysis on GIFT. *IACR Cryptol. ePrint Arch.*, 2020:1242, 2020.

[KLT15] Stefan Kölbl, Gregor Leander, and Tyge Tiessen. Observations on the SIMON block cipher family. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, pages 161–185, 2015.

[LLJW19] Zhengbin Liu, Yongqiang Li, Lin Jiao, and Mingsheng Wang. A new method for searching optimal differential and linear trails in ARX ciphers. *IACR Cryptol. ePrint Arch.*, 2019:1438, 2019.

[LLL+19] Yu Liu, Huicong Liang, Muzhou Li, Luning Huang, Kai Hu, Chenhe Yang, and Meiqin Wang. STP models of optimal differential and linear trail for S-box based ciphers. *IACR Cryptol. ePrint Arch.*, 2019:25, 2019.

[LS19] Yunwen Liu and Yu Sasaki. Related-key boomerang attacks on GIFT with automated trail search including BCT effect. In *Information Security and Privacy - 24th Australasian Conference, ACISP 2019, Christchurch, New Zealand, July 3-5, 2019, Proceedings*, pages 555–572, 2019.

[LWR16] Yunwen Liu, Qingju Wang, and Vincent Rijmen. Automatic search of linear trails in ARX with applications to SPECK and Chaskey. In *Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings*, pages 485–499, 2016.

[LWZZ19] Lingchen Li, Wenling Wu, Yafei Zheng, and Lei Zhang. The relationship between the construction and solution of the MILP models and applications. *IACR Cryptol. ePrint Arch.*, 2019:49, 2019.

[Mat93] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, pages 386–397, 1993.

[Mat94] Mitsuru Matsui. On correlation between the order of S-boxes and the strength of DES. In *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, pages 366–375, 1994.

[MP13] Nicky Mouha and Bart Preneel. Towards finding optimal differential characteristics for ARX: Application to Salsa20. Technical report, Cryptology ePrint Archive, Report 2013/328, 2013.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. 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*, pages 57–76, 2011.

[RLA20] Adrián Ranea, Yunwen Liu, and Tomer Ashur. An easy-to-use tool for rotational-xor cryptanalysis of ARX block ciphers. *IACR Cryptol. ePrint Arch.*, 2020:727, 2020.

[RN10] Stuart J. Russell and Peter Norvig. *Artificial Intelligence - A Modern Approach, Third International Edition*. Pearson Education, 2010.

[SHW+14] Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. 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, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 158–178, 2014.

[SHY16] Ling Song, Zhangjie Huang, and Qianqian Yang. Automatic differential analysis of ARX block ciphers with application to SPECK and LEA. In *Information Security and Privacy - 21st Australasian Conference, ACISP 2016, Melbourne, VIC, Australia, July 4-6, 2016, Proceedings, Part II*, pages 379–394, 2016.

[Sin05] Carsten Sinz. Towards an optimal CNF encoding of Boolean cardinality constraints. In *Principles and Practice of Constraint Programming - CP 2005, 11th International Conference, CP 2005, Sitges, Spain, October 1-5, 2005, Proceedings*, pages 827–831, 2005.

[SLR+15] Bing Sun, Zhiqiang Liu, Vincent Rijmen, Ruilin Li, Lei Cheng, Qingju Wang, Hoda AlKhzaimi, and Chao Li. Links among impossible differential, integral and zero correlation linear cryptanalysis. In *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, pages 95–115, 2015.

[SMMK12] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. TWINE : A lightweight block cipher for multiple platforms. In *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, pages 339–354, 2012.

[SNC09] Mate Soos, Karsten Nohl, and Claude Castelluccia. Extending SAT solvers to cryptographic problems. In Oliver Kullmann, editor, *Theory and Applications of Satisfiability Testing - SAT 2009, 12th International Conference, SAT 2009, Swansea, UK, June 30 - July 3, 2009. Proceedings*, volume 5584 of *Lecture Notes in Computer Science*, pages 244–257. Springer, 2009.

[Sob10] S.K. Sobolev. Conjunctive normal form. Technical report, Encyclopedia of Mathematics, http://encyclopediaofmath.org/index.php?title=Conjunctive_normal_form&oldid=35078, 2010.

[SS96] João P. Marques Silva and Karem A. Sakallah. GRASP - a new search algorithm for satisfiability. In *Proceedings of the 1996 IEEE/ACM International Conference on Computer-Aided Design, ICCAD 1996, San Jose, CA, USA, November 10-14, 1996*, pages 220–227, 1996.

[ST17a] Yu Sasaki and Yosuke Todo. New algorithm for modeling S-box in MILP based differential and division trail search. In *Innovative Security Solutions for Information Technology and Communications - 10th International Conference, SecITC 2017, Bucharest, Romania, June 8-9, 2017, Revised Selected Papers*, pages 150–165, 2017.

[ST17b] Yu Sasaki and Yosuke Todo. 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*, pages 185–215, 2017.

[Ste] Stefan Kölbl. CryptoSMT: An easy to use tool for cryptanalysis of symmetric primitives. https://github.com/kste/cryptosmt.

[SWW17] Ling Sun, Wei Wang, and Meiqin Wang. Automatic search of bit-based division property for ARX ciphers and word-based division property. In *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I*, pages 128–157, 2017.

[SWW18] Ling Sun, Wei Wang, and Meiqin Wang. More accurate differential properties of LED64 and Midori64. *IACR Trans. Symmetric Cryptol.*, 2018(3):93–123, 2018.

[TIHM17] Yosuke Todo, Takanori Isobe, Yonglin Hao, and Willi Meier. Cube attacks on non-blackbox polynomials based on division property. In *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, pages 250–279, 2017.

[WZ11] Wenling Wu and Lei Zhang. LBlock: A lightweight block cipher. In *Applied Cryptography and Network Security - 9th International Conference, ACNS 2011, Nerja, Spain, June 7-10, 2011. Proceedings*, pages 327–344, 2011.

[XZBL16] Zejun Xiang, Wentao Zhang, Zhenzhen Bao, and Dongdai Lin. Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, pages 648–678, 2016.

[ZBL+15] Wentao Zhang, Zhenzhen Bao, Dongdai Lin, Vincent Rijmen, Bohan Yang, and Ingrid Verbauwhede. RECTANGLE: A bit-slice lightweight block cipher suitable for multiple platforms. *Sci. China Inf. Sci.*, 58(12):1–15, 2015.

[ZSCH18] Yingjie Zhang, Siwei Sun, Jiahao Cai, and Lei Hu. Speeding up MILP aided differential characteristic search with Matsui's strategy. In *Information Security - 21st International Conference, ISC 2018, Guildford, UK, September 9-12, 2018, Proceedings*, pages 101–115, 2018.

[ZZDX19] Chunning Zhou, Wentao Zhang, Tianyou Ding, and Zejun Xiang. Improving the MILP-based security evaluation algorithm against differential/linear cryptanalysis using a divide-and-conquer approach. *IACR Trans. Symmetric Cryptol.*, 2019(4):438–469, 2019.

A Experimental Results of Three SPN Ciphers

PRESENT. Please find in Table 3 for the experimental results of PRESENT.

Table 3: Experimental results of PRESENT.

Round	Differential property						Linear property					
	$\#S_D$	T_{SAT}	T_{MILP}	Pr_{Opt}	T_{SAT}	T_{MILP}	$\#S_L$	T_{SAT}	T_{MILP}	Cor_{Opt}	T_{SAT}	T_{MILP}
1	1	**0.0s**	0s	2^{-2}	**0.0s**	1s	1	**0.0s**	0s	2^{-1}	**0.0s**	0s
2	2	**0.0s**	1s	2^{-4}	**0.0s**	2s	2	**0.0s**	2s	2^{-2}	**0.0s**	2s
3	4	**0.1s**	2s	2^{-8}	**0.2s**	3s	3	**0.0s**	3s	2^{-4}	**0.1s**	71s
4	6	**0.1s**	4s	2^{-12}	**0.4s**	4s	4	**0.0s**	6s	2^{-6}	**0.1s**	88s
5	10	**0.8s**	5s	2^{-20}	**4.3s**	5s	5	**0.0s**	9s	2^{-8}	**0.4s**	152s
6	12	**0.5s**	8s	2^{-24}	**2.6s**	249s	6	**0.0s**	8s	2^{-10}	**0.6s**	128s
7	14	**0.4s**	10s	2^{-28}	**3.2s**	9s	7	**0.0s**	7s	2^{-12}	**0.9s**	18s
8	16	**0.5s**	11s	2^{-32}	**2.5s**	11s	8	**0.0s**	8s	2^{-14}	**1.4s**	98s
9	18	**0.4s**	15s	2^{-36}	**2.7s**	14s	9	**0.0s**	10s	2^{-16}	**1.4s**	15s
10	20	**0.3s**	16s	2^{-41}	**4.7s**	1298s	10	**0.1s**	11s	2^{-18}	**1.7s**	300s
11	22	**0.3s**	18s	2^{-46}	**9.5s**	438s	11	**0.1s**	12s	2^{-20}	**1.8s**	11s
12	24	**0.4s**	22s	2^{-52}	**14.8s**	311s	12	**0.1s**	14s	2^{-22}	**1.7s**	978s
13	26	**0.4s**	24s	2^{-56}	**6.8s**	22s	13	**0.1s**	15s	2^{-24}	**2.1s**	14s
14	28	**0.4s**	31s	2^{-62}	**23.9s**	18859s	14	**0.1s**	17s	2^{-26}	**2.7s**	3507s
15	30	**0.5s**	32s	2^{-66}	**7.2s**	2594s	15	**0.1s**	19s	2^{-28}	**4.9s**	16s
16	32	**0.7s**	19s	2^{-70}	**6.3s**	370s	16	**0.1s**	21s	2^{-30}	**7.8s**	3080s
17	34	**0.6s**	20s	2^{-74}	**9.7s**	20s	17	**0.1s**	23s	2^{-32}	**5.5s**	16302s
18	36	**0.8s**	22s	2^{-78}	**7.6s**	629s	18	**0.1s**	24s	2^{-34}	**6.6s**	14105s
19	38	**0.6s**	34s	2^{-82}	**7.3s**	-	19	**0.1s**	26s	2^{-36}	**3.7s**	-
20	40	**0.9s**	29s	2^{-86}	**6.7s**	-	20	**0.1s**	28s	2^{-38}	**11.8s**	-
21	42	**0.9s**	28s	2^{-90}	**7.3s**	-	21	**0.1s**	30s	2^{-40}	**6.5s**	-
22	44	**1.1s**	29s	2^{-96}	**21.4s**	-	22	**0.1s**	34s	2^{-42}	**7.8s**	-
23	46	**1.2s**	37s	2^{-100}	**9.7s**	-	23	**0.1s**	35s	2^{-44}	**9.3s**	-
24	48	**1.8s**	34s	2^{-106}	**35.8s**	-	24	**0.1s**	37s	2^{-46}	**9.5s**	-
25	50	**1.7s**	36s	2^{-110}	**17.1s**	-	25	**0.1s**	40s	2^{-48}	**10.0s**	-
26	52	**1.2s**	38s	2^{-116}	**45.6s**	-	26	**0.2s**	42s	2^{-50}	**10.9s**	-
27	54	**1.3s**	40s	2^{-120}	**16.3s**	-	27	**0.2s**	44s	2^{-52}	**10.6s**	-
28	56	**1.3s**	42s	2^{-124}	**17.5s**	-	28	**0.2s**	46s	2^{-54}	**11.8s**	-
29	58	**1.3s**	42s	2^{-128}	**13.9s**	-	29	**0.2s**	49s	2^{-56}	**18.8s**	-
30	60	**2.0s**	44s	2^{-132}	**15.1s**	-	30	**0.2s**	49s	2^{-58}	**18.1s**	-
31	62	**1.8s**	47s	2^{-136}	**15.1s**	-	31	**0.2s**	51s	2^{-60}	**14.5s**	-
Total		**24.4s**	740s		**335.2s**	6.9h		**2.9s**	720s		**182.9s**	10.8h

GIFT-64. Please find in Table 4 for the experimental results of GIFT-64.

Table 4: Experimental results of GIFT-64.

Round	Differential property						Linear property					
	$\#S_D$	T_{SAT}	T_{MILP}	Pr_{Opt}	T_{SAT}	T_{MILP}	$\#S_L$	T_{SAT}	T_{MILP}	Cor_{Opt}	T_{SAT}	T_{MILP}
1	1	**0.0s**	1s	$2^{-1.415}$	**0.0s**	1s	1	**0.0s**	0s	2^{-1}	**0.0s**	0s
2	2	**0.0s**	2s	$2^{-3.415}$	**0.0s**	47s	2	**0.0s**	1s	2^{-2}	**0.0s**	2s
3	3	**0.0s**	3s	2^{-7}	**0.2s**	108s	3	**0.0s**	3s	2^{-3}	**0.0s**	3s
4	5	**0.1s**	69s	$2^{-11.415}$	**0.5s**	291s	5	**0.1s**	61s	2^{-5}	**0.1s**	77s
5	7	**0.1s**	61s	2^{-17}	**3.9s**	849s	7	**0.2s**	60s	2^{-7}	**0.2s**	99s
6	10	**0.6s**	144s	$2^{-22.415}$	**5.6s**	181s	9	**0.4s**	65s	2^{-10}	**0.7s**	160s
7	13	**1.9s**	115s	$2^{-28.415}$	**25.7s**	385s	12	**1.1s**	177s	2^{-13}	**1.9s**	225s
8	16	**3.3s**	271s	2^{-38}	**533.8s**	19934s	15	**2.6s**	243s	2^{-16}	**5.0s**	263s
9	18	**2.3s**	28s	2^{-42}	**23.1s**	32s	18	**8.7s**	493s	2^{-20}	**46.4s**	8713s
10	20	**1.0s**	124s	2^{-48}	**92.2s**	7569s	20	**3.0s**	681s	2^{-25}	**331.0s**	11615s
11	22	**1.6s**	77s	2^{-52}	**34.5s**	121s	22	**2.4s**	392s	2^{-29}	**1181.1s**	34019s
12	24	**1.5s**	19s	2^{-58}	**64.1s**	61001s	24	**3.4s**	3206s	2^{-31}	**298.4s**	14644s
13	26	**1.4s**	75s	2^{-62}	**32.7s**	604s	26	**4.5s**	11229s	2^{-34}	**1185.1s**	121716s
14	28	**1.0s**	15s	2^{-68}	**63.3s**	9121s	28	**4.2s**	7982s	2^{-37}	**891.9s**	-
15	30	**0.9s**	17s	2^{-72}	**45.3s**	1595s	30	**1.4s**	18410s	2^{-40}	**2106.1s**	-
16	32	**1.3s**	18s	2^{-78}	**74.2s**	-	32	**1.1s**	-	2^{-43}	**2451.7s**	-
17	34	**1.1s**	-	2^{-82}	**31.1s**	-	34	**1.3s**	-	2^{-46}	**2973.2s**	-
18	36	**1.2s**	-	2^{-88}	**99.3s**	-	36	**1.2s**	-	2^{-49}	**2261.3s**	-
19	38	**1.1s**	-	2^{-92}	**56.8s**	-	38	**1.5s**	-	2^{-52}	**1675.2s**	-
20	40	**1.9s**	-	2^{-98}	**120.7s**	-	40	**1.4s**	-	2^{-55}	**1773.6s**	-
21	42	**1.2s**	-	2^{-102}	**67.5s**	-	42	**2.5s**	-	2^{-58}	**1550.9s**	-
22	44	**1.3s**	-	2^{-108}	**209.6s**	-	44	**1.4s**	-	2^{-61}	**639.3s**	-
23	46	**2.0s**	-	2^{-112}	**50.8s**	-	46	**4.3s**	-	2^{-64}	**461.9s**	-
24	48	**1.4s**	-	2^{-118}	**124.4s**	-	48	**2.8s**	-	2^{-67}	**1279.2s**	-
25	50	**1.3s**	-	2^{-122}	**81.1s**	-	50	**2.6s**	-	2^{-70}	**1479.2s**	-
26	52	**1.4s**	-	2^{-128}	**123.5s**	-	52	**1.7s**	-	2^{-73}	**623.4s**	-
27	54	**1.4s**	-	2^{-132}	**104.2s**	-	54	**1.9s**	-	2^{-76}	**1079.4s**	-
28	56	**2.1s**	-	2^{-138}	**142.3s**	-	56	**3.2s**	-	2^{-79}	**1125.2s**	-
Total		**34.7s**	1039s		**2210.6s**	28.3h		**59.1s**	11.9h		**7.1h**	53.2h

GIFT-128. Please find in Table 5 for the experimental results of GIFT-128.

Table 5: Experimental results of GIFT-128.

Round	Differential property		Linear property		Round	Differential property		Linear property	
	$\#S_D$	Pr_{Opt}	$\#S_L$	Cor_{Opt}		$\#S_D$	Pr_{Opt}	$\#S_L$	Cor_{Opt}
1	1	$2^{-1.415}$	1	2^{-1}	21	51	$2^{-126.415}$	54	2^{-68}
2	2	$2^{-3.415}$	2	2^{-2}	22	54	$2^{-132.415}$	57	2^{-74}
3	3	2^{-7}	3	2^{-3}	23	57	$2^{-139.415}$	61	2^{-79}
4	5	$2^{-11.415}$	5	2^{-5}	24	60	$2^{-146.83}$	65	2^{-82}
5	7	2^{-17}	7	2^{-7}	25	63	$2^{-157.415}$	69	2^{-86}
6	10	$2^{-22.415}$	9	2^{-10}	26	65	$2^{-162.415}$	72	-
7	13	$2^{-28.415}$	12	2^{-13}	27	68	$2^{-168.415}$	74	-
8	17	2^{-39}	14	2^{-17}	28	71	$2^{-174.415}$	77	-
9	19	$2^{-45.415}$	18	2^{-22}	29	74	$2^{-181.83}$	79	-
10	21	$2^{-49.415}$	22	2^{-26}	30	77	-	82	-
11	23	$2^{-54.415}$	26	2^{-31}	31	79	-	86	-
12	26	$2^{-60.415}$	29	2^{-36}	32	82	-	89	-
13	29	$2^{-67.83}$	31	2^{-38}	33	85	-	91	-
14	33	2^{-79}	33	2^{-41}	34	88	-	95	-
15	35	$2^{-85.415}$	36	2^{-45}	35	91	-	99	-
16	37	$2^{-90.415}$	39	2^{-48}	36	93	-	102	-
17	40	$2^{-96.415}$	42	2^{-51}	37	96	-	105	-
18	43	$2^{-103.415}$	46	2^{-56}	38	99	-	109	-
19	46	$2^{-110.83}$	49	2^{-59}	39	102	-	113	-
20	49	$2^{-121.415}$	52	2^{-64}	40	105	-	116	-

RECTANGLE. Please find in Table 6 for the experimental results of RECTANGLE.

Table 6: Experimental results of RECTANGLE.

Round	Differential property						Linear property					
	$\#S_D$	T_{SAT}	T_{MILP}	Pr_{Opt}	T_{SAT}	T_{MILP}	$\#S_L$	T_{SAT}	T_{MILP}	Cor_{Opt}	T_{SAT}	T_{MILP}
1	1	**0.0s**	1s	2^{-2}	**0.0s**	1s	1	**0.0s**	1s	2^{-1}	**0.0s**	0s
2	2	**0.0s**	1s	2^{-4}	**0.0s**	1s	2	**0.0s**	1s	2^{-2}	**0.0s**	1s
3	3	**0.0s**	1s	2^{-7}	**0.1s**	8s	3	**0.0s**	1s	2^{-4}	**0.1s**	5s
4	4	**0.0s**	2s	2^{-10}	**0.1s**	27s	4	**0.0s**	2s	2^{-6}	**0.1s**	9s
5	6	**0.1s**	11s	2^{-14}	**0.3s**	128s	6	**0.1s**	6s	2^{-8}	**0.2s**	41s
6	8	**0.2s**	13s	2^{-18}	**1.0s**	6s	8	**0.2s**	8s	2^{-10}	**0.4s**	6s
7	11	**0.9s**	11s	2^{-25}	**6.5s**	17s	10	**0.4s**	5s	2^{-13}	**1.9s**	15s
8	13	**0.6s**	11s	2^{-31}	**19.2s**	28s	12	**0.7s**	9s	2^{-16}	**6.0s**	24s
9	15	**1.0s**	11s	2^{-36}	**18.4s**	41s	14	**1.5s**	11s	2^{-19}	**18.4s**	78s
10	17	**1.2s**	25s	2^{-41}	**22.7s**	96s	16	**2.5s**	25s	2^{-22}	**75.6s**	260s
11	19	**1.5s**	47s	2^{-46}	**52.7s**	297s	18	**4.4s**	38s	2^{-25}	**242.3s**	1772s
12	21	**1.7s**	120s	2^{-51}	**99.2s**	669s	20	**6.5s**	131s	2^{-28}	**411.4s**	5927s
13	23	**3.2s**	597s	2^{-56}	**68.0s**	2798s	22	**10.1s**	428s	2^{-31}	**835.1s**	31491s
14	25	**1.7s**	2218s	2^{-61}	**90.2s**	12410s	24	**14.0s**	1615s	2^{-34}	**1411.4s**	177473s
15	27	**2.0s**	12753s	2^{-66}	**91.4s**	40989s	26	**23.1s**	5588s	2^{-37}	**2771.5s**	-
16	29	**2.4s**	36891s	2^{-71}	**94.4s**	-	28	**36.2s**	21352s	2^{-40}	**5037.8s**	-
17	31	**4.4s**	-	2^{-76}	**86.5s**	-	30	**39.1s**	-	2^{-42}	**1684.5s**	-
18	33	**4.6s**	-	2^{-81}	**97.7s**	-	32	**57.9s**	-	2^{-45}	**4785.4s**	-
19	35	**12.4s**	-	2^{-86}	**120.1s**	-	34	**58.3s**	-	2^{-48}	**6503.3s**	-
20	37	**8.9s**	-	2^{-91}	**131.4s**	-	36	**107.2s**	-	2^{-51}	**2.8h**	-
21	39	**5.0s**	-	2^{-96}	**233.9s**	-	38	**118.6s**	-	2^{-54}	**3.2h**	-
22	41	**12.0s**	-	2^{-101}	**225.3s**	-	40	**178.7s**	-	2^{-57}	**4.3h**	-
23	43	**11.7s**	-	2^{-106}	**254.5s**	-	42	**175.2s**	-	2^{-60}	**5.0**	-
24	45	**8.4s**	-	2^{-111}	**352.5s**	-	44	**176.8s**	-	2^{-63}	**8.7h**	-
25	47	**12.7s**	-	2^{-116}	**354.1s**	-	46	**239.9s**	-	2^{-66}	**12.0h**	-
Total		**96.6s**	14.6h		**2420.2s**	15.9h		**1251.5s**	8.1h		**42.7h**	60.3h

B Experimental Results of Two Feistel Ciphers

LBlock. Please find in Table 7 for the experimental results of LBlock.

Table 7: Experimental results of LBlock.

Round	Differential property						Linear property					
	$\#S_D$	T_{SAT}	T_{MILP}	Pr_{Opt}	T_{SAT}	T_{MILP}	$\#S_L$	T_{SAT}	T_{MILP}	Cor_{Opt}	T_{SAT}	T_{MILP}
1	0	**0.2s**	0s	1	**0.0s**	0s	0	**0.0s**	0s	1	**0.0s**	0s
2	1	**0.4s**	0s	2^{-2}	**0.1s**	1s	1	**0.0s**	0s	2^{-1}	**0.0s**	0s
3	2	**0.4s**	0s	2^{-4}	**0.1s**	0s	2	**0.0s**	0s	2^{-2}	**0.1s**	0s
4	3	**0.3s**	1s	2^{-6}	**0.1s**	1s	3	**0.0s**	0s	2^{-3}	**0.1s**	0s
5	4	**0.3s**	1s	2^{-8}	**0.1s**	1s	4	**0.0s**	1s	2^{-4}	**0.1s**	1s
6	6	**0.6s**	1s	2^{-12}	**0.4s**	1s	6	**0.1s**	1s	2^{-6}	**0.2s**	1s
7	8	**0.8s**	1s	2^{-16}	**0.8s**	1s	8	**0.2s**	1s	2^{-8}	**0.3s**	2s
8	11	**1.2s**	2s	2^{-22}	**2.7s**	2s	11	**0.5s**	2s	2^{-11}	**0.9s**	2s
9	14	**1.4s**	2s	2^{-28}	**3.5s**	2s	14	**0.8s**	2s	2^{-14}	**1.1s**	2s
10	18	**2.8s**	6s	2^{-36}	**7.1s**	6s	18	**1.3s**	6s	2^{-18}	**2.0s**	8s
11	22	**3.6s**	4s	2^{-44}	**14.8s**	4s	22	**2.1s**	4s	2^{-22}	**3.6s**	4s
12	24	**2.1s**	5s	2^{-48}	**7.8s**	5s	24	**1.2s**	6s	2^{-24}	**2.0s**	5s
13	27	**3.1s**	25s	2^{-56}	**30.4s**	812s	27	**1.8s**	38s	2^{-27}	**6.1s**	2103s
14	30	**4.9s**	8s	2^{-62}	**23.5s**	848s	30	**2.7s**	10s	2^{-30}	**4.2s**	15s
15	32	**4.3s**	19s	2^{-66}	**17.6s**	820s	32	**1.4s**	28s	2^{-33}	**8.7s**	5669s
16	35	**6.6s**	30s	2^{-72}	**37.0s**	6002s	35	**3.5s**	55s	2^{-36}	**8.3s**	-
17	36	**1.5s**	29s	2^{-76}	**23.5s**	-	36	**1.5s**	31s	2^{-37}	**4.0s**	-
18	39	**3.3s**	10s	2^{-82}	**52.0s**	-	39	**3.0s**	11s	2^{-40}	**7.0s**	-
19	41	**2.3s**	190s	2^{-86}	**49.1s**	-	41	**2.0s**	6s	2^{-42}	**4.9s**	-
20	44	**4.1s**	1828s	2^{-92}	**74.9s**	-	44	**2.6s**	40s	2^{-45}	**17.4s**	-
21	45	**1.9s**	-	2^{-96}	**29.6s**	-	45	**0.8s**	-	2^{-47}	**7.5s**	-
22	48	**3.4s**	-	2^{-102}	**64.9s**	-	48	**1.9s**	-	2^{-50}	**17.5s**	-
23	50	**2.2s**	-	2^{-106}	**48.5s**	-	50	**1.4s**	-	2^{-52}	**6.5s**	-
24	53	**3.8s**	-	2^{-112}	**68.4s**	-	53	**2.6s**	-	2^{-55}	**19.7s**	-
25	54	**2.1s**	-	2^{-115}	**22.5s**	-	54	**1.3s**	-	2^{-56}	**4.6s**	-
26	57	**3.4s**	-	2^{-121}	**69.9s**	-	57	**2.6s**	-	2^{-59}	**7.1s**	-
27	59	**2.7s**	-	2^{-126}	**84.3s**	-	59	**2.0s**	-	2^{-62}	**12.4s**	-
28	62	**4.0s**	-	2^{-131}	**57.7s**	-	62	**3.7s**	-	2^{-65}	**29.3s**	-
29	63	**2.4s**	-	2^{-135}	**50.4s**	-	63	**2.1s**	-	2^{-66}	**7.2s**	-
30	66	**5.4s**	-	2^{-141}	**70.6s**	-	66	**2.6s**	-	2^{-69}	**22.0s**	-
31	68	**4.5s**	-	2^{-146}	**111.9s**	-	68	**2.4s**	-	2^{-72}	**31.3s**	-
32	71	**6.1s**	-	2^{-151}	**89.6s**	-	71	**3.1s**	-	2^{-74}	**21.4s**	-
Total		**86.1s**	352s		**1113.7s**	2.4h		**51.4s**	242s		**257.3s**	2.2h

TWINE. Please find in Table 8 for the experimental results of TWINE.

Table 8: Experimental results of TWINE.

Round	Differential property						Linear property					
	$\#S_D$	T_{SAT}	T_{MILP}	Pr_{Opt}	T_{SAT}	T_{MILP}	$\#S_L$	T_{SAT}	T_{MILP}	Cor_{Opt}	T_{SAT}	T_{MILP}
1	0	**0.0s**	0s	1	**0.0s**	0s	0	**0.0s**	0s	1	**0.0s**	0s
2	1	**0.0s**	0s	2^{-2}	**0.0s**	0s	1	**0.0s**	0s	2^{-1}	**0.0s**	0s
3	2	**0.0s**	0s	2^{-4}	**0.1s**	0s	2	**0.0s**	0s	2^{-2}	**0.0s**	1s
4	3	**0.0s**	0s	2^{-6}	**0.1s**	1s	3	**0.0s**	1s	2^{-3}	**0.0s**	1s
5	4	**0.1s**	1s	2^{-8}	**0.1s**	1s	4	**0.0s**	1s	2^{-4}	**0.1s**	1s
6	6	**0.1s**	1s	2^{-12}	**0.3s**	1s	6	**0.1s**	1s	2^{-6}	**0.1s**	2s
7	8	**0.1s**	1s	2^{-16}	**0.6s**	1s	8	**0.1s**	2s	2^{-8}	**0.2s**	2s
8	11	**0.3s**	1s	2^{-22}	**2.0s**	2s	11	**0.2s**	3s	2^{-11}	**0.4s**	3s
9	14	**0.5s**	2s	2^{-28}	**2.9s**	2s	14	**0.5s**	3s	2^{-14}	**0.8s**	4s
10	18	**1.4s**	6s	2^{-38}	**8.2s**	52s	18	**1.4s**	10s	2^{-18}	**2.1s**	17s
11	22	**2.0s**	4s	2^{-46}	**15.3s**	49s	22	**1.9s**	4s	2^{-22}	**3.3s**	12s
12	24	**1.0s**	4s	2^{-51}	**13.4s**	63s	24	**1.2s**	6s	2^{-24}	**1.9s**	8s
13	27	**2.0s**	24s	2^{-58}	**33.2s**	7905s	27	**1.4s**	53s	2^{-27}	**2.4s**	364s
14	30	**1.9s**	14s	2^{-64}	**46.0s**	17153s	30	**2.2s**	12s	2^{-30}	**3.5s**	16s
15	32	**1.4s**	14s	2^{-68}	**17.1s**	28840s	32	**1.2s**	37s	2^{-32}	**2.4s**	261s
16	35	**3.0s**	19s	2^{-74}	**37.1s**	-	35	**2.6s**	49s	2^{-35}	**4.2s**	66s
17	36	**0.6s**	17s	2^{-77}	**11.7s**	-	36	**0.8s**	57s	2^{-36}	**1.4s**	-
18	39	**1.9s**	9s	2^{-83}	**30.9s**	-	39	**2.1s**	11s	2^{-39}	**3.5s**	-
19	41	**1.1s**	5s	2^{-88}	**41.7s**	-	41	**1.3s**	7s	2^{-41}	**2.5s**	-
20	44	**2.0s**	17s	2^{-94}	**71.3s**	-	44	**2.1s**	42s	2^{-44}	**3.7s**	-
21	45	**0.9s**	-	2^{-97}	**14.7s**	-	45	**0.8s**	-	2^{-45}	**1.2s**	-
22	48	**1.5s**	-	2^{-103}	**25.4s**	-	48	**1.4s**	-	2^{-48}	**3.5s**	-
23	50	**1.3s**	-	2^{-107}	**11.9s**	-	50	**1.1s**	-	2^{-50}	**2.3s**	-
24	53	**2.3s**	-	2^{-113}	**29.8s**	-	53	**2.0s**	-	2^{-53}	**4.6s**	-
25	54	**0.7s**	-	2^{-116}	**11.3s**	-	54	**0.9s**	-	2^{-54}	**1.6s**	-
26	57	**1.8s**	-	2^{-122}	**17.5s**	-	57	**1.7s**	-	2^{-57}	**2.9s**	-
27	59	**1.4s**	-	2^{-126}	**19.8s**	-	59	**1.7s**	-	2^{-59}	**3.4s**	-
28	62	**2.4s**	-	2^{-132}	**22.5s**	-	62	**2.4s**	-	2^{-62}	**4.4s**	-
29	63	**1.4s**	-	2^{-136}	**23.1s**	-	63	**1.1s**	-	2^{-63}	**1.7s**	-
30	66	**2.2s**	-	2^{-142}	**43.1s**	-	66	**2.2s**	-	2^{-66}	**4.6s**	-
31	68	**1.9s**	-	2^{-146}	**26.2s**	-	68	**1.8s**	-	2^{-68}	**3.2s**	-
32	71	**2.6s**	-	2^{-152}	**45.2s**	-	71	**2.2s**	-	2^{-71}	**5.4s**	-
33	72	**1.2s**	-	2^{-155}	**24.2s**	-	72	**1.2s**	-	2^{-72}	**2.7s**	-
34	75	**2.3s**	-	2^{-161}	**34.9s**	-	75	**2.6s**	-	2^{-75}	**5.7s**	-
35	77	**2.5s**	-	2^{-166}	**64.5s**	-	77	**2.1s**	-	2^{-77}	**4.7s**	-
36	80	**2.9s**	-	2^{-172}	**90.5s**	-	80	**2.9s**	-	2^{-80}	**7.0s**	-
Total		**48.9s**	139s		**836.7s**	15.0h		**47.3s**	299s		**91.6s**	758s

C Experimental Results of SIMON and SPECK

SIMON family of block ciphers. Please find in Table 9 for the experimental results.

Table 9: Experimental results of SIMON Family of Block Ciphers.

SIMON32																		
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Pr_{Opt}	1	2^{-2}	2^{-4}	2^{-6}	2^{-8}	2^{-12}	2^{-14}	2^{-18}	2^{-20}	2^{-25}	2^{-30}	2^{-34}	2^{-36}	2^{-38}	2^{-40}	2^{-42}	2^{-44}	2^{-48}
Cor_{Opt}	1	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-6}	2^{-7}	2^{-9}	2^{-10}	2^{-13}	2^{-15}	2^{-17}	2^{-18}	2^{-19}	2^{-20}	2^{-21}	2^{-22}	2^{-24}
Round	19	20	21	22	23	24	25	26	27	28	29	30	31	32				
Pr_{Opt}	2^{-50}	2^{-54}	2^{-56}	2^{-61}	2^{-66}	2^{-70}	2^{-72}	2^{-74}	2^{-76}	2^{-78}	2^{-80}	2^{-84}	2^{-86}	2^{-90}				
Cor_{Opt}	2^{-25}	2^{-27}	2^{-28}	2^{-31}	2^{-33}	2^{-35}	2^{-36}	2^{-37}	2^{-38}	2^{-39}	2^{-40}	2^{-42}	2^{-43}	2^{-45}				

SIMON48																		
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Pr_{Opt}	1	2^{-2}	2^{-4}	2^{-6}	2^{-8}	2^{-12}	2^{-14}	2^{-18}	2^{-20}	2^{-26}	2^{-30}	2^{-35}	2^{-38}	2^{-44}	2^{-46}	2^{-50}	2^{-52}	2^{-57}
Cor_{Opt}	1	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-6}	2^{-7}	2^{-9}	2^{-10}	2^{-13}	2^{-15}	2^{-18}	2^{-19}	2^{-22}	2^{-23}	2^{-25}	2^{-26}	2^{-29}
Round	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Pr_{Opt}	2^{-59}	2^{-63}	2^{-65}	2^{-70}	2^{-72}	2^{-76}	2^{-78}	2^{-84}	2^{-88}	2^{-93}	2^{-96}	2^{-102}	2^{-104}	2^{-108}	2^{-110}	2^{-115}	2^{-117}	2^{-121}
Cor_{Opt}	2^{-31}	2^{-33}	2^{-34}	2^{-36}	2^{-37}	2^{-39}	2^{-40}	2^{-43}	2^{-45}	2^{-48}	2^{-49}	2^{-52}	2^{-53}	2^{-55}	2^{-56}	2^{-59}	2^{-61}	2^{-63}

SIMON64																		
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Pr_{Opt}	1	2^{-2}	2^{-4}	2^{-6}	2^{-8}	2^{-12}	2^{-14}	2^{-18}	2^{-20}	2^{-26}	2^{-30}	2^{-36}	2^{-38}	2^{-44}	2^{-48}	2^{-54}	2^{-56}	2^{-62}
Cor_{Opt}	1	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-6}	2^{-7}	2^{-9}	2^{-10}	2^{-13}	2^{-15}	2^{-18}	2^{-19}	2^{-22}	2^{-24}	2^{-27}	2^{-28}	2^{-31}
Round	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Pr_{Opt}	2^{-64}	2^{-66}	2^{-68}	2^{-72}	2^{-74}	2^{-78}	2^{-80}	2^{-86}	2^{-90}	2^{-96}	2^{-98}	2^{-104}	2^{-108}	2^{-114}	2^{-116}	2^{-122}	2^{-124}	2^{-126}
Cor_{Opt}	2^{-32}	2^{-33}	2^{-34}	2^{-36}	2^{-37}	2^{-39}	2^{-40}	2^{-43}	2^{-45}	2^{-48}	2^{-49}	2^{-52}	2^{-54}	2^{-57}	2^{-58}	2^{-61}	2^{-62}	2^{-63}
Round	37	38	39	40	41	42	43	44										
Pr_{Opt}	2^{-128}	2^{-132}	2^{-134}	2^{-138}	2^{-140}	2^{-146}	2^{-150}	2^{-156}										
Cor_{Opt}	2^{-64}	2^{-66}	2^{-67}	2^{-69}	2^{-70}	2^{-73}	2^{-75}	2^{-78}										

SIMON96																		
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Pr_{Opt}	1	2^{-2}	2^{-4}	2^{-6}	2^{-8}	2^{-12}	2^{-14}	2^{-18}	2^{-20}	2^{-26}	2^{-30}	2^{-36}	2^{-38}	2^{-44}	2^{-48}	2^{-54}	2^{-56}	2^{-62}
Cor_{Opt}	1	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-6}	2^{-7}	2^{-9}	2^{-10}	2^{-13}	2^{-15}	2^{-18}	2^{-19}	2^{-22}	2^{-24}	2^{-27}	2^{-28}	2^{-31}
Round	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Pr_{Opt}	2^{-64}	2^{-66}	2^{-68}	2^{-72}	2^{-74}	2^{-78}	2^{-80}	2^{-86}	2^{-90}	2^{-96}	2^{-98}	2^{-104}	2^{-108}	2^{-114}	2^{-116}	2^{-122}	2^{-124}	2^{-126}
Cor_{Opt}	2^{-32}	2^{-33}	2^{-34}	2^{-36}	2^{-37}	2^{-39}	2^{-40}	2^{-43}	2^{-45}	2^{-48}	2^{-49}	2^{-52}	2^{-54}	2^{-57}	2^{-58}	2^{-61}	2^{-62}	2^{-63}
Round	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
Pr_{Opt}	2^{-128}	2^{-132}	2^{-134}	2^{-138}	2^{-140}	2^{-146}	2^{-150}	2^{-156}	2^{-158}	2^{-164}	2^{-168}	2^{-174}	2^{-176}	2^{-182}	2^{-184}	2^{-186}	2^{-188}	2^{-192}
Cor_{Opt}	2^{-64}	2^{-66}	2^{-67}	2^{-69}	2^{-70}	2^{-73}	2^{-75}	2^{-78}	2^{-79}	2^{-82}	2^{-84}	2^{-87}	2^{-88}	2^{-91}	2^{-92}	2^{-93}	2^{-94}	2^{-96}

SIMON128																		
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Pr_{Opt}	1	2^{-2}	2^{-4}	2^{-6}	2^{-8}	2^{-12}	2^{-14}	2^{-18}	2^{-20}	2^{-26}	2^{-30}	2^{-36}	2^{-38}	2^{-44}	2^{-48}	2^{-54}	2^{-56}	2^{-62}
Cor_{Opt}	1	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-6}	2^{-7}	2^{-9}	2^{-10}	2^{-13}	2^{-15}	2^{-18}	2^{-19}	2^{-22}	2^{-24}	2^{-27}	2^{-28}	2^{-31}
Round	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
Pr_{Opt}	2^{-64}	2^{-66}	2^{-68}	2^{-72}	2^{-74}	2^{-78}	2^{-80}	2^{-86}	2^{-90}	2^{-96}	2^{-98}	2^{-104}	2^{-108}	2^{-114}	2^{-116}	2^{-122}	2^{-124}	2^{-126}
Cor_{Opt}	2^{-32}	2^{-33}	2^{-34}	2^{-36}	2^{-37}	2^{-39}	2^{-40}	2^{-43}	2^{-45}	2^{-48}	2^{-49}	2^{-52}	2^{-54}	2^{-57}	2^{-58}	2^{-61}	2^{-62}	2^{-63}
Round	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
Pr_{Opt}	2^{-128}	2^{-132}	2^{-134}	2^{-138}	2^{-140}	2^{-146}	2^{-150}	2^{-156}	2^{-158}	2^{-164}	2^{-168}	2^{-174}	2^{-176}	2^{-182}	2^{-184}	2^{-186}	2^{-188}	2^{-192}
Cor_{Opt}	2^{-64}	2^{-66}	2^{-67}	2^{-69}	2^{-70}	2^{-73}	2^{-75}	2^{-78}	2^{-79}	2^{-82}	2^{-84}	2^{-87}	2^{-88}	2^{-91}	2^{-92}	2^{-93}	2^{-94}	2^{-96}
Round	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72
Pr_{Opt}	2^{-194}	2^{-198}	2^{-200}	2^{-206}	2^{-210}	2^{-216}	2^{-218}	2^{-224}	2^{-228}	2^{-234}	2^{-236}	2^{-242}	2^{-244}	2^{-246}	2^{-248}	2^{-252}	2^{-254}	2^{-258}
Cor_{Opt}	2^{-97}	2^{-99}	2^{-100}	2^{-103}	2^{-105}	2^{-108}	2^{-109}	2^{-112}	2^{-114}	2^{-117}	2^{-118}	2^{-121}	2^{-122}	2^{-123}	2^{-124}	2^{-126}	2^{-127}	2^{-129}

SPECK family of block ciphers. Please find in Table 10 for the experimental results.

Table 10: Experimental results of SPECK Family of Block Ciphers.

SPECK32														
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Pr_{Opt}	1	2^{-1}	2^{-3}	2^{-5}	2^{-9}	2^{-13}	2^{-18}	2^{-24}	2^{-30}	2^{-34}	2^{-38}	2^{-42}	2^{-45}	2^{-49}
Cor_{Opt}	1	1	2^{-1}	2^{-3}	2^{-5}	2^{-7}	2^{-9}	2^{-12}	2^{-14}	2^{-17}	2^{-19}	2^{-20}	2^{-22}	2^{-24}
Round	15	16	17	18	19	20	21	22						
Pr_{Opt}	2^{-54}	2^{-58}	2^{-63}	2^{-69}	2^{-74}	2^{-77}	2^{-81}	2^{-85}						
Cor_{Opt}	2^{-26}	2^{-28}	2^{-30}	2^{-34}	2^{-36}	2^{-38}	2^{-40}	2^{-42}						
SPECK48														
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Pr_{Opt}	1	2^{-1}	2^{-3}	2^{-6}	2^{-10}	2^{-14}	2^{-19}	2^{-26}	2^{-33}	2^{-40}	2^{-45}	2^{-49}	2^{-54}	2^{-58}
Cor_{Opt}	1	1	2^{-1}	2^{-3}	2^{-6}	2^{-8}	2^{-12}	2^{-15}	2^{-19}	2^{-22}	2^{-25}	2^{-28}	2^{-30}	2^{-33}
Round	15	16	17	18	19	20	21	22	23					
Pr_{Opt}	2^{-63}	2^{-68}	2^{-75}	2^{-82}	-	-	-	-	-					
Cor_{Opt}	2^{-37}	2^{-39}	2^{-43}	2^{-45}	2^{-48}	2^{-51}	2^{-54}	2^{-57}	2^{-59}					
SPECK64														
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Pr_{Opt}	1	2^{-1}	2^{-3}	2^{-6}	2^{-10}	2^{-15}	2^{-21}	2^{-29}	2^{-34}	2^{-38}	2^{-42}	2^{-46}	2^{-50}	2^{-56}
Cor_{Opt}	1	1	2^{-1}	2^{-3}	2^{-6}	2^{-9}	2^{-13}	2^{-17}	2^{-19}	2^{-21}	2^{-24}	2^{-27}	2^{-30}	2^{-33}
Round	15	16	17	18	19	20	21	22	23	24	25	26	27	
Pr_{Opt}	2^{-62}	2^{-70}	2^{-73}	2^{-76}	2^{-81}	2^{-85}	2^{-89}	2^{-94}	2^{-99}	2^{-107}	2^{-112}	2^{-116}	2^{-121}	
Cor_{Opt}	2^{-37}	2^{-41}	2^{-43}	2^{-45}	2^{-47}	2^{-49}	2^{-52}	2^{-54}	2^{-59}	2^{-63}	2^{-66}	2^{-68}	2^{-70}	
SPECK96														
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Pr_{Opt}	1	2^{-1}	2^{-3}	2^{-6}	2^{-10}	2^{-15}	2^{-21}	2^{-30}	2^{-39}	2^{-49}	-	-	-	-
Cor_{Opt}	1	1	2^{-1}	2^{-3}	2^{-6}	2^{-9}	2^{-13}	2^{-18}	2^{-22}	2^{-27}	2^{-31}	2^{-33}	2^{-36}	2^{-39}
SPECK128														
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Pr_{Opt}	1	2^{-1}	2^{-3}	2^{-6}	2^{-10}	2^{-15}	2^{-21}	2^{-30}	2^{-39}	-	-	-	-	-
Cor_{Opt}	1	1	2^{-1}	2^{-3}	2^{-6}	2^{-9}	2^{-13}	2^{-18}	2^{-22}	2^{-27}	-	-	-	-

D Auxiliary Materials for the 26-Round Attack on GIFT-64

D.1 18-Round Related-Key Differential Trail

The 18-round related-key differential trail is shown in Figure 8. The 128-bit master key of this trail is 0x0000 0x0000 0x0000 0x0000 0x0000 0x0000 0x0028 0x0000.

D.2 A Brief Introduction of GIFT-64

To clearly explain the key-recovery attack, we give a brief introduction of GIFT-64.

- **Key schedule and round constants.** Denote $K = k_0\|k_1\|\cdots\|k_7$ the 128-bit master key. After extracting two 16-bit words of the key state as the round key $RK = U\|V = k_6\|k_7$, the key state is updated as follows,

$$k_0\|k_1\|\cdots\|k_7 \leftarrow (k_6 \ggg 2)\|(k_7 \ggg 12)\|k_0\|\cdots\|k_4\|k_5.$$

  Since the values of the round constants do not affect the key-recovery attack, the generating method is not covered here. We refer readers to look up the document [BPP+17] for more details.

Each round of GIFT-64 includes the following three steps.

- **SubCells.** The 4-bit S-box GS is applied to every nibble of the 64-bit cipher state $s_0\|s_1\|\cdots\|s_{63}$.
- **PermBits.** It maps bits from bit position i[4] of the cipher state to bit position $P(i)$,

$$P(i) = 63 - \left\{ 4 \cdot \left\lfloor \frac{63-i}{16} \right\rfloor + 16 \cdot \left[3 \cdot \left\lfloor \frac{(63-i) \bmod 16}{4} \right\rfloor + (63-i) \bmod 16 \right] + (63-i) \bmod 4 \right\} \bmod 64.$$

- **AddRoundKey.** This step consists of adding the round key and round constants. A 32-bit round key RK is extracted from the key state and is further partitioned into two 16-bit words as $RK = U\|V = u_0\|\cdots\|u_{15}\|v_0\|\cdots\|v_{15}$. Then, U and V are XORed to $\{s_{4\cdot i+2} \mid 0 \leqslant i \leqslant 15\}$ and $\{s_{4\cdot i+3} \mid 0 \leqslant i \leqslant 15\}$ of the cipher state, respectively. To be specific,

$$s_{4\cdot i+2} \leftarrow s_{4\cdot i+2} \oplus u_i,\ s_{4\cdot i+3} \leftarrow s_{4\cdot i+3} \oplus v_i,\ 0 \leqslant i \leqslant 15.$$

D.3 An Illustration for the Key-Recovery Attack

The key-recovery attack is demonstrated in Figure 9.

D.4 Detailed Computation of Complexity

The detailed analysis of the complexity can be found in Table 11.

E Comprehensive Comparison of the Accelerating Effect

To clearly illustrate the accelerating effect of the new method, we test the runtime in different settings with two SAT solvers CaDiCaL and Cryptominisat. All the tests in this section are implemented on a server with AMD EPYC 7302 16-Core Processor, and each program utilises one processor. The following notations are exploited to distinguish the runtime in different cases.

[4] Note that the bit position is reversed in this paper.

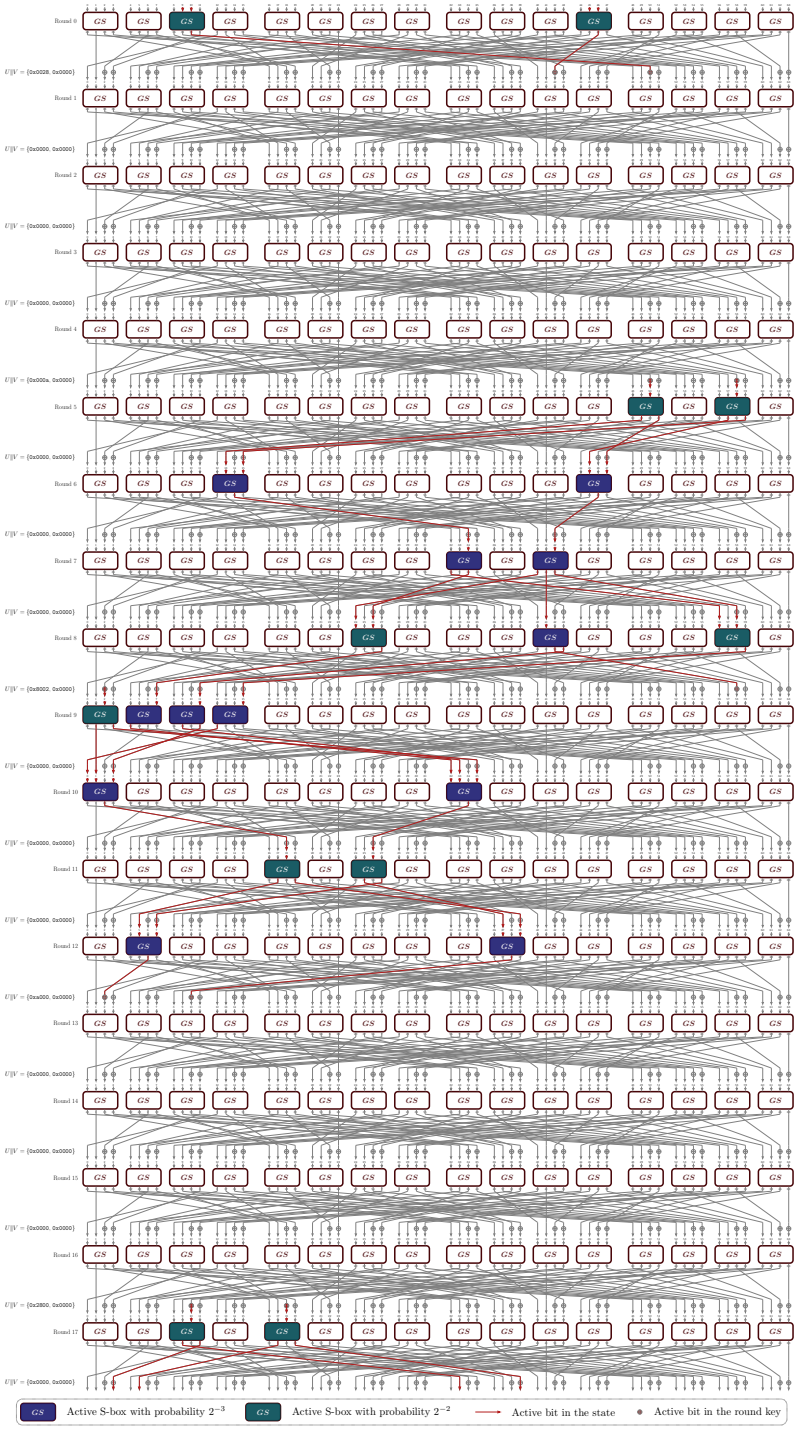


Figure 8: 18-round related-key differential trail of `GIFT-64` with probability 2^{-58}.

- $T^{\emptyset}_{\mathrm{CaD}}$: runtime using CaDiCaL without bounding condition.
- $T^{\mathcal{R}-1}_{\mathrm{CaD}}$: runtime using CaDiCaL with the set $\mathcal{C}_{(*,\mathcal{R}-1)}$.
- T^{0}_{CaD}: runtime using CaDiCaL with the set $\mathcal{C}_{(0,*)}$.

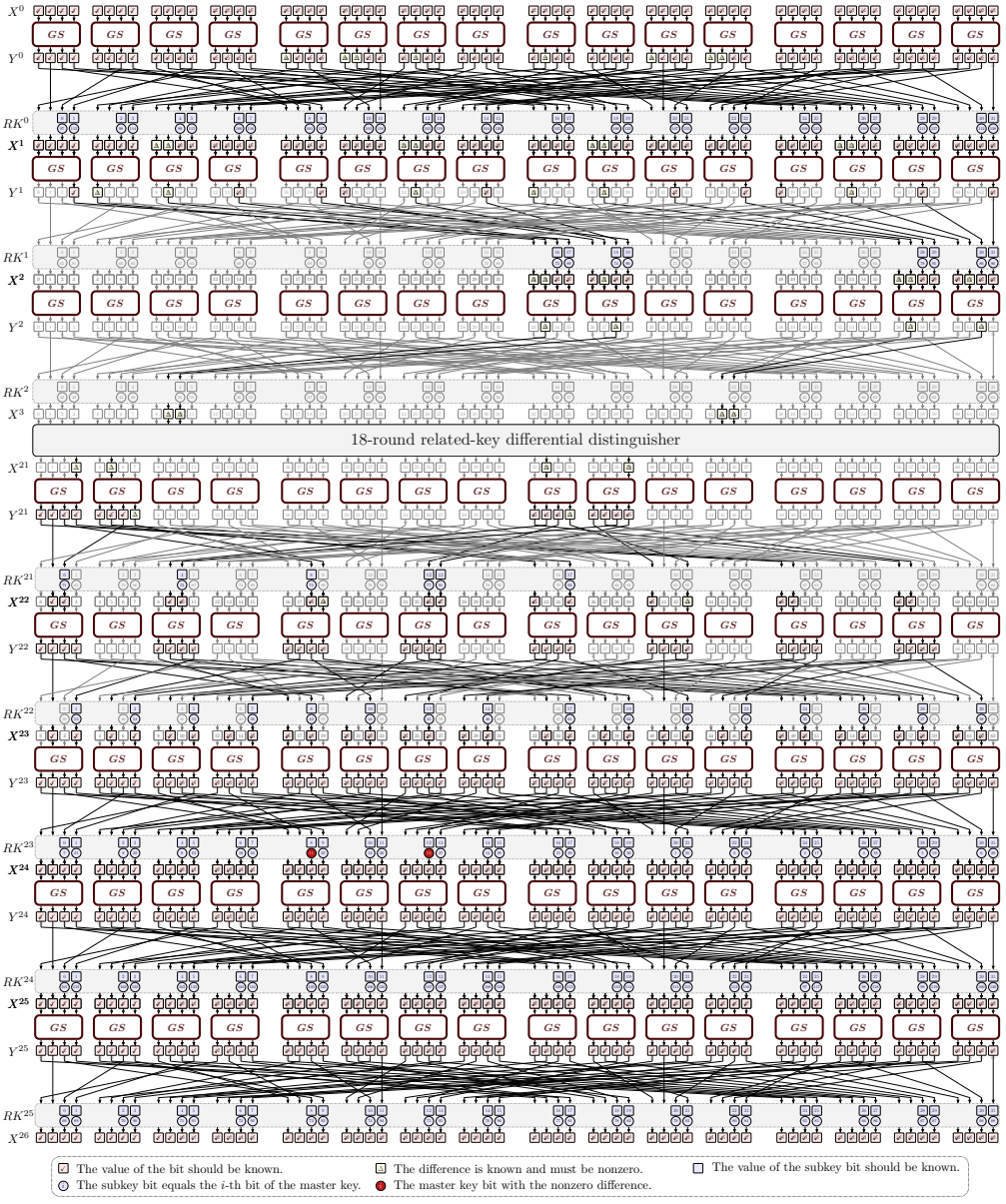


Figure 9: Key-recovery attack on 26-round `GIFT-64`.

- $T^{\emptyset}_{\text{Cry}}$: runtime using Cryptominisat without bounding condition.
- $T^{\mathcal{R}-1}_{\text{Cry}}$: runtime using Cryptominisat with the set $\mathcal{C}_{(*,\mathcal{R}-1)}$.
- T^{0}_{Cry}: runtime using Cryptominisat with the set $\mathcal{C}_{(0,*)}$.

Please find in Table 12 - 21 for the experimental results of `PRESENT`, `GIFT-64`, `RECTANGLE`, `LBlock`, `TWINE` and all versions belonging to `SPECK` family of block ciphers. Note that the values of $T_{\texttt{MILP}}$ for `SPECK32` and `SPECK48` stem from [ZSCH18], where the authors claimed that the tests employed 16 threads of a server with Intel® Xeon® E5-2637V3 CPU 3.50 GHz.

Table 11: Detailed computation of complexity.

Step	Guessed subkey	Condition on the difference	#{Remaining pairs}	Time complexity (GS operations)
1	$RK^0[0,1]$	$\Delta Y^1[0] = \Delta Y^1[1] = \Delta Y^1[2] = 0$	$N_1 \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^2$
2	$RK^0[2,3]$	$\overline{\Delta Y^1[4]} = \Delta Y^1[5] = \Delta Y^1[6] = \Delta Y^1[7] = 0$	$N_1 \cdot 2^{-3} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-3} \cdot 2^2 \cdot 2^2$
3	$RK^0[4,5]$	$\Delta Y^1[8] = \overline{\Delta Y^1[9]} = \Delta Y^1[10] = \Delta Y^1[11] = 0$	$N_1 \cdot 2^{-7} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-7} \cdot 2^4 \cdot 2^2$
4	$RK^0[6,7]$	$\Delta Y^1[12] = \Delta Y^1[13] = \Delta Y^1[15] = 0$	$N_1 \cdot 2^{-9} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-9} \cdot 2^6 \cdot 2^2$
5	$RK^0[8,9]$	$\Delta Y^1[16] = \Delta Y^1[17] = \Delta Y^1[18] = 0$	$N_1 \cdot 2^{-12} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-12} \cdot 2^8 \cdot 2^2$
6	$RK^0[10,11]$	$\Delta Y^1[21] = \Delta Y^1[22] = \Delta Y^1[23] = 0$	$N_1 \cdot 2^{-15} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-15} \cdot 2^{10} \cdot 2^2$
7	$RK^0[12,13]$	$\Delta Y^1[24] = \overline{\Delta Y^1[25]} = \Delta Y^1[26] = \Delta Y^1[27] = 0$	$N_1 \cdot 2^{-18} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-18} \cdot 2^{12} \cdot 2^2$
8	$RK^0[14,15]$	$\Delta Y^1[28] = \Delta Y^1[29] = \Delta Y^1[31] = 0$	$N_1 \cdot 2^{-20} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-20} \cdot 2^{14} \cdot 2^2$
9	$RK^0[16,17]$	$\overline{\Delta Y^1[32]} = \Delta Y^1[33] = \Delta Y^1[34] = \Delta Y^1[35] = 0$	$N_1 \cdot 2^{-23} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-23} \cdot 2^{16} \cdot 2^2$
10	$RK^0[18,19]$	$\Delta Y^1[36] = \overline{\Delta Y^1[37]} = \Delta Y^1[38] = \Delta Y^1[39] = 0$	$N_1 \cdot 2^{-27} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-27} \cdot 2^{18} \cdot 2^2$
11	$RK^0[20,21]$	$\Delta Y^1[40] = \Delta Y^1[41] = \Delta Y^1[43] = 0$	$N_1 \cdot 2^{-29} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-29} \cdot 2^{20} \cdot 2^2$
12	$RK^0[22,23]$	$\Delta Y^1[44] = \Delta Y^1[45] = \Delta Y^1[46] = 0$	$N_1 \cdot 2^{-32} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-32} \cdot 2^{22} \cdot 2^2$
13	$RK^0[24,25]$	$\Delta Y^1[49] = \Delta Y^1[50] = \Delta Y^1[51] = 0$	$N_1 \cdot 2^{-35} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-35} \cdot 2^{24} \cdot 2^2$
14	$RK^0[26,27]$	$\Delta Y^1[52] = \overline{\Delta Y^1[53]} = \Delta Y^1[54] = \Delta Y^1[55] = 0$	$N_1 \cdot 2^{-38} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-38} \cdot 2^{26} \cdot 2^2$
15	$RK^0[28,29]$	$\Delta Y^1[56] = \Delta Y^1[57] = \Delta Y^1[59] = 0$	$N_1 \cdot 2^{-40} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-40} \cdot 2^{28} \cdot 2^2$
16	$RK^0[30,31]$	$\Delta Y^1[60] = \Delta Y^1[61] = \Delta Y^1[62] = 0$	$N_1 \cdot 2^{-43} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-43} \cdot 2^{30} \cdot 2^2$
17	$RK^1[16,17]$	$\Delta Y^2[32] = \overline{\Delta Y^2[33]} = \Delta Y^2[34] = \Delta Y^2[35] = 0$	$N_1 \cdot 2^{-46} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-46} \cdot 2^{32} \cdot 2^2$
18	$RK^1[18,19]$	$\Delta Y^2[36] = \Delta Y^2[37] = \overline{\Delta Y^2[38]} = \Delta Y^2[39] = 0$	$N_1 \cdot 2^{-48} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-48} \cdot 2^{34} \cdot 2^2$
19	$RK^1[28,29]$	$\Delta Y^2[56] = \overline{\Delta Y^2[57]} = \Delta Y^2[58] = \Delta Y^2[59] = 0$	$N_1 \cdot 2^{-51} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-51} \cdot 2^{36} \cdot 2^2$
20	$RK^1[30,31]$	$\Delta Y^2[60] = \Delta Y^2[61] = \overline{\Delta Y^2[62]} = \Delta Y^2[63] = 0$	$N_1 \cdot 2^{-53} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-53} \cdot 2^{38} \cdot 2^2$
21	24 bits of RK^{25}	-	$N_1 \cdot 2^{-56}$	$2 \cdot N_1 \cdot 2^{-56} \cdot 2^{40} \cdot 2^{24} \cdot 32$
22	$RK^{23}[8,17]$	$\Delta X^{23}[0] = \Delta X^{23}[2] = 0$	$N_1 \cdot 2^{-56} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-56} \cdot 2^{64} \cdot 2^2$
23	$RK^{23}[0,9]$	$\Delta X^{23}[4] = \Delta X^{23}[6] = 0$	$N_1 \cdot 2^{-58} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-58} \cdot 2^{66} \cdot 2^2$
24	$RK^{23}[1,24]$	$\Delta X^{23}[8] = \Delta X^{23}[10] = 0$	$N_1 \cdot 2^{-60} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-60} \cdot 2^{68} \cdot 2^2$
25	$RK^{23}[16,25]$	$\Delta X^{23}[12] = \Delta X^{23}[14] = 0$	$N_1 \cdot 2^{-62} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-62} \cdot 2^{70} \cdot 2^2$
26	$RK^{23}[10,19]$	$\Delta X^{23}[17] = \Delta X^{23}[19] = 0$	$N_1 \cdot 2^{-64} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-64} \cdot 2^{72} \cdot 2^2$
27	$RK^{23}[2,11]$	$\Delta X^{23}[21] = \Delta X^{23}[23] = 0$	$N_1 \cdot 2^{-66} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-66} \cdot 2^{74} \cdot 2^2$
28	$RK^{23}[3,26]$	$\Delta X^{23}[25] = \Delta X^{23}[27] = 0$	$N_1 \cdot 2^{-68} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-68} \cdot 2^{76} \cdot 2^2$
29	$RK^{23}[18,27]$	$\Delta X^{23}[29] = \Delta X^{23}[31] = 0$	$N_1 \cdot 2^{-70} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-70} \cdot 2^{78} \cdot 2^2$
30	$RK^{23}[12,21]$	$\Delta X^{23}[32] = \Delta X^{23}[34] = 0$	$N_1 \cdot 2^{-72} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-72} \cdot 2^{80} \cdot 2^2$
31	$RK^{23}[4,13]$	$\Delta X^{23}[36] = \Delta X^{23}[38] = 0$	$N_1 \cdot 2^{-74} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-74} \cdot 2^{82} \cdot 2^2$
32	$RK^{23}[5,28]$	$\Delta X^{23}[40] = \Delta X^{23}[42] = 0$	$N_1 \cdot 2^{-76} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-76} \cdot 2^{84} \cdot 2^2$
33	$RK^{23}[20,29]$	$\Delta X^{23}[44] = \Delta X^{23}[46] = 0$	$N_1 \cdot 2^{-78} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-78} \cdot 2^{86} \cdot 2^2$
34	$RK^{23}[14,23]$	$\Delta X^{23}[49] = \Delta X^{23}[51] = 0$	$N_1 \cdot 2^{-80} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-80} \cdot 2^{88} \cdot 2^2$
35	$RK^{23}[6,15]$	$\Delta X^{23}[53] = \Delta X^{23}[55] = 0$	$N_1 \cdot 2^{-82} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-82} \cdot 2^{90} \cdot 2^2$
36	$RK^{23}[7,30]$	$\Delta X^{23}[57] = \Delta X^{23}[59] = 0$	$N_1 \cdot 2^{-84} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-84} \cdot 2^{92} \cdot 2^2$
37	$RK^{23}[22,31]$	$\Delta X^{23}[61] = \Delta X^{23}[63] = 0$	$N_1 \cdot 2^{-86} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-86} \cdot 2^{94} \cdot 2^2$
38	$RK^{22}[8,17]$	$\Delta X^{22}[0] = \Delta X^{22}[3] = 0$	$N_1 \cdot 2^{-88} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-88} \cdot 2^{96} \cdot 2^2$
39	$RK^{22}[1,24]$	$\Delta X^{22}[8] = \Delta X^{22}[11] = 0$	$N_1 \cdot 2^{-90} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-90} \cdot 2^{98} \cdot 2^2$
40	$RK^{22}[10,19]$	$\Delta X^{22}[16] = \Delta X^{22}[17] = \overline{\Delta X^{22}[19]} = 0$	$N_1 \cdot 2^{-92} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-92} \cdot 2^{100} \cdot 2^2$
41	$RK^{22}[3,26]$	$\Delta X^{22}[24] = \Delta X^{22}[25] = 0$	$N_1 \cdot 2^{-95} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-95} \cdot 2^{102} \cdot 2^2$
42	$RK^{22}[12,21]$	$\Delta X^{22}[33] = \Delta X^{22}[34] = 0$	$N_1 \cdot 2^{-97} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-97} \cdot 2^{104} \cdot 2^2$
43	$RK^{22}[5,28]$	$\Delta X^{22}[41] = \Delta X^{22}[42] = \overline{\Delta X^{22}[43]} = 0$	$N_1 \cdot 2^{-99} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-99} \cdot 2^{106} \cdot 2^2$
44	$RK^{22}[14,23]$	$\Delta X^{22}[50] = \Delta X^{22}[51] = 0$	$N_1 \cdot 2^{-102} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-102} \cdot 2^{108} \cdot 2^2$
45	$RK^{22}[7,30]$	$\Delta X^{22}[58] = \Delta X^{22}[59] = 0$	$N_1 \cdot 2^{-104} \cdot 2^{-2}$	$2 \cdot N_1 \cdot 2^{-104} \cdot 2^{110} \cdot 2^2$
46	-	$\Delta X^{21}[0] = \Delta X^{21}[1] = \Delta X^{21}[2] = \overline{\Delta X^{21}[3]} = 0$	$N_1 \cdot 2^{-106} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-106} \cdot 2^{112}$
47	-	$\Delta X^{21}[4] = \overline{\Delta X^{21}[5]} = \Delta X^{21}[6] = \Delta X^{21}[7] = 0$	$N_1 \cdot 2^{-110} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-110} \cdot 2^{112}$
48	-	$\Delta X^{21}[32] = \overline{\Delta X^{21}[33]} = \Delta X^{21}[34] = \Delta X^{21}[35] = 0$	$N_1 \cdot 2^{-113} \cdot 2^{-3}$	$2 \cdot N_1 \cdot 2^{-113} \cdot 2^{112}$
49	-	$\Delta X^{21}[36] = \Delta X^{21}[37] = \Delta X^{21}[38] = \overline{\Delta X^{21}[39]} = 0$	$N_1 \cdot 2^{-116} \cdot 2^{-4}$	$2 \cdot N_1 \cdot 2^{-116} \cdot 2^{112}$
Total	-	-	-	$N_1 \cdot 2^{15.90}$

In the following, we list some observations to assist readers in understanding the main reason for the acceleration.

1. Note that the difference between the values of $T^{\emptyset}_{\text{CaD}}$ and $T^{\emptyset}_{\text{Cry}}$ exhibits the gain resulted from applying a different solver. It can be seen from Table 12 - 21 that changing the solver is not the crucial reason for the acceleration.

2. The comparison between the value of $T^{\emptyset}_{\text{CaD}}$ and the value of $T^{\mathcal{R}-1}_{\text{CaD}}$ or T^{0}_{CaD} indicates the gain of the new encoding method. Therefore, we confirm that the significant improvement on the runtime mainly benefits from the new encoding approach. The results in Table 12 - 21 evidence that the strategies proposed in Sect. 4 can be generally applied to various block ciphers concerning different searching tasks, even though the idea is demonstrated with the tests on `GIFT-64`. Also, the comparison between the value of $T^{\emptyset}_{\text{Cry}}$ and the value of $T^{\mathcal{R}-1}_{\text{Cry}}$ or T^{0}_{Cry} reveals that the new encoding method also works for the solver Cryptominisat.

3. With the experimental results for all versions of `SPECK` family of block ciphers in Table 17 - 21, we note that the acceleration is not significant. As we mentioned in Sect. 5.3, adding bounding conditions regarding the test for `SPECK` cannot significantly improve the automatic search with the SAT method. This circumstance coincides with the observation raised by Zhang et al. [ZSCH18].

4. Another interesting observation is that for problems that are not time-consuming, e.g., targeting the minimum number of active S-boxes, CaDiCaL does not show significant advantages. However, when it comes to more challenging tasks for the optimal trails with the maximum differential probability and linear bias, CaDiCaL dramatically reduces the runtime. This observation may help readers to select the SAT solver according to their customised searching problems.

Table 12: Experimental results of PRESENT.

Differential property																
Round	$\#\mathtt{S}_\mathtt{D}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.1s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
2	2	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	1s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	2s
3	4	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	2s	2^{-8}	0.2s	0.2s	0.2s	0.2s	0.2s	0.3s	3s
4	6	0.1s	0.1s	0.2s	0.1s	0.1s	0.2s	4s	2^{-12}	0.5s	0.3s	0.4s	0.4s	0.3s	0.4s	4s
5	10	0.8s	0.7s	0.9s	0.8s	0.7s	0.9s	5s	2^{-20}	4.4s	3.4s	4.0s	4.1s	3.9s	4.3s	5s
6	12	0.7s	0.4s	0.4s	0.8s	0.2s	0.5s	8s	2^{-24}	5.6s	2.0s	2.4s	6.5s	2.2s	1.8s	249s
7	14	1.2s	0.3s	0.5s	1.7s	0.2s	1.0s	10s	2^{-28}	12.0s	2.6s	2.4s	13.6s	6.0s	5.7s	9s
8	16	1.9s	0.4s	0.4s	2.7s	0.2s	0.5s	11s	2^{-32}	15.9s	2.1s	2.6s	29.5s	5.0s	2.4s	11s
9	18	3.0s	0.3s	0.2s	5.2s	0.2s	0.2s	15s	2^{-36}	34.3s	2.1s	3.0s	40.6s	5.0s	1.4s	14s
10	20	3.9s	0.3s	0.3s	7.2s	0.2s	0.3s	16s	2^{-41}	54.1s	3.2s	4.1s	87.5s	12.1s	4.2s	1298s
11	22	7.7s	0.2s	0.3s	11.0s	0.2s	0.3s	18s	2^{-46}	118.8s	5.6s	8.0s	168.1s	11.8s	9.1s	438s
12	24	10.9s	0.3s	0.3s	19.7s	0.2s	0.3s	22s	2^{-52}	275.5s	10.1s	21.7s	454.8s	27.9s	48.8s	311s
13	26	12.7s	0.3s	0.4s	21.3s	0.2s	0.3s	24s	2^{-56}	230.2s	5.3s	6.1s	467.9s	17.0s	19.1s	22s
14	28	19.2s	0.3s	0.4s	21.3s	0.2s	0.4s	31s	2^{-62}	520.9s	19.0s	42.4s	954.7s	45.7s	128.4s	18859s
15	30	21.7s	0.4s	0.6s	24.4s	0.2s	0.4s	32s	2^{-66}	487.5s	9.9s	6.9s	884.3s	25.6s	10.2s	2594s
16	32	34.6s	0.5s	0.9s	49.2s	0.3s	0.4s	19s	2^{-70}	550.9s	4.8s	12.5s	1095.7s	24.6s	12.1s	370s
17	34	25.0s	0.5s	0.5s	53.0s	0.3s	0.4s	20s	2^{-74}	671.7s	5.1s	8.6s	1520.4s	26.3s	14.2s	20s
18	36	39.4s	0.6s	0.5s	71.4s	0.3s	0.5s	22s	2^{-78}	880.3s	7.0s	8.4s	1901.5s	16.2s	9.6s	629s
19	38	38.7s	0.5s	0.5s	58.2s	0.4s	0.5s	34s	2^{-82}	1027.5s	5.5s	7.6s	2358.9s	6.9s	6.7s	-
20	40	45.6s	0.7s	0.6s	92.9s	0.3s	0.5s	29s	2^{-86}	1304.6s	5.2s	10.4s	2748.6s	14.6s	12.1s	-
21	42	57.1s	0.7s	0.5s	61.6s	0.4s	0.5s	28s	2^{-90}	1308.9s	5.6s	11.1s	3576.9s	12.3s	11.2s	-
22	44	89.2s	0.8s	0.6s	97.7s	0.5s	0.5s	29s	2^{-96}	2219.2s	15.7s	33.0s	6943.5s	31.5s	81.5s	-
23	46	93.5s	0.9s	0.8s	100.8s	0.4s	0.6s	37s	2^{-100}	1501.8s	10.6s	13.5s	5530.6s	13.8s	9.6s	-
24	48	95.3s	1.4s	0.7s	126.4s	0.4s	0.6s	34s	2^{-106}	3077.1s	25.9s	52.7s	8891.3s	54.0s	104.9s	-
25	50	122.3s	1.3s	0.8s	102.3s	0.4s	0.7s	36s	2^{-110}	2421.6s	12.5s	26.7s	7672.8s	38.9s	156.8s	-
26	52	97.1s	0.9s	0.8s	132.7s	0.5s	0.6s	38s	2^{-116}	4307.1s	42.7s	74.4s	3.6h	125.1s	303.9s	-
27	54	124.9s	1.0s	1.2s	163.9s	0.5s	0.7s	40s	2^{-120}	3073.1s	15.4s	20.0s	2.2h	56.3s	27.6s	-
28	56	137.2s	1.0s	0.9s	171.3s	0.8s	0.8s	42s	2^{-124}	3499.7s	11.4s	42.5s	4.2h	46.3s	33.0s	-
29	58	126.3s	1.0s	0.8s	148.3s	0.7s	0.8s	42s	2^{-128}	3316.8s	12.2s	27.1s	4.5h	38.9s	34.3s	-
30	60	136.0s	1.5s	0.9s	198.3s	0.6s	0.7s	44s	2^{-132}	3618.3s	14.8s	22.7s	4.4h	33.0s	24.5s	-
31	62	200.9s	1.4s	1.6s	171.2s	0.7s	0.8s	47s	2^{-136}	3826.7s	12.6s	18.4s	4.9h	22.2s	24.3s	-
Total		1546.9s	19.1s	17.9s	1915.6s	10.5s	15.5s	740s		10.7h	273.1s	494.2s	36.4h	723.4s	1102.5s	6.9h
Linear property																
Round	$\#\mathtt{S}_\mathtt{L}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-1}	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s
2	2	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	2s	2^{-2}	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	2s
3	3	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	3s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	71s
4	4	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	6s	2^{-6}	0.1s	0.1s	0.2s	0.1s	0.1s	0.1s	88s
5	5	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	9s	2^{-8}	0.3s	0.3s	0.2s	0.2s	0.2s	0.2s	152s
6	6	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	8s	2^{-10}	0.5s	0.4s	0.3s	0.4s	0.4s	0.3s	128s
7	7	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	7s	2^{-12}	1.3s	0.7s	0.7s	1.1s	0.4s	0.4s	18s
8	8	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	8s	2^{-14}	2.1s	1.1s	0.6s	2.1s	0.8s	0.6s	98s
9	9	0.0s	0.1s	0.1s	0.1s	0.1s	0.1s	10s	2^{-16}	2.8s	1.1s	0.8s	2.6s	1.1s	0.8s	15s
10	10	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	11s	2^{-18}	4.7s	1.5s	1.3s	6.1s	1.2s	1.3s	300s
11	11	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	12s	2^{-20}	5.3s	1.2s	1.2s	7.0s	2.0s	1.9s	11s
12	12	0.1s	0.1s	0.1s	0.2s	0.1s	0.1s	14s	2^{-22}	9.4s	1.2s	1.3s	11.4s	2.2s	1.9s	978s
13	13	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	15s	2^{-24}	12.7s	1.7s	1.4s	15.5s	3.1s	2.9s	14s
14	14	0.1s	0.1s	0.1s	0.2s	0.1s	0.1s	17s	2^{-26}	11.5s	2.0s	1.7s	22.9s	3.2s	4.7s	3507s
15	15	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	19s	2^{-28}	24.1s	2.8s	2.5s	24.4s	4.7s	5.6s	16s
16	16	0.1s	0.1s	0.1s	0.2s	0.1s	0.1s	21s	2^{-30}	28.3s	2.9s	2.5s	45.0s	4.1s	6.1s	3080s
17	17	0.1s	0.1s	0.1s	0.3s	0.1s	0.1s	23s	2^{-32}	30.9s	3.9s	3.4s	50.6s	4.5s	6.6s	16302s
18	18	0.1s	0.1s	0.1s	0.3s	0.1s	0.1s	24s	2^{-34}	31.1s	5.1s	4.7s	67.9s	6.8s	7.8s	14105s
19	19	0.1s	0.1s	0.1s	0.1s	0.1s	0.2s	26s	2^{-36}	31.5s	4.4s	4.2s	94.4s	12.5s	9.0s	-
20	20	0.1s	0.1s	0.1s	0.2s	0.1s	0.2s	28s	2^{-38}	48.0s	6.0s	6.4s	106.9s	11.5s	8.3s	-
21	21	0.1s	0.1s	0.1s	0.3s	0.1s	0.2s	30s	2^{-40}	74.1s	4.7s	4.0s	148.9s	24.4s	12.3s	-
22	22	0.1s	0.1s	0.1s	0.4s	0.1s	0.2s	34s	2^{-42}	55.2s	6.2s	6.4s	153.7s	18.6s	24.7s	-
23	23	0.1s	0.1s	0.2s	0.5s	0.1s	0.2s	35s	2^{-44}	88.3s	6.7s	5.2s	213.3s	39.7s	31.5s	-
24	24	0.1s	0.1s	0.2s	0.5s	0.1s	0.2s	37s	2^{-46}	79.3s	11.3s	6.6s	236.9s	58.8s	24.9s	-
25	25	0.1s	0.1s	0.2s	0.6s	0.1s	0.2s	40s	2^{-48}	109.9s	11.6s	13.0s	273.4s	43.5s	64.2s	-
26	26	0.1s	0.1s	0.2s	0.2s	0.1s	0.2s	42s	2^{-50}	109.2s	11.9s	11.5s	182.2s	41.6s	46.0s	-
27	27	0.1s	0.1s	0.2s	0.6s	0.2s	0.2s	44s	2^{-52}	125.8s	12.4s	12.5s	267.5s	52.0s	65.5s	-
28	28	0.1s	0.1s	0.2s	0.6s	0.2s	0.3s	46s	2^{-54}	140.3s	14.1s	10.4s	278.3s	28.8s	64.1s	-
29	29	0.1s	0.2s	0.2s	0.7s	0.2s	0.3s	49s	2^{-56}	144.3s	13.8s	21.7s	363.0s	49.4s	90.8s	-
30	30	0.1s	0.2s	0.2s	0.8s	0.2s	0.4s	49s	2^{-58}	168.7s	11.8s	11.0s	511.7s	80.6s	91.5s	-
31	31	0.2s	0.2s	0.2s	0.9s	0.2s	0.3s	51s	2^{-60}	163.4s	22.0s	11.5s	459.7s	78.9s	70.8s	-
Total		2.5s	2.6s	3.8s	8.5s	3.1s	4.4s	720s		1503.4s	163.0s	147.6s	3547.4s	575.5s	645.1s	10.8h

Table 13: Experimental results of `GIFT-64`.

Differential property																	
Round	$\#S_D$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}-1}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}-1}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Pr_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}-1}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}-1}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	1	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	$2^{-1.415}$	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
2	2	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	2s	$2^{-3.415}$	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	47s
3	3	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	3s	2^{-7}	0.2s	0.2s	0.2s	0.3s	0.3s	0.3s	108s
4	5	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	69s	$2^{-11.415}$	0.9s	0.6s	0.5s	0.7s	0.5s	0.5s	291s
5	7	0.2s	0.1s	0.1s	0.1s	0.1s	0.2s	61s	2^{-17}	4.0s	4.3s	3.9s	3.8s	3.3s	4.4s	849s
6	10	0.7s	0.7s	0.4s	0.5s	0.4s	0.4s	144s	$2^{-22.415}$	15.6s	5.2s	9.8s	19.6s	8.5s	14.1s	181s
7	13	1.8s	1.9s	1.3s	2.1s	1.3s	1.5s	115s	$2^{-28.415}$	69.8s	16.6s	40.6s	77.4s	25.7s	66.9s	385s
8	16	6.0s	3.1s	4.1s	9.4s	3.3s	10.5s	271s	2^{-38}	867.0s	591.5s	676.0s	1254.7s	577.8s	773.2s	19934s
9	18	6.5s	1.8s	2.7s	8.0s	0.5s	2.8s	28s	2^{-42}	689.9s	30.0s	122.8s	1064.1s	77.3s	80.0s	32s
10	20	9.6s	1.1s	1.6s	19.4s	1.0s	4.1s	124s	2^{-48}	1560.6s	172.3s	430.2s	2357.1s	200.7s	310.5s	7569s
11	22	17.7s	1.6s	1.5s	24.5s	1.0s	11.6s	77s	2^{-52}	1582.9s	21.5s	51.2s	1634.5s	86.2s	90.3s	121s
12	24	25.8s	1.2s	2.1s	40.5s	0.4s	10.7s	19s	2^{-58}	4189.3s	108.6s	239.1s	5058.3s	136.1s	324.6s	61001s
13	26	34.3s	1.1s	1.0s	42.6s	0.3s	1.9s	75s	2^{-62}	3273.1s	53.3s	55.5s	6321.5s	32.7s	82.6s	604s
14	28	58.7s	0.8s	0.7s	93.8s	0.5s	1.8s	15s	2^{-68}	6411.6s	60.6s	102.0s	4.3h	118.9s	209.5s	9121s
15	30	65.5s	1.0s	2.7s	136.5s	0.3s	1.8s	17s	2^{-72}	4159.1s	50.9s	60.4s	4.2h	41.6s	88.5s	1595s
16	32	74.8s	1.2s	2.3s	243.9s	0.3s	0.5s	18s	2^{-78}	3.2h	80.0s	101.4s	10.3h	85.5s	372.5s	-
17	34	159.5s	1.2s	2.3s	233.7s	0.3s	1.3s	-	2^{-82}	2.3h	34.7s	29.9s	10.3h	34.2s	201.7s	-
18	36	164.9s	1.3s	2.4s	201.7s	0.3s	1.0s	-	2^{-88}	4.9h	78.1s	110.0s	22.3h	281.0s	578.5s	-
19	38	143.7s	1.2s	2.3s	257.1s	0.6s	1.0s	-	2^{-92}	4.5h	24.0s	96.1s	-	103.1s	173.2s	-
20	40	132.4s	1.6s	2.5s	226.2s	0.7s	1.0s	-	2^{-98}	8.5h	105.4s	120.6s	-	158.9s	625.3s	-
21	42	246.1s	1.0s	1.1s	408.3s	0.6s	2.8s	-	2^{-102}	5.3h	56.5s	69.7s	-	215.8s	232.3s	-
22	44	143.7s	1.4s	2.8s	591.5s	0.6s	2.2s	-	2^{-108}	8.8h	140.0s	209.0s	-	244.5s	265.6s	-
23	46	441.1s	2.1s	4.8s	450.2s	0.5s	1.8s	-	2^{-112}	5.7h	77.5s	92.8s	-	68.0s	78.2s	-
24	48	368.2s	1.5s	3.0s	833.8s	0.6s	1.2s	-	2^{-118}	9.5h	223.3s	255.2s	-	444.2s	614.6s	-
25	50	630.8s	1.4s	2.5s	758.4s	0.5s	1.5s	-	2^{-122}	-	47.6s	175.6s	-	104.1s	150.8s	-
26	52	298.6s	1.6s	3.0s	784.3s	0.6s	2.7s	-	2^{-128}	-	143.4s	208.8s	-	576.3s	783.9s	-
27	54	520.4s	1.6s	1.8s	799.0s	1.4s	1.9s	-	2^{-132}	-	104.2s	102.3s	-	357.2s	268.1s	-
28	56	309.2s	2.1s	1.8s	1114.8s	0.9s	2.6s	-	2^{-138}	-	279.2s	212.2s	-	511.2s	1190.7s	-
Total		3860.2s	34.0s	51.2s	7280.6s	17.5s	69.2s	1039s		58.9h	2509.5s	3575.7s	56.3h	4493.8s	7580.9s	28.3h
Linear property																	
Round	$\#S_L$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}-1}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}-1}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Cor_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}-1}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}-1}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	1	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
2	2	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	1s	2^{-2}	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	2s
3	3	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	3s	2^{-3}	0.0s	0.0s	0.1s	0.0s	0.0s	0.0s	3s
4	5	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	61s	2^{-5}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	77s
5	7	0.2s	0.2s	0.2s	0.2s	0.1s	0.1s	60s	2^{-7}	0.2s	0.2s	0.2s	0.1s	0.2s	0.2s	99s
6	9	0.3s	0.5s	0.2s	0.3s	0.2s	0.2s	65s	2^{-10}	1.0s	0.8s	0.7s	0.8s	0.6s	0.5s	160s
7	12	1.4s	1.2s	1.1s	1.4s	1.2s	1.2s	177s	2^{-13}	2.4s	2.0s	1.5s	2.7s	2.8s	1.7s	225s
8	15	3.7s	3.1s	2.6s	5.7s	3.2s	4.2s	243s	2^{-16}	6.4s	6.1s	5.0s	16.1s	10.0s	7.8s	263s
9	18	14.0s	8.8s	6.7s	26.8s	17.5s	14.2s	493s	2^{-20}	40.5s	31.7s	23.3s	48.6s	52.9s	65.4s	8713s
10	20	14.7s	2.6s	2.8s	27.9s	5.4s	10.2s	681s	2^{-25}	395.0s	294.5s	298.9s	454.2s	521.1s	777.6s	11615s
11	22	37.4s	2.6s	8.0s	38.6s	7.2s	12.8s	392s	2^{-29}	1287.0s	1119.7s	1059.9s	1801.8s	1174.8s	1557.2s	34019s
12	24	41.8s	4.6s	2.4s	58.2s	8.3s	7.7s	3206s	2^{-31}	1008.0s	244.1s	55.9s	1141.9s	124.6s	223.9s	14644s
13	26	60.4s	5.4s	2.3s	147.1s	2.2s	7.4s	11229s	2^{-34}	2005.1s	524.7s	1403.5s	3112.7s	1405.7s	382.3s	121716s
14	28	78.7s	2.9s	2.3s	156.8s	2.4s	2.8s	7982s	2^{-37}	4399.5s	935.5s	591.6s	5133.3s	1838.6s	1181.9s	-
15	30	95.0s	1.6s	2.2s	294.9s	1.5s	1.6s	18410s	2^{-40}	4736.3s	768.5s	712.7s	2.9h	2548.4s	1286.6s	-
16	32	112.4s	1.3s	1.4s	340.1s	0.9s	5.3s	-	2^{-43}	6442.4s	2330.9s	573.0s	4.0h	2966.5s	1001.4s	-
17	34	181.5s	1.4s	2.5s	261.9s	1.1s	2.9s	-	2^{-46}	9198.6s	1150.4s	368.8s	5.8h	3260.9s	2077.2s	-
18	36	170.5s	2.0s	3.3s	502.3s	1.1s	34.1s	-	2^{-49}	2.8h	1948.5s	444.5s	10.7h	5711.6s	375.1s	-
19	38	177.3s	1.7s	6.0s	411.2s	1.4s	2.8s	-	2^{-52}	3.4h	521.6s	549.1s	13.9h	4728.6s	532.7s	-
20	40	189.7s	1.6s	2.6s	460.1s	1.7s	2.4s	-	2^{-55}	4.1h	513.8s	328.5s	20.7h	3838.8s	1148.8s	-
21	42	554.7s	2.3s	2.7s	789.4s	1.4s	2.0s	-	2^{-58}	7.6h	423.1s	217.9s	-	1595.1s	586.6s	-
22	44	485.2s	1.7s	7.5s	503.5s	1.5s	3.0s	-	2^{-61}	7.9h	490.5s	264.0s	-	3626.3s	1132.7s	-
23	46	318.3s	4.3s	2.9s	802.4s	2.2s	1.6s	-	2^{-64}	8.7h	1218.2s	341.8s	-	3193.6s	480.8s	-
24	48	428.3s	3.1s	8.0s	538.1s	1.5s	3.5s	-	2^{-67}	10.4h	371.8s	356.6s	-	1468.7s	1126.2s	-
25	50	324.2s	2.3s	3.1s	360.2s	1.5s	2.8s	-	2^{-70}	8.8h	452.8s	472.1s	-	6135.3s	1991.7s	-
26	52	417.6s	2.0s	1.8s	771.1s	1.7s	4.6s	-	2^{-73}	-	346.1s	404.7s	-	1183.7s	786.5s	-
27	54	730.5s	2.1s	8.3s	1133.3s	1.3s	4.4s	-	2^{-76}	-	424.6s	421.9s	-	1619.1s	1711.5s	-
28	56	450.8s	3.3s	7.2s	1417.3s	1.7s	4.3s	-	2^{-79}	-	332.9s	408.4s	-	6503.0s	512.4s	-
Total		4889.1s	62.7s	88.3s	9049.1s	68.6s	136.3s	11.9h		62.0h	4.0h	2.6h	61.2h	14.9h	5.3h	53.2h

Table 14: Experimental results of RECTANGLE.

Differential property																	
Round	$\#S_D$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Pr_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	1s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
2	2	0.0s	0.0s	0.1s	0.0s	0.0s	0.0s	1s	2^{-4}	0.0s	0.1s	0.1s	0.1s	0.1s	0.1s	8s
3	3	0.0s	0.0s	0.1s	0.0s	0.0s	0.0s	1s	2^{-7}	0.1s	0.1s	0.2s	0.1s	0.1s	0.2s	27s
4	4	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	2s	2^{-10}	0.2s	0.1s	0.2s	0.2s	0.2s	0.2s	128s
5	6	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	11s	2^{-14}	0.7s	0.3s	0.4s	0.5s	0.3s	0.4s	6s
6	8	0.3s	0.2s	0.2s	0.2s	0.1s	0.2s	13s	2^{-18}	1.4s	0.8s	0.6s	1.0s	0.4s	0.6s	17s
7	11	0.7s	0.7s	0.7s	0.6s	0.6s	0.7s	11s	2^{-25}	8.0s	5.1s	6.3s	7.3s	5.6s	7.5s	28s
8	13	0.9s	0.4s	0.9s	0.7s	0.5s	1.0s	11s	2^{-31}	29.3s	15.1s	22.6s	35.8s	20.0s	44.6s	41s
9	15	1.7s	0.8s	1.6s	2.1s	0.6s	2.2s	11s	2^{-36}	76.0s	19.1s	23.2s	71.1s	20.4s	33.1s	96s
10	17	3.0s	1.0s	2.4s	3.9s	0.9s	4.3s	25s	2^{-41}	165.9s	20.7s	82.1s	157.1s	57.8s	92.3s	297s
11	19	5.0s	1.1s	2.9s	7.3s	0.8s	11.4s	47s	2^{-46}	289.7s	38.4s	77.8s	325.6s	40.5s	159.8s	669s
12	21	10.3s	1.3s	11.6s	11.1s	1.6s	15.1s	120s	2^{-51}	562.2s	40.3s	142.8s	793.1s	114.6s	205.2s	2798s
13	23	15.7s	1.3s	10.5s	17.2s	1.2s	16.4s	597s	2^{-56}	1167.3s	49.0s	200.3s	1401.6s	114.2s	560.0s	12410s
14	25	21.2s	1.3s	12.7s	39.5s	2.5s	28.7s	2218s	2^{-61}	1701.4s	59.7s	396.8s	2519.3s	154.6s	618.2s	40989s
15	27	35.1s	1.5s	30.0s	41.3s	1.7s	35.3s	12753s	2^{-66}	2880.8s	60.7s	407.3s	3237.2s	271.4s	644.4s	-
16	29	60.3s	1.8s	22.6s	96.8s	2.3s	51.4s	36891s	2^{-71}	3851.9s	68.8s	606.7s	5751.6s	184.5s	1125.7s	-
17	31	48.0s	2.6s	34.0s	115.0s	5.1s	145.5s	-	2^{-76}	4619.1s	95.5s	494.9s	8050.6s	320.9s	2270.6s	-
18	33	91.0s	3.9s	38.8s	193.7s	3.3s	129.0s	-	2^{-81}	6793.6s	91.7s	715.2s	4.0h	354.2s	2576.7s	-
19	35	120.8s	9.6s	83.3s	184.8s	3.8s	193.5s	-	2^{-86}	8283.0s	128.7s	402.2s	3.9h	444.8s	930.2s	-
20	37	169.0s	6.4s	63.7s	277.8s	4.7s	102.6s	-	2^{-91}	3.2h	107.3s	304.8s	8.4h	527.5s	938.3s	-
21	39	194.0s	8.2s	36.8s	311.6s	7.7s	180.6s	-	2^{-96}	4.6h	148.3s	460.4s	12.4h	624.3s	1132.1s	-
22	41	223.1s	8.7s	70.6s	387.5s	7.2s	245.5s	-	2^{-101}	7.0h	219.5s	702.2s	11.6h	927.0s	3118.8s	-
23	43	315.7s	9.2s	60.8s	396.7s	12.6s	402.2s	-	2^{-106}	8.7h	187.1s	391.6s	16.1h	721.4s	3243.3s	-
24	45	270.6s	6.0s	117.8s	572.3s	10.0s	365.5s	-	2^{-111}	9.2h	241.0s	1106.4s	-	1124.1s	2689.9s	-
25	47	292.4s	10.0s	80.8s	796.7s	28.5s	370.8s	-	2^{-116}	10.4h	262.1s	806.3s	-	1888.1s	3221.4s	-
Total		1878.9s	76.2s	682.9s	3457.1s	96.1s	2302.3s	14.6h		51.7h	1859.4s	7351.2s	62.6h	7917.1s	6.6h	15.9h
Linear property																	
Round	$\#S_L$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Cor_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	1s	2^{-1}	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s
2	2	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	1s	2^{-2}	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	1s
3	3	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	1s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	5s
4	4	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	2s	2^{-6}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	9s
5	6	0.1s	0.1s	0.1s	0.1s	0.1s	0.2s	6s	2^{-8}	0.3s	0.1s	0.2s	0.3s	0.1s	0.2s	41s
6	8	0.2s	0.1s	0.1s	0.2s	0.1s	0.2s	8s	2^{-10}	0.6s	0.3s	0.4s	0.5s	0.2s	0.3s	6s
7	10	0.5s	0.4s	0.2s	0.3s	0.2s	0.3s	5s	2^{-13}	3.2s	1.4s	1.9s	2.9s	1.4s	3.1s	15s
8	12	0.6s	0.6s	0.4s	0.6s	0.4s	0.5s	9s	2^{-16}	11.3s	4.3s	4.4s	12.9s	11.1s	14.6s	24s
9	14	1.0s	1.2s	0.8s	1.3s	1.2s	0.6s	11s	2^{-19}	40.0s	14.6s	15.1s	31.0s	27.0s	24.8s	78s
10	16	2.3s	1.8s	0.8s	3.1s	2.8s	1.6s	25s	2^{-22}	95.5s	46.6s	30.2s	88.6s	72.1s	47.2s	260s
11	18	4.5s	3.5s	1.4s	6.8s	6.5s	1.8s	38s	2^{-25}	222.7s	155.0s	75.6s	314.2s	169.9s	126.7s	1772s
12	20	6.4s	5.2s	2.2s	9.3s	10.0s	3.1s	131s	2^{-28}	483.3s	258.4s	168.7s	677.1s	283.2s	225.5s	5927s
13	22	9.8s	7.7s	2.6s	13.9s	12.8s	3.8s	428s	2^{-31}	1340.7s	828.6s	312.4s	1250.9s	832.9s	834.5s	31491s
14	24	18.2s	9.2s	7.0s	21.7s	29.5s	7.3s	1615s	2^{-34}	2618.2s	1545.9s	585.7s	3349.3s	1742.9s	583.8s	177473s
15	26	25.5s	17.7s	6.1s	40.7s	26.0s	12.8s	5588s	2^{-37}	3794.3s	2588.5s	1311.7s	4285.1s	3572.2s	1109.3s	-
16	28	34.6s	27.4s	6.9s	75.4s	47.8s	12.7s	21352s	2^{-40}	7710.3s	4636.6s	1644.8s	8345.7s	7103.5s	2965.5s	-
17	30	42.4s	33.0s	8.6s	97.4s	58.6s	14.3s	-	2^{-42}	4966.9s	1278.3s	294.4s	4328.0s	1192.9s	139.8s	-
18	32	62.5s	42.5s	8.0s	81.4s	70.2s	26.5s	-	2^{-45}	3.0h	4152.5s	765.6s	3.9h	6637.5s	677.9s	-
19	34	99.5s	44.7s	20.7s	147.6s	86.9s	23.4s	-	2^{-48}	4.0h	5819.1s	2215.6s	12.6h	9508.1s	2565.4s	-
20	36	126.6s	63.1s	21.0s	260.1s	152.0s	32.7s	-	2^{-51}	5.3h	7073.1s	2853.6s	19.4h	4.0h	2214.9s	-
21	38	121.8s	90.1s	18.5s	212.1s	183.7s	35.8s	-	2^{-54}	11.2h	3.1h	1498.2s	-	9.0h	5496.8s	-
22	40	179.6s	128.1s	18.2s	213.0s	214.0s	51.4s	-	2^{-57}	15.2h	3.4h	3556.9s	-	15.9h	4551.9s	-
23	42	229.7s	125.6s	30.1s	395.4s	222.6s	21.6s	-	2^{-60}	-	4.0h	2198.4s	-	17.1h	5900.3s	-
24	44	179.6s	115.1s	40.4s	361.9s	251.6s	27.1s	-	2^{-63}	-	10.8h	2669.2s	-	-	4.9h	-
25	46	331.4s	149.2s	32.1s	547.5s	325.5s	66.4s	-	2^{-66}	-	12.3h	6083.9s	-	-	6.8h	-
Total		1476.9s	866.4s	226.3s	2489.8s	1702.6s	344.2s	8.1h		44.6h	41.5h	7.3h	42.3h	54.6h	19.3h	60.3h

Table 15: Experimental results of `LBlock`.

Differential property																	
Round	$\#S_D$	$T^{\emptyset}_{\text{CaD}}$	$T^{\mathcal{R}^{-1}}_{\text{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\text{Cry}}$	$T^{\mathcal{R}^{-1}}_{\text{Cry}}$	T^{0}_{Cry}	$T_{\texttt{MILP}}$	$\text{Pr}_{\texttt{Opt}}$	$T^{\emptyset}_{\text{CaD}}$	$T^{\mathcal{R}^{-1}}_{\text{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\text{Cry}}$	$T^{\mathcal{R}^{-1}}_{\text{Cry}}$	T^{0}_{Cry}	$T_{\texttt{MILP}}$
1	0	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	0s	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	0s
2	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
3	2	0.1s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-4}	0.1s	0.1s	0.2s	0.1s	0.1s	0.2s	0s
4	3	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-6}	0.1s	0.1s	0.2s	0.1s	0.1s	0.2s	1s
5	4	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-8}	0.1s	0.1s	0.2s	0.2s	0.1s	0.2s	1s
6	6	0.1s	0.1s	0.1s	0.1s	0.1s	0.2s	1s	2^{-12}	0.4s	0.3s	0.5s	0.4s	0.3s	0.4s	1s
7	8	0.2s	0.1s	0.2s	0.2s	0.2s	0.2s	1s	2^{-16}	1.1s	0.6s	0.9s	0.6s	0.4s	0.7s	1s
8	11	0.3s	0.3s	0.4s	0.3s	0.3s	0.3s	2s	2^{-22}	2.1s	1.9s	2.5s	1.2s	1.0s	1.3s	2s
9	14	0.5s	0.4s	0.5s	0.4s	0.4s	0.4s	2s	2^{-28}	2.9s	2.3s	3.4s	2.0s	1.7s	2.1s	2s
10	18	1.2s	0.9s	1.2s	1.0s	0.7s	1.1s	6s	2^{-36}	5.8s	4.6s	6.6s	5.5s	4.2s	6.4s	6s
11	22	1.8s	1.5s	1.8s	2.9s	2.0s	2.7s	4s	2^{-44}	11.7s	8.9s	10.7s	13.9s	10.6s	17.3s	4s
12	24	1.6s	0.8s	0.7s	2.9s	0.8s	1.0s	5s	2^{-48}	11.5s	4.0s	3.7s	15.6s	4.4s	6.3s	5s
13	27	3.1s	1.4s	1.8s	5.8s	1.4s	2.1s	25s	2^{-56}	33.5s	17.5s	15.6s	49.5s	28.9s	53.2s	812s
14	30	4.5s	1.7s	1.5s	10.5s	2.7s	2.8s	8s	2^{-62}	57.2s	12.9s	25.2s	87.8s	33.7s	64.8s	848s
15	32	4.2s	1.4s	1.7s	9.0s	0.8s	2.1s	19s	2^{-66}	69.2s	9.4s	13.2s	83.1s	14.6s	24.6s	820s
16	35	10.2s	2.2s	2.5s	23.7s	2.6s	4.1s	30s	2^{-72}	135.2s	26.6s	27.0s	182.2s	36.8s	106.1s	6002s
17	36	5.0s	0.6s	0.4s	13.8s	0.4s	0.4s	29s	2^{-76}	108.4s	11.9s	17.4s	167.0s	33.9s	35.7s	-
18	39	19.5s	1.4s	1.9s	48.6s	1.2s	2.4s	10s	2^{-82}	269.4s	22.6s	39.4s	518.0s	63.4s	123.4s	-
19	41	19.4s	0.9s	1.1s	32.3s	0.5s	0.7s	190s	2^{-86}	411.6s	19.3s	18.7s	515.4s	41.0s	79.0s	-
20	44	28.4s	1.6s	1.7s	106.0s	1.0s	2.1s	1828s	2^{-92}	606.8s	40.6s	42.5s	1202.6s	151.1s	265.9s	-
21	45	13.1s	0.6s	0.4s	76.3s	0.3s	0.4s	-	2^{-96}	582.5s	25.2s	32.5s	1080.4s	38.8s	81.8s	-
22	48	54.6s	1.1s	1.0s	112.8s	0.6s	0.9s	-	2^{-102}	1021.6s	32.4s	40.4s	2629.1s	148.4s	211.8s	-
23	50	54.1s	0.8s	1.3s	146.5s	0.5s	0.8s	-	2^{-106}	710.2s	21.2s	17.5s	1490.3s	60.1s	60.1s	-
24	53	81.5s	1.8s	1.3s	485.3s	0.8s	1.4s	-	2^{-112}	1432.8s	39.2s	33.8s	4623.3s	79.1s	113.9s	-
25	54	36.0s	0.8s	0.8s	280.0s	0.3s	0.7s	-	2^{-115}	924.9s	18.0s	21.5s	1620.6s	42.3s	46.9s	-
26	57	107.8s	1.3s	1.8s	347.2s	0.7s	1.0s	-	2^{-121}	2069.9s	35.3s	43.9s	3488.8s	65.1s	93.2s	-
27	59	73.1s	1.1s	0.8s	497.5s	0.6s	1.3s	-	2^{-126}	1746.7s	37.8s	58.7s	3892.5s	99.6s	111.3s	-
28	62	114.6s	1.6s	2.7s	765.8s	1.0s	1.1s	-	2^{-131}	2012.3s	27.1s	52.6s	5177.3s	73.2s	93.9s	-
29	63	87.9s	0.8s	1.5s	425.1s	0.4s	0.7s	-	2^{-135}	1971.9s	23.7s	34.9s	3514.1s	78.9s	132.5s	-
30	66	168.0s	1.9s	1.4s	874.6s	0.9s	1.2s	-	2^{-141}	3388.0s	32.8s	54.8s	2.5h	102.5s	152.1s	-
31	68	141.3s	1.5s	1.7s	859.6s	0.7s	1.2s	-	2^{-146}	2578.6s	48.1s	60.9s	3.4h	127.1s	226.9s	-
32	71	219.5s	2.2s	3.1s	1097.4s	1.2s	1.3s	-	2^{-151}	3397.1s	43.0s	69.2s	2.5h	125.8s	137.9s	-
Total		1251.8s	31.1s	35.4s	6225.8s	23.5s	35.1s	352s		6.5h	567.9s	748.5s	16.9h	1467.7s	2250.2s	2.4h
Linear property																	
Round	$\#S_L$	$T^{\emptyset}_{\text{CaD}}$	$T^{\mathcal{R}^{-1}}_{\text{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\text{Cry}}$	$T^{\mathcal{R}^{-1}}_{\text{Cry}}$	T^{0}_{Cry}	$T_{\texttt{MILP}}$	$\text{Cor}_{\texttt{Opt}}$	$T^{\emptyset}_{\text{CaD}}$	$T^{\mathcal{R}^{-1}}_{\text{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\text{Cry}}$	$T^{\mathcal{R}^{-1}}_{\text{Cry}}$	T^{0}_{Cry}	$T_{\texttt{MILP}}$
1	0	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	0s	1	0.0s	0.0s	0.1s	0.0s	0.0s	0.1s	0s
2	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
3	2	0.0s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
4	3	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
5	4	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
6	6	0.1s	0.1s	0.1s	0.1s	0.1s	0.2s	1s	2^{-6}	0.2s	0.2s	0.2s	0.2s	0.2s	0.2s	1s
7	8	0.2s	0.2s	0.2s	0.2s	0.2s	0.2s	1s	2^{-8}	0.4s	0.3s	0.4s	0.3s	0.3s	0.3s	2s
8	11	0.4s	0.4s	0.4s	0.3s	0.3s	0.4s	2s	2^{-11}	0.7s	0.7s	0.7s	0.6s	0.5s	0.6s	2s
9	14	0.7s	0.6s	0.7s	0.4s	0.4s	0.5s	2s	2^{-14}	1.0s	0.8s	1.0s	0.8s	0.6s	0.9s	2s
10	18	1.3s	1.0s	1.2s	1.2s	0.9s	1.3s	6s	2^{-18}	1.7s	1.6s	1.9s	2.4s	1.4s	1.9s	8s
11	22	1.9s	1.6s	2.0s	2.9s	2.5s	3.7s	4s	2^{-22}	3.1s	2.6s	3.4s	5.2s	3.8s	5.6s	4s
12	24	1.7s	0.9s	1.1s	3.0s	1.2s	1.3s	6s	2^{-24}	3.9s	1.5s	1.5s	6.9s	3.1s	1.8s	5s
13	27	3.9s	1.3s	1.9s	7.3s	1.6s	1.8s	38s	2^{-27}	9.2s	4.2s	4.1s	9.6s	6.1s	6.6s	2103s
14	30	5.1s	2.1s	2.2s	13.1s	2.9s	2.8s	10s	2^{-30}	8.4s	2.8s	3.8s	26.8s	4.8s	11.8s	15s
15	32	3.9s	1.1s	1.6s	17.5s	0.9s	1.3s	28s	2^{-33}	18.9s	5.0s	6.9s	32.6s	15.5s	11.1s	5669s
16	35	10.2s	2.1s	2.5s	26.0s	2.7s	3.5s	55s	2^{-36}	24.1s	6.3s	10.5s	65.8s	10.7s	16.0s	-
17	36	4.5s	0.5s	0.6s	15.1s	0.4s	0.6s	31s	2^{-37}	16.9s	1.7s	2.8s	41.5s	4.8s	3.0s	-
18	39	17.1s	1.5s	1.8s	39.4s	1.6s	1.9s	11s	2^{-40}	37.3s	4.1s	5.6s	129.6s	6.7s	7.7s	-
19	41	17.0s	0.8s	1.1s	38.0s	0.5s	0.7s	6s	2^{-42}	40.1s	3.6s	4.0s	132.5s	5.3s	16.3s	-
20	44	34.4s	1.6s	1.6s	80.0s	1.6s	2.8s	40s	2^{-45}	74.2s	6.3s	4.6s	306.1s	7.7s	12.3s	-
21	45	11.7s	0.6s	0.4s	79.1s	0.3s	0.6s	-	2^{-47}	59.6s	6.2s	4.7s	240.7s	7.7s	7.6s	-
22	48	58.0s	1.3s	1.8s	173.6s	0.9s	0.9s	-	2^{-50}	144.8s	6.2s	6.5s	411.0s	18.2s	12.5s	-
23	50	48.5s	1.0s	1.2s	168.4s	0.6s	0.7s	-	2^{-52}	171.2s	4.1s	8.7s	604.2s	16.1s	21.7s	-
24	53	89.8s	2.0s	1.9s	312.5s	0.9s	1.3s	-	2^{-55}	264.9s	12.7s	9.2s	1089.1s	15.8s	34.1s	-
25	54	32.3s	0.7s	0.4s	250.3s	0.4s	0.6s	-	2^{-56}	128.4s	3.6s	3.9s	787.1s	4.9s	8.2s	-
26	57	106.0s	1.5s	2.2s	587.7s	1.1s	1.5s	-	2^{-59}	311.9s	8.7s	12.3s	1531.2s	14.9s	15.6s	-
27	59	87.7s	1.3s	1.6s	630.3s	0.8s	1.9s	-	2^{-62}	365.2s	9.0s	11.1s	1923.1s	19.6s	38.6s	-
28	62	136.8s	2.4s	2.2s	863.4s	1.5s	2.3s	-	2^{-65}	438.5s	18.5s	22.1s	1504.3s	24.0s	51.1s	-
29	63	49.5s	1.1s	1.0s	431.6s	0.5s	1.0s	-	2^{-66}	392.4s	5.2s	6.1s	598.4s	13.1s	20.0s	-
30	66	168.3s	1.6s	2.8s	1093.4s	1.2s	1.7s	-	2^{-69}	632.0s	12.4s	10.0s	1619.4s	18.2s	32.0s	-
31	68	124.9s	1.5s	2.1s	645.3s	1.2s	1.3s	-	2^{-72}	799.8s	19.1s	27.7s	2942.4s	35.6s	50.3s	-
32	71	225.9s	2.6s	3.3s	957.0s	1.8s	1.5s	-	2^{-74}	747.4s	12.2s	16.6s	2177.9s	19.9s	21.8s	-
Total		1242.0s	33.5s	39.9s	6437.4s	29.5s	38.8s	242s		4696.7s	159.9s	190.8s	4.5h	279.8s	410.2s	2.2h

Table 16: Experimental results of TWINE.

Differential property																
Round	$\#S_D$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Pr_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	0	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	0s	1	0.0s	0.0s	0.1s	0.0s	0.0s	0.0s	0s
2	1	0.0s	0.0s	0.1s	0.1s	0.1s	0.1s	0s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
3	2	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
4	3	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-6}	0.2s	0.1s	0.1s	0.2s	0.2s	0.2s	1s
5	4	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-8}	0.2s	0.2s	0.2s	0.2s	0.2s	0.2s	1s
6	6	0.1s	0.1s	0.2s	0.2s	0.2s	0.2s	1s	2^{-12}	0.6s	0.4s	0.5s	0.5s	0.4s	0.5s	1s
7	8	0.2s	0.1s	0.2s	0.2s	0.2s	0.2s	1s	2^{-16}	1.0s	0.6s	0.8s	0.7s	0.5s	0.6s	1s
8	11	0.3s	0.3s	0.4s	0.4s	0.3s	0.3s	1s	2^{-22}	2.7s	1.9s	2.3s	1.5s	1.0s	1.4s	2s
9	14	0.5s	0.4s	0.5s	0.4s	0.3s	0.4s	2s	2^{-28}	3.6s	2.5s	3.3s	2.6s	1.5s	2.1s	2s
10	18	1.2s	1.2s	1.3s	1.0s	1.1s	1.1s	6s	2^{-38}	10.7s	8.2s	10.0s	10.1s	8.8s	9.8s	52s
11	22	2.2s	1.9s	1.9s	3.4s	3.0s	3.5s	4s	2^{-46}	20.6s	14.5s	15.4s	28.0s	27.2s	29.6s	49s
12	24	2.0s	0.9s	1.0s	3.1s	0.9s	1.0s	4s	2^{-51}	21.4s	11.7s	10.9s	43.5s	15.3s	22.1s	63s
13	27	4.7s	1.7s	1.3s	4.7s	1.8s	2.2s	24s	2^{-58}	53.7s	30.2s	30.4s	81.7s	69.0s	82.4s	7905s
14	30	6.2s	1.6s	2.5s	15.2s	3.4s	3.4s	14s	2^{-64}	105.1s	31.9s	51.8s	139.7s	82.3s	118.4s	17153s
15	32	4.5s	1.4s	1.0s	10.7s	0.9s	1.3s	14s	2^{-68}	94.9s	16.6s	22.1s	163.5s	56.8s	75.5s	28840s
16	35	14.6s	2.6s	2.5s	40.1s	3.9s	3.6s	19s	2^{-74}	207.8s	48.1s	59.7s	380.7s	97.0s	133.1s	-
17	36	6.5s	0.5s	0.6s	17.2s	0.4s	0.5s	17s	2^{-77}	138.5s	11.6s	15.9s	294.5s	46.2s	76.1s	-
18	39	21.4s	1.6s	1.8s	71.1s	1.7s	1.4s	9s	2^{-83}	395.8s	35.8s	37.2s	814.1s	116.3s	140.8s	-
19	41	17.3s	1.0s	0.9s	55.4s	0.8s	0.7s	5s	2^{-88}	564.2s	51.9s	52.9s	1153.9s	118.8s	182.6s	-
20	44	37.3s	2.0s	2.0s	169.3s	1.3s	1.9s	17s	2^{-94}	1091.2s	79.2s	94.3s	1852.2s	235.2s	223.7s	-
21	45	19.7s	0.9s	0.4s	85.6s	0.3s	0.3s	-	2^{-97}	745.8s	20.5s	16.7s	1382.7s	45.4s	63.1s	-
22	48	61.3s	1.8s	1.0s	392.2s	0.8s	0.9s	-	2^{-103}	1302.9s	31.0s	26.0s	2560.2s	89.6s	132.2s	-
23	50	53.7s	1.2s	0.6s	260.6s	0.7s	0.7s	-	2^{-107}	1031.4s	17.0s	25.9s	3388.3s	32.5s	50.4s	-
24	53	103.1s	2.5s	1.3s	472.7s	1.1s	1.3s	-	2^{-113}	2105.8s	38.1s	29.0s	4141.7s	25.5s	51.4s	-
25	54	55.7s	0.7s	0.5s	249.4s	0.6s	0.4s	-	2^{-116}	1047.5s	8.7s	12.5s	4121.9s	20.5s	33.1s	-
26	57	125.5s	1.6s	1.2s	526.3s	1.0s	1.0s	-	2^{-122}	2166.5s	22.2s	24.4s	7015.3s	63.0s	69.8s	-
27	59	94.9s	1.4s	0.8s	481.3s	0.8s	1.2s	-	2^{-126}	1780.4s	16.5s	25.5s	5017.5s	21.0s	70.6s	-
28	62	141.3s	2.6s	2.0s	872.2s	1.0s	1.3s	-	2^{-132}	3061.5s	19.8s	33.8s	7978.8s	33.5s	97.5s	-
29	63	76.7s	1.0s	1.2s	472.8s	0.7s	0.6s	-	2^{-136}	2248.9s	29.9s	31.5s	6112.0s	131.5s	91.0s	-
30	66	141.8s	1.9s	1.5s	878.2s	1.1s	1.5s	-	2^{-142}	3717.6s	36.2s	40.9s	3.9h	97.7s	141.6s	-
31	68	99.6s	1.9s	2.4s	796.7s	0.9s	0.8s	-	2^{-146}	3306.6s	23.8s	25.8s	3.3h	64.5s	77.3s	-
32	71	222.6s	2.9s	2.7s	801.9s	1.6s	1.4s	-	2^{-152}	5053.9s	44.1s	72.8s	6.7h	71.6s	89.8s	-
33	72	94.0s	1.4s	0.9s	405.7s	0.6s	1.0s	-	2^{-155}	3176.0s	29.0s	20.8s	4.1h	82.2s	77.2s	-
34	75	335.3s	2.3s	1.5s	1520.1s	1.3s	1.4s	-	2^{-161}	5699.4s	36.6s	49.8s	7.8h	90.5s	133.3s	-
35	77	284.3s	2.3s	1.2s	785.0s	1.5s	1.2s	-	2^{-166}	5624.9s	71.2s	94.3s	7.9h	205.6s	183.2s	-
36	80	404.8s	3.5s	3.2s	1530.4s	2.0s	1.5s	-	2^{-172}	7104.5s	74.4s	94.8s	8.7h	317.3s	376.0s	-
Total		2433.4s	47.5s	40.5s	3.0h	36.5s	38.8s	139s		14.4h	864.6s	1032.5s	55.3h	2269.0s	2836.9s	15.0h
Linear property																
Round	$\#S_L$	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}	Cor_{Opt}	$T^{\emptyset}_{CaD}$	$T^{\mathcal{R}^{-1}}_{CaD}$	T^{0}_{CaD}	$T^{\emptyset}_{Cry}$	$T^{\mathcal{R}^{-1}}_{Cry}$	T^{0}_{Cry}	T_{MILP}
1	0	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	0s	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.1s	0s
2	1	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s
3	2	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	0s	2^{-2}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
4	3	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
5	4	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s	2^{-4}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	1s
6	6	0.2s	0.1s	0.2s	0.2s	0.2s	0.2s	1s	2^{-6}	0.2s	0.2s	0.3s	0.2s	0.2s	0.2s	2s
7	8	0.2s	0.1s	0.2s	0.2s	0.2s	0.2s	2s	2^{-8}	0.4s	0.2s	0.3s	0.3s	0.2s	0.2s	2s
8	11	0.4s	0.3s	0.4s	0.3s	0.3s	0.3s	3s	2^{-11}	0.6s	0.5s	0.6s	0.5s	0.4s	0.5s	3s
9	14	0.6s	0.5s	0.4s	0.4s	0.5s	0.4s	3s	2^{-14}	1.5s	0.9s	0.9s	0.9s	0.5s	0.7s	4s
10	18	1.3s	1.3s	1.3s	1.1s	1.0s	1.2s	10s	2^{-18}	2.6s	2.1s	1.8s	2.0s	1.2s	1.8s	17s
11	22	2.7s	1.6s	2.2s	4.4s	2.4s	2.8s	4s	2^{-22}	3.7s	3.4s	2.9s	6.0s	4.3s	5.0s	12s
12	24	2.1s	1.0s	0.9s	3.4s	1.2s	1.1s	6s	2^{-24}	4.4s	2.0s	1.2s	6.2s	2.5s	3.1s	8s
13	27	4.9s	1.3s	1.7s	9.1s	2.3s	2.1s	53s	2^{-27}	7.6s	2.4s	2.8s	13.7s	5.0s	2.6s	364s
14	30	5.6s	1.8s	2.5s	15.2s	1.8s	5.0s	12s	2^{-30}	12.0s	3.7s	3.5s	27.9s	4.9s	6.1s	16s
15	32	6.3s	1.0s	1.2s	17.6s	2.0s	1.1s	37s	2^{-32}	10.8s	2.6s	2.1s	30.1s	3.5s	2.9s	261s
16	35	13.5s	2.2s	2.7s	36.8s	4.0s	4.1s	49s	2^{-35}	22.2s	4.4s	5.3s	55.7s	5.5s	5.8s	66s
17	36	6.3s	0.8s	0.7s	33.4s	0.3s	0.5s	57s	2^{-36}	17.6s	1.6s	1.5s	53.4s	0.7s	1.1s	-
18	39	24.5s	1.7s	1.4s	82.6s	1.3s	2.4s	11s	2^{-39}	35.1s	3.6s	3.4s	84.7s	2.6s	3.3s	-
19	41	17.3s	1.4s	1.9s	45.1s	0.7s	1.1s	7s	2^{-41}	23.0s	2.8s	1.9s	138.6s	1.5s	1.9s	-
20	44	48.2s	1.8s	1.8s	140.2s	1.9s	1.7s	42s	2^{-44}	72.5s	4.3s	3.9s	226.9s	2.8s	3.1s	-
21	45	19.6s	0.8s	0.4s	90.2s	0.4s	0.4s	-	2^{-45}	24.1s	1.0s	2.1s	116.6s	0.6s	0.8s	-
22	48	56.0s	1.5s	1.4s	191.6s	0.8s	0.9s	-	2^{-48}	127.5s	4.5s	3.1s	442.7s	2.0s	2.2s	-
23	50	61.0s	1.2s	0.6s	197.6s	0.6s	0.6s	-	2^{-50}	81.2s	2.7s	2.1s	398.6s	1.7s	1.8s	-
24	53	95.3s	2.2s	1.9s	289.1s	1.0s	1.1s	-	2^{-53}	171.5s	4.8s	3.5s	760.4s	1.9s	2.4s	-
25	54	48.3s	1.0s	0.4s	285.4s	0.5s	0.5s	-	2^{-54}	94.4s	2.2s	1.9s	180.5s	0.8s	1.5s	-
26	57	106.1s	1.8s	1.2s	700.0s	1.0s	1.2s	-	2^{-57}	219.0s	3.4s	2.5s	1219.8s	2.4s	2.4s	-
27	59	109.3s	1.8s	0.8s	610.2s	1.1s	0.7s	-	2^{-59}	202.8s	3.8s	3.2s	754.1s	1.5s	1.2s	-
28	62	128.4s	2.4s	2.8s	1025.2s	1.1s	1.3s	-	2^{-62}	378.8s	5.0s	5.8s	1025.0s	2.7s	3.1s	-
29	63	59.2s	1.3s	0.5s	366.7s	0.6s	0.5s	-	2^{-63}	176.4s	1.7s	1.9s	500.0s	1.0s	1.6s	-
30	66	168.0s	2.3s	2.3s	868.9s	1.2s	1.1s	-	2^{-66}	402.4s	5.2s	4.2s	1039.4s	1.5s	2.4s	-
31	68	195.1s	2.1s	0.9s	709.7s	1.0s	0.9s	-	2^{-68}	327.0s	3.9s	4.6s	975.0s	2.8s	2.2s	-
32	71	220.2s	2.2s	1.9s	1207.4s	1.3s	1.8s	-	2^{-71}	474.9s	6.3s	5.6s	2072.6s	3.4s	2.9s	-
33	72	143.5s	1.4s	0.6s	471.4s	0.5s	1.1s	-	2^{-72}	244.7s	3.5s	2.1s	687.6s	1.7s	1.2s	-
34	75	379.8s	2.7s	1.6s	927.0s	1.1s	1.4s	-	2^{-75}	605.9s	6.6s	5.0s	2083.3s	2.6s	3.1s	-
35	77	370.2s	2.4s	1.4s	1060.9s	1.1s	1.1s	-	2^{-77}	445.7s	4.5s	3.9s	1276.3s	2.7s	2.2s	-
36	80	407.6s	2.8s	3.8s	1422.2s	1.5s	1.9s	-	2^{-80}	843.3s	5.6s	7.4s	3231.0s	3.7s	5.1s	-
Total		2702.0s	47.3s	42.1s	3.0h	35.0s	41.5s	299s		5034.2s	99.5s	91.8s	4.8h	69.3s	75.1s	758s

Table 17: Experimental results of SPECK32.

	Differential property								Linear property							
Round	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}-1}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}-1}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}-1}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}-1}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
2	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
3	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
4	2^{-5}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
5	2^{-9}	0.7s	0.5s	0.4s	0.7s	0.6s	0.6s	9.8s	2^{-5}	0.2s	0.2s	0.1s	0.2s	0.2s	0.2s	-
6	2^{-13}	1.7s	1.2s	1.9s	3.5s	3.5s	3.7s	173.7s	2^{-7}	0.5s	0.5s	0.4s	0.5s	0.4s	0.5s	-
7	2^{-18}	12.8s	6.4s	11.3s	26.3s	25.5s	19.6s	7175.9s	2^{-9}	1.1s	1.0s	0.8s	1.3s	1.2s	1.1s	-
8	2^{-24}	87.5s	53.1s	72.1s	170.7s	195.8s	204.4s	-	2^{-12}	6.5s	8.0s	6.7s	16.9s	24.2s	22.3s	-
9	2^{-30}	417.5s	367.0s	376.1s	1106.5s	722.1s	767.6s	-	2^{-14}	16.4s	11.3s	7.8s	85.8s	49.4s	42.3s	-
10	2^{-34}	1080.6s	515.0s	688.1s	3757.9s	1261.0s	1780.9s	-	2^{-17}	98.5s	47.2s	78.3s	173.3s	180.0s	140.0s	-
11	2^{-38}	0.7h	0.2h	0.3h	3.2h	1.5h	1.2h	-	2^{-19}	138.2s	73.6s	75.4s	195.6s	235.3s	122.8s	-
12	2^{-42}	1.2h	0.3h	0.5h	4.4h	0.9h	1.5h	-	2^{-20}	57.9s	38.3s	60.8s	157.7s	90.2s	50.9s	-
13	2^{-45}	1.5h	0.3h	0.2h	5.8h	1.1h	0.5h	-	2^{-22}	138.3s	41.3s	19.4s	362.7s	174.2s	49.2s	-
14	2^{-49}	2.3h	0.3h	0.2h	12.7h	1.8h	1.3h	-	2^{-24}	202.0s	43.5s	88.5s	590.3s	112.6s	53.0s	-
15	2^{-54}	4.5h	0.7h	0.9h	34.4h	3.6h	3.2h	-	2^{-26}	294.4s	60.1s	21.9s	1146.0s	83.0s	39.7s	-
16	2^{-58}	6.2h	0.3h	0.2h	48.6h	2.8h	1.7h	-	2^{-28}	731.3s	44.9s	79.9s	1476.3s	227.3s	156.4s	-
17	2^{-63}	13.8h	1.0h	1.3h	158.9h	5.9h	7.6h	-	2^{-30}	829.0s	112.0s	32.7s	3581.2s	87.5s	179.6s	-
18	2^{-69}	41.3h	7.3h	12.5h	-	49.2h	46.5h	-	2^{-34}	1.6h	0.7h	0.2h	6.2h	1.2h	1.5h	-
19	2^{-74}	71.3h	18.3h	43.1h	-	43.8h	129.4h	-	2^{-36}	1.8h	1.2h	0.7h	8.6h	2.7h	0.8h	-
20	2^{-77}	73.6h	4.7h	2.8h	-	8.7h	57.0h	-	2^{-38}	1.6h	1.8h	1.1h	10.0h	3.4h	1.9h	-
21	2^{-81}	-	17.2h	11.4h	-	21.9h	-	-	2^{-40}	2.0h	1.0h	0.8h	9.3h	0.8h	1.7h	-
22	2^{-85}	-	15.1h	10.5h	-	36.9h	-	-	2^{-42}	1.7h	0.6h	0.7h	13.9h	1.6h	1.6h	-
Total		216.8h	66.0h	84.2h	269.2h	178.6h	250.8h	2.0h		9.3h	5.4h	3.7h	50.2h	10.0h	7.7h	-

Table 18: Experimental results of SPECK48.

	Differential property								Linear property							
Round	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}-1}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}-1}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}-1}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}-1}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
2	2^{-1}	0.0s	0.1s	0.1s	0.1s	0.1s	0.1s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
3	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
4	2^{-6}	0.3s	0.3s	0.3s	0.2s	0.3s	0.3s	-	2^{-3}	0.1s	0.1s	0.1s	0.2s	0.2s	0.2s	-
5	2^{-10}	1.2s	1.1s	1.2s	1.0s	1.4s	1.2s	32.9s	2^{-6}	0.7s	0.5s	0.7s	0.5s	0.6s	1.0s	-
6	2^{-14}	5.1s	3.0s	5.9s	5.4s	5.1s	6.5s	1482.7s	2^{-8}	1.5s	1.7s	2.0s	3.3s	2.2s	3.7s	-
7	2^{-19}	27.9s	19.0s	31.6s	40.3s	33.0s	42.0s	11.4h	2^{-12}	21.8s	14.5s	18.8s	52.8s	44.5s	38.9s	-
8	2^{-26}	319.7s	207.8s	283.6s	1053.2s	640.0s	529.6s	-	2^{-15}	111.9s	69.8s	71.5s	276.4s	234.4s	190.9s	-
9	2^{-33}	1.0h	0.6h	0.9h	2.4h	3.2h	2.1h	-	2^{-19}	1679.3s	1030.9s	1257.3s	4362.6s	4617.3s	2902.8s	-
10	2^{-40}	7.9h	5.3h	6.1h	33.7h	22.8h	27.2h	-	2^{-22}	1.6h	1.2h	0.8h	5.5h	4.6h	2.3h	-
11	2^{-45}	18.1h	5.0h	7.2h	94.7h	18.3h	14.4h	-	2^{-25}	5.9h	3.4h	2.8h	16.1h	10.3h	11.4h	-
12	2^{-49}	24.5h	2.8h	4.2h	-	9.9h	8.9h	-	2^{-28}	15.8h	5.9h	5.4h	75.9h	23.3h	32.2h	-
13	2^{-54}	45.3h	4.9h	6.3h	-	16.2h	24.8h	-	2^{-30}	10.6h	1.8h	4.2h	96.4h	7.7h	12.9h	-
14	2^{-58}	59.8h	4.1h	4.0h	-	12.6h	15.1h	-	2^{-33}	33.8h	8.5h	9.5h	-	57.1h	163.7h	-
15	2^{-63}	107.3h	3.8h	8.2h	-	8.8h	24.6h	-	2^{-37}	178.1h	39.0h	45.7h	-	-	-	-
16	2^{-68}	-	5.5h	7.8h	-	27.7h	17.0h	-	2^{-39}	-	24.8h	42.3h	-	-	-	-
17	2^{-75}	-	42.0h	64.0h	-	-	-	-	2^{-43}	-	146.3h	162.0h	-	-	-	-
18	2^{-82}	-	291.2h	-	-	-	-	-	2^{-45}	-	124.3h	-	-	-	-	-
19	-	-	-	-	-	-	-	-	2^{-48}	-	56.9h	-	-	-	-	-
20	-	-	-	-	-	-	-	-	2^{-51}	-	101.1h	-	-	-	-	-
21	-	-	-	-	-	-	-	-	2^{-54}	-	50.1h	-	-	-	-	-
22	-	-	-	-	-	-	-	-	2^{-57}	-	182.4h	-	-	-	-	-
23	-	-	-	-	-	-	-	-	2^{-59}	-	79.5h	-	-	-	-	-
Total		264.0h	365.2h	108.8h	131.0h	119.6h	134.2h	11.8h		246.4h	825.7h	273.1h	195.3h	104.3h	223.4h	-

Table 19: Experimental results of SPECK64.

	Differential property								Linear property							
Round	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
2	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
3	2^{-3}	0.1s	0.1s	0.1s	0.1s	0.1s	0.2s	-	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
4	2^{-6}	0.5s	0.5s	0.4s	0.5s	0.5s	0.3s	-	2^{-3}	0.2s	0.2s	0.1s	0.2s	0.2s	0.2s	-
5	2^{-10}	2.3s	1.6s	1.8s	3.5s	1.9s	3.0s	-	2^{-6}	1.2s	0.8s	1.0s	1.2s	1.6s	0.9s	-
6	2^{-15}	18.0s	13.9s	19.5s	29.2s	28.7s	25.8s	-	2^{-9}	5.7s	4.2s	4.8s	10.3s	13.6s	18.1s	-
7	2^{-21}	135.8s	69.6s	111.7s	212.5s	147.0s	213.2s	-	2^{-13}	61.6s	46.7s	60.5s	79.5s	132.2s	169.6s	-
8	2^{-29}	2233.4s	1465.9s	2223.1s	4943.1s	4181.3s	4714.0s	-	2^{-17}	867.7s	484.0s	765.9s	1814.9s	1630.6s	2580.6s	-
9	2^{-34}	1.8h	0.7h	0.8h	5.1h	1.6h	1.5h	-	2^{-19}	1163.1s	436.8s	1029.8s	2334.5s	1102.1s	1548.7s	-
10	2^{-38}	3.3h	0.5h	0.3h	9.8h	1.3h	1.2h	-	2^{-21}	1505.1s	167.0s	462.9s	4210.2s	702.4s	837.4s	-
11	2^{-42}	4.9h	0.2h	0.3h	19.0h	0.4h	0.5h	-	2^{-24}	1.6h	0.5h	0.2h	3.9h	0.6h	0.3h	-
12	2^{-46}	6.8h	0.3h	0.2h	35.6h	0.3h	0.4h	-	2^{-27}	3.5h	0.3h	0.6h	13.1h	1.4h	0.6h	-
13	2^{-50}	10.3h	0.1h	0.2h	54.6h	0.4h	0.4h	-	2^{-30}	7.0h	0.5h	0.4h	26.9h	1.0h	0.7h	-
14	2^{-56}	29.1h	1.0h	0.9h	-	0.7h	1.7h	-	2^{-33}	17.4h	1.4h	0.6h	113.7h	4.5h	2.4h	-
15	2^{-62}	71.3h	2.1h	3.7h	-	2.3h	4.9h	-	2^{-37}	65.9h	21.7h	15.1h	-	54.4h	23.0h	-
16	2^{-70}	-	57.9h	66.5h	-	80.6h	176.1h	-	2^{-41}	-	110.7h	66.6h	-	153.0h	109.2h	-
17	2^{-73}	-	2.0h	4.3h	-	0.5h	2.2h	-	2^{-43}	-	24.9h	8.5h	-	10.3h	35.3h	-
18	2^{-76}	-	0.1h	0.7h	-	0.1h	0.4h	-	2^{-45}	-	2.7h	0.4h	-	6.3h	2.0h	-
19	2^{-81}	-	0.7h	1.6h	-	0.8h	1.6h	-	2^{-47}	-	2.1h	1.8h	-	0.8h	1.5h	-
20	2^{-85}	-	0.4h	1.0h	-	0.4h	0.3h	-	2^{-49}	-	0.2h	0.0h	-	0.4h	0.1h	-
21	2^{-89}	-	0.2h	0.6h	-	0.2h	0.5h	-	2^{-52}	-	0.1h	0.0h	-	0.9h	0.1h	-
22	2^{-94}	-	0.4h	0.9h	-	0.3h	0.7h	-	2^{-54}	-	0.0h	0.0h	-	0.1h	0.1h	-
23	2^{-99}	-	0.4h	1.0h	-	0.8h	0.8h	-	2^{-59}	-	7.8h	1.1h	-	3.7h	5.6h	-
24	2^{-107}	-	10.5h	23.0h	-	37.8h	27.9h	-	2^{-63}	-	35.5h	43.6h	-	-	70.7h	-
25	2^{-112}	-	19.3h	28.2h	-	18.0h	18.8h	-	2^{-66}	-	52.4h	54.1h	-	-	-	-
26	2^{-116}	-	6.0h	2.5h	-	1.8h	2.4h	-	2^{-68}	-	72.2h	6.3h	-	-	-	-
27	2^{-121}	-	8.7h	7.8h	-	5.6h	7.5h	-	2^{-70}	-	3.5h	0.4h	-	-	-	-
Total		128.2h	112.0h	145.4h	125.6h	155.1h	251.2h	-		96.5h	337.0h	200.4h	160.0h	238.4h	252.9h	-

Table 20: Experimental results of SPECK96.

	Differential property								Linear property							
Round	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
2	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
3	2^{-3}	0.1s	0.2s	0.1s	0.2s	0.2s	0.2s	-	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
4	2^{-6}	0.8s	0.8s	0.8s	0.6s	0.6s	0.6s	-	2^{-3}	0.3s	0.2s	0.2s	0.3s	0.4s	0.3s	-
5	2^{-10}	4.4s	2.5s	2.9s	6.2s	5.7s	4.8s	-	2^{-6}	1.4s	1.5s	1.9s	3.1s	1.7s	2.7s	-
6	2^{-15}	31.6s	23.0s	36.8s	46.2s	37.9s	40.1s	-	2^{-9}	15.3s	10.2s	13.0s	18.7s	21.1s	25.3s	-
7	2^{-21}	237.3s	154.2s	280.3s	339.6s	440.4s	325.9s	-	2^{-13}	163.9s	85.7s	162.2s	255.0s	324.8s	307.9s	-
8	2^{-30}	1.9h	1.4h	3.3h	2.6h	2.8h	4.3h	-	2^{-18}	3113.1s	2200.0s	4872.5s	5336.1s	8491.6s	6364.4s	-
9	2^{-39}	38.7h	24.9h	51.9h	92.7h	94.9h	131.6h	-	2^{-22}	6.5h	4.7h	7.1h	13.9h	12.0h	20.1h	-
10	2^{-49}	-	489.1h	-	-	-	-	-	2^{-27}	141.2h	85.1h	144.2h	-	-	-	-
11	-	-	-	-	-	-	-	-	2^{-31}	-	327.9h	-	-	-	-	-
12	-	-	-	-	-	-	-	-	2^{-33}	-	36.2h	-	-	-	-	-
13	-	-	-	-	-	-	-	-	2^{-36}	-	48.5h	-	-	-	-	-
14	-	-	-	-	-	-	-	-	2^{-39}	-	42.6h	-	-	-	-	-
Total		40.6h	515.5h	55.3h	95.4h	97.8h	135.9h	-		148.6h	545.6h	152.6h	15.5h	14.5h	22.0h	-

Table 21: Experimental results of SPECK128.

	Differential property								Linear property							
Round	$\mathrm{Pr}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$	$\mathrm{Cor}_{\mathtt{Opt}}$	$T^{\emptyset}_{\mathrm{CaD}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{CaD}}$	T^{0}_{CaD}	$T^{\emptyset}_{\mathrm{Cry}}$	$T^{\mathcal{R}^{-1}}_{\mathrm{Cry}}$	T^{0}_{Cry}	$T_{\mathtt{MILP}}$
1	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
2	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-	1	0.0s	0.0s	0.0s	0.0s	0.0s	0.0s	-
3	2^{-3}	0.1s	0.2s	0.2s	0.2s	0.2s	0.2s	-	2^{-1}	0.1s	0.1s	0.1s	0.1s	0.1s	0.1s	-
4	2^{-6}	0.9s	1.1s	1.0s	1.2s	0.7s	1.0s	-	2^{-3}	0.3s	0.2s	0.3s	0.4s	0.4s	0.4s	-
5	2^{-10}	7.3s	5.3s	5.5s	9.7s	5.4s	5.6s	-	2^{-6}	2.3s	1.7s	4.4s	3.5s	3.2s	4.6s	-
6	2^{-15}	65.9s	44.5s	54.0s	55.7s	46.8s	55.9s	-	2^{-9}	24.3s	24.5s	36.2s	31.6s	33.5s	41.2s	-
7	2^{-21}	484.8s	331.3s	395.7s	399.7s	706.5s	426.6s	-	2^{-13}	275.5s	339.7s	395.0s	392.0s	262.2s	518.9s	-
8	2^{-30}	2.6h	2.7h	3.7h	4.1h	2.7h	3.4h	-	2^{-18}	1.2h	1.1h	2.0h	2.2h	2.3h	3.6h	-
9	2^{-39}	40.8h	37.4h	68.5h	76.0h	88.9h	63.2h	-	2^{-22}	6.9h	6.0h	9.4h	12.1h	14.1h	25.5h	-
10	-	-	-	-	-	-	-	-	2^{-27}	152.0h	141.0h	157.3h	-	-	-	-
Total		43.6h	40.1h	72.4h	80.2h	91.8h	66.8h	-		160.1h	148.3h	168.9h	14.4h	16.5h	29.3h	-