The Simplest SAT Model of Combining Matsui's Bounding Conditions with Sequential Encoding Method

Senpeng Wang[1,2], Dengguo Feng[1], Bin Hu[2], Jie Guan[2], Tairong Shi[2], and Kai Zhang[2]

[1] State Key Laboratory of Cryptology, Beijing, China, wsp2110@126.com
[2] PLA SSF Information Engineering University, Zhengzhou, China

Abstract. As the first generic method for finding the optimal differential and linear characteristics, Matsui's branch and bound search algorithm has played an important role in evaluating the security of symmetric ciphers. By combining the Matsui's bounding conditions with automatic search models, the search efficiency can be improved. All the previous methods realize the bounding conditions by adding a set of constraints. This may increase the searching complexity of models. In this paper, by using Information Theory to quantify the effect of bounding conditions, we give the general form of bounding conditions that can use all the information provided by Matsui's bounding conditions. Then, a new method of combining bounding conditions with sequential encoding method is proposed. Different from all the previous methods, our new method can realize the bounding conditions by removing the variables and clauses from Satisfiability Problem (SAT) models based on the original sequential encoding method. With the help of some small size Mixed Integer Linear Programming (MILP) models, we build the simplest SAT model of combining Matsui's bounding conditions with sequential encoding method. Then, we apply our new method to search the optimal differential and linear characteristics of some SPN, Feistel, and ARX block ciphers. The number of variables, clauses and the solving time of the SAT models are decreased significantly. And we find some new differential and linear characteristics covering more rounds. For example, the optimal differential probability of the full rounds GIFT128 is obtained for the first time.

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

1 Introduction

Differential [BS90] and linear [Mat93] cryptanalysis are two powerful methods which have been widely used in the security analysis of many symmetric ciphers. The core idea of these methods is to identify the differential or linear trails with high probability or correlation. However, searching the optimal differential or linear trails is not an easy work. At EUROCRYPT 1994, Matsui [Mat94] proposed a branch and bound search algorithm which can be used to identify the optimal differentials with the maximum probability. Matsui's algorithm is one of the most powerful and efficient search tools. However, implementing it needs sophisticated programming skills when taking the cipher-specific optimizations into consideration. In order to meet the demands of security analysis of ciphers, many automatic search methods have been proposed and widely used in the search of numerous distinguishers.

Mixed Integer Linear Programming (MILP) is a kind of optimization or feasibility program whose objective function and constraints are linear, and the variables are restricted to be integers. MILP problem can be solved automatically with MILP solvers such as Gurobi [GRB]. In [WW11,MWGP11], the first automatic search method based on MILP was proposed to evaluate the security of word-oriented block ciphers against differential and linear cryptanalysis. Later, Sun *et al.* [SHS+13,SHW+14] proposed methods for generating inequalities to describe the bit-wise differential or linear characteristics of S-box. Therefore, their models can be used to obtain the minimum number of active S-box and search the best differential and linear characteristics of bit-oriented block ciphers. However, the above methods only work on small size S-box (e.g. 4-bit). At FSE 2017, Abdelkhalek *et al.* [AST+17] put forward the first MILP model for large S-box (e.g. 8-bit). Then, some efficient methods were proposed to generate inequalities of large S-box (e.g. [BC20,Udo21]). For ARX ciphers, Fu *el al.* [FWG+16] built the MILP models for the differential and linear characteristics of modular addition and applied them to search the best differential and linear characteristics for SPECK. Moreover, as a powerful automatic search tool, MILP has been also widely used in other attacks, such as integral attacks [XZBL16,WHG+19], cube attacks [TIHM17], impossible differential attacks [ST17b], and zero-correlation linear attacks [CJF+16].

The Boolean Satisfiability Problem (SAT) is a problem which considers the satisfiability of a given boolean formula. And there are also many SAT solvers, such as CaDiCal [Bie19]. The first automatic search method based on SAT is introduced by Mouha and Preneel [MP13]. Then, at CRYPTO 2015, Kölbl *et al.* [KLT15] used the SAT/SMT solver to find the optimal differential and linear characteristics for SIMON. And at ACNS 2016, Liu *et al.* [LWR16] extended the SAT based automatic search algorithm to search the linear characteristics for ARX ciphers. At FSE 2018, Sun *et al.* [SWW18] built the SAT-based models for differential characteristics and got more accurate differential probability for LED64 and Midori64. Moreover, SAT can be used in searching impossible differential trails [LLL+21] and integral distinguishers [SWW17].

Unlike Matsui's algorithm, the automatic search tools enable cryptanalysts to complete the search of distinguishers without sophisticated programming skills. It brings great convenience to the security evaluation of ciphers. However, when the number of variables or constrains in the model is large, the solver may not return the result within a reasonable time. Therefore, it is of great importance to improve the efficiency of automatic search method. And a lots of work have been done on this issue. We divide them into three main categories.

Reducing the Variables and Constraints in the Model. Although Sasaki and Todo [ST17a] pointed out that the number of inequalities can not strictly determinant the efficiency of solving model, it still has an important impact on the solving time. And a lot of methods have been proposed to reduce the variables and constraints modeling S-box or linear layers [AST+17,BC20,Udo21].

Divide and Conquer Approach. In order to obtain the result of a large model in reasonable time, we can divide it into appropriate parts. In [SHW+14], Sun *et al.* split r-rounds cipher into the two parts (the first r_0 and the last $(r - r_0)$ rounds). Then, they combined them after solving the models of the two parts respectively. At FSE 2019, Zhou *et al.* [ZZDX19] proposed a divide-and-conquer approach which divide the whole searching space according to the number of active S-boxes at a certain round.

Combining Matsui's Bounding Conditions into the Model. Matsui's bounding conditions may reduce the feasible region of the original model. The first method of combining Matsui's branch and bound search algorithm with the MILP based search model is proposed by Zhang *et al.* [ZSCH18]. Later, Sun *et al.* [SWW21] put forward a new encoding method to convert the Matsui's bounding conditions into boolean formulas of SAT model. Both methods are realized by adding the constraints derived from the Matsui's bounding conditions into the original model.

From the perspective of implementation effect, the SAT model combining Matsui's bounding conditions proposed by Sun *et al.* [SWW21] is the best choice at present. This method can obtain the complete bounds (full rounds) on the number of active S-boxes, the differential probabilities and linear correlations for many block ciphers for the first time. The efficiency of automatic search has been greatly improved. Just like the MILP models of combining Matsui's bounding conditions, according to the experiment results in [SWW21], adding more Matsui's bounding conditions may not necessarily improve the efficiency. This may because that all the previous methods realize the bounding conditions by adding a set of constraints. And some added constrains increase the searching complexity of models. Regrettably, there is no relevant theory for us to identify the constrains which have negative effects. By doing a considerable amount of experiments, Sun *et al.* put forward a strategy on how to organise the sets of bounding conditions that potentially achieve better performance. Because this strategy is experimental and lack sufficient theoretical guidance, we cannot really know its performance until completing its application. Therefore, it is meaningful to research the better way of combining Matsui's bounding conditions with the automatic search models and improve the search efficiency.

1.1 Our Contributions

In this paper, we study the properties of Matsui's bounding conditions and the new way of combining Matsui's bounding conditions into the SAT model. The contributions of this paper are classified into the following three parts.

The Properties of Matsui's Bounding Conditions. Although we know that the effect of Matsui's bounding conditions is to reduce the feasible region, no one has been able to describe it accurately. By separating Matsui's bounding conditions from specific ciphers, we use Information Theory to quantify the effect of bounding conditions. Thus, when converting the bounding conditions into other formula, we can evaluate the quality of the transformation. In this way, we give the general form of inequality constraints that can utilize all the information provided by Matsui's bounding conditions.

The Simplest SAT Model of Combining Matsui's Bounding Conditions with Sequential Encoding Method. Different from all the previous methods, we propose a new method which can realize the Matsui's bounding conditions by removing variables and constrains from the SAT model based on sequential encoding method. This will decrease the solving complexity of models. Then, with the help of some small size MILP models, we get the simplest SAT model of combining Matsui's bounding conditions with sequential encoding method which has the least variables and clauses.

Searching the Optimal Differential and Linear Characteristics of Block Ciphers. We apply the simplest SAT model to search the optimal differential and linear characteristics of SPN, Feistel and ARX block ciphers. Compared with the previous method, the number of variables, clauses and the solving time of the SAT models are decreased significantly which can be seen in Table 2. For block ciphers PRESENT, RECTANGLE, GIFT64, LBlock, TWINE, SPECK32, SPECK64, the optimal differential and linear characteristics of the full rounds are obtained which are consistent with the results in [SWW21]. For SPECK48, SPECK96, SPECK128 and GIFT128, we find some new differential and linear characteristics covering more rounds. For example, the optimal differential probability of the full rounds GIFT128 is obtained for the first time. And a comparison of the maximum length of optimal differential and linear trails with previous results are provided in Table 1. For all the above ciphers, our results reach the maximum length of optimal differential and linear trails at present.

Table 1. The comparison of the maximum length of optimal trails

Trail	GIFT128	SPECK48	SPECK96	SPECK128	Ref.
Differential	29	18	10	9	[SWW21]
	40 (Full)	**19**	10	9	Sect. 5
Linear	25	23	14	10	[SWW21]
	27	23	**15**	**11**	Sect. 5

1.2 Outline

This paper is organized as follows: Sect. 2 provides the background of automatic search method based on SAT. In Sect. 3, the properties of Matsui's bounding conditions are studied. In Sect. 4, we propose the simplest SAT model of combining bounding conditions with sequential encoding method. Sect. 5 uses the new method to search the optimal differential and linear characteristics of block ciphers. Sect.6 concludes the paper. And some auxiliary materials are supplied in Appendix.

2 Automatic Search Method Based on SAT

2.1 Boolean Satisfiability Problem

For a formula, if it only consists of boolean variables, operators AND ($\wedge$), OR ($\vee$), NOT ($\bar{\cdot}$) and parentheses, we call it boolean formula. And SAT is the boolean satisfiability problem which considers whether there is a valid assignment to boolean variables such that the formula equals one. If such an assignment exists, the SAT problem is said satisfiable. It was shown that the problem is NP-complete [Coo71]. However, many problem with millions of variables can be solved by modern SAT solvers, such as [Bie19].

For any boolean formula, we can convert it into Conjunctive Normal Form (CNF) denoted as $\bigwedge_{i=0}^{m}\left(\bigvee_{j=0}^{n_i} c_{i,j}\right)$, where $c_{i,j}$ is a boolean variable or constant or the NOT of a boolean variable. And each disjunction $\bigvee_{j=0}^{n_i} c_{i,j}$ is called a clause. Because CNF is a standard input format of SAT solvers. When using SAT to solve a problem, we have to translate it into a model consisted of boolean variables and clauses.

2.2 SAT Models for Some Basic Operations

When we use SAT to search differential or linear characteristics, we should translate the search problem into a series of clauses. And the clauses should describe the propagation properties of differential or linear characteristics through the cipher. Here, we will briefly introduce the SAT models for some basic operations which will be used in this paper. For more information, please refer to [SWW21,LWR16]. And in the following, we use x_0 to denote the most significant bit of the n-bit vector $x = (x_0, x_1, \ldots, x_{n-1}) \in \mathbb{F}_2^n$.

Differential Model 1 (Branching) [SWW21]. *Let $y = f(x)$ be a branching function, where $x \in \mathbb{F}_2$ is the input variable, and the output variables $y = (y_0, y_1, \ldots, y_{n-1}) \in \mathbb{F}_2^n$ is calculated as $y_0 = y_1 = \cdots = y_{n-1} = x$. Then, $(\alpha, \beta_0, \beta_1, \ldots, \beta_{n-1})$ is a valid differential trail of f if and only if it satisfies all the equations in the following:*

$$\left.\begin{array}{l} \alpha \vee \overline{\beta_i} = 1 \\ \overline{\alpha} \vee \beta_i = 1 \end{array}\right\}, 0 \le i \le n-1.$$

Differential Model 2 (Xor) [SWW21]. *Let $y = f(x)$ be a function compressed by an Xor, where $x = (x_0, x_1, \ldots, x_{n-1}) \in \mathbb{F}_2^n$ is the input variables, and the output variable $y \in \mathbb{F}_2$ is calculated as $y = x_0 \oplus x_1 \oplus \cdots \oplus x_{n-1}$.*

When $n = 2$, $(\alpha_0, \alpha_1, \beta)$ *is a valid differential trail of* f *if and only if it satisfies all the equations in the following:*

$$\left.\begin{array}{l}\alpha_0 \vee \alpha_1 \vee \overline{\beta} = 1\\ \alpha_0 \vee \overline{\alpha_1} \vee \beta = 1\\ \overline{\alpha_0} \vee \alpha_1 \vee \beta = 1\\ \overline{\alpha_0} \vee \overline{\alpha_1} \vee \overline{\beta} = 1\end{array}\right\}.$$

When $n \geq 3$*, there are two main methods to model the Xor function. The first method decomposes the* n*-input Xor operation into* $(n-1)$ 2*-input Xor operations by introducing auxiliary boolean variables* $u_0, u_1, \ldots, u_{n-3}$*. Then* $y = f(x)$ *can be represented as the following* 2*-input Xor operations:*

$$\begin{cases} x_0 \oplus x_1 = u_0;\\ x_i \oplus u_{i-2} = u_{i-1}, 2 \leq i \leq n-2;\\ x_{n-1} \oplus u_{n-3} = y.\end{cases}$$

After applying 2*-input Xor model to the* $(n-1)$ 2*-input Xor operations one by one, the model of* n*-input Xor operation can be expressed with* $4 \times (n-1)$ *clauses.*

The second method does not introduce auxiliary boolean variables. Let A *be the set* $\{(a_0, a_1, \ldots, a_n) \in \mathbb{F}_2^{n+1} | a_0 \oplus a_1 \oplus \ldots \oplus a_n = 1\}$*. Then, the differential trail* $(\alpha_0, \alpha_1, \ldots, \alpha_{n-1}, \beta)$ *is valid if and only if it satisfies all the following equations.*

$$(\alpha_0 \oplus a_0) \vee (\alpha_1 \oplus a_1) \vee \cdots \vee (\alpha_{n-1} \oplus a_{n-1}) \vee (\beta \oplus a_n) = 1, (a_0, a_1, \ldots, a_n) \in A.$$

According to [SLR+15], the linear masks propagation model for branching (resp. Xor) operation is the same as the differences propagation model for Xor (resp. branching) operation. Thus, we do not introduce the SAT models for linear mask propagation through branching and Xor operation.

Differential Model 3 (Modular Addition) [SWW21,LWR16]. *Let* $z = f(x, y)$ *be a* n*-bit modular addition operation. Then,* $(\alpha, \beta, \gamma) \in \mathbb{F}_2^{3\times n}$ *is a valid differential trail if and only if it satisfies all the following equations:*

$$\begin{array}{l}\alpha_{n-1} \oplus \beta_{n-1} \oplus \gamma_{n-1} = 0;\\ \left.\begin{array}{l}\alpha_i \vee \beta_i \vee \overline{\gamma_i} \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\ \alpha_i \vee \overline{\beta_i} \vee \gamma_i \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\ \overline{\alpha_i} \vee \beta_i \vee \gamma_i \vee \alpha_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\ \overline{\alpha_i} \vee \overline{\beta_i} \vee \overline{\gamma_i} \vee x_{i+1} \vee \beta_{i+1} \vee \gamma_{i+1} = 1\\ \alpha_i \vee \beta_i \vee \gamma_i \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\ \alpha_i \vee \overline{\beta_i} \vee \overline{\gamma_i} \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\ \overline{\alpha_i} \vee \beta_i \vee \overline{\gamma_i} \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\\ \overline{\alpha_i} \vee \overline{\beta_i} \vee \gamma_i \vee \overline{\alpha_{i+1}} \vee \overline{\beta_{i+1}} \vee \overline{\gamma_{i+1}} = 1\end{array}\right\} 0 \leq i \leq n-2.\end{array}$$

where the Xor operation denoted by $\oplus$ *is symbolic representations which can be converted into CNF formulas with the method in* Differential Model 2 (Xor). *In* *order to model the different probability, we will introduce* $(n-1)$ *binary variables denoted as* $w_0, w_1, \ldots, w_{n-2}$. *When they satisfy the following equations:*

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

the differential probability can be computed as $p(\alpha, \beta, \gamma) = 2^{-\sum_{i=0}^{n-2} w_i}$.

The papers [SWW21,LWR16] have showed the model for the linear correlations through modular addition. Because the most-significant bit of modular addition is a constant value, we can omit this variable. So we give a new linear model for modular addition which is a little different from the previous.

Linear Model 1 (Modular Addition). *For* n*-bit modular addition operation* $z = f(x, y)$*, we denote the two input linear masks as* α *and* β *and the output mask as* γ*. And in order to model the correlation,* $(n-1)$ *binary variables denoted as* $w = (w_0, w_1, \ldots, w_{n-2})$ *are introduced. Then, the correlation of the linear approximation* $(\alpha, \beta, \gamma) \in \mathbb{F}_2^{3\times n}$ *is nonzero if and only if* $(\alpha, \beta, \gamma, w)$ *satisfies all the following equations:*

$$\begin{array}{l}\alpha_0 \oplus \beta_0 \oplus \gamma_0 \oplus w_0 = 0; \\ \alpha_{j+1} \oplus \beta_{j+1} \oplus \gamma_{j+1} \oplus w_j \oplus w_{j+1} = 0, 0 \le j \le n-3; \\ \alpha_0 = \beta_0 = \gamma_0; \\ \left.\begin{array}{l}\alpha_i \vee \overline{\gamma_i} \vee w_{i-1} = 1 \\ \overline{\alpha_i} \vee \gamma_i \vee w_{i-1} = 1 \\ \beta_i \vee \overline{\gamma_i} \vee w_{i-1} = 1 \\ \overline{\beta_i} \vee \gamma_i \vee w_{i-1} = 1\end{array}\right\} 1 \le i \le n-1.\end{array}$$

Then, the linear correlation is computed as $p(\alpha, \beta, \gamma) = 2^{-\sum_{i=0}^{n-2} w_i}$.

For S-box, the paper [SWW18] showed an example of building the differential SAT model of 4-bit S-box. Then, the paper [SWW21] proposed the SAT model of active n-bit S-box. Based on the above two methods, we will show a general method for building SAT model of S-box.

Differential Model 4 (S-box). *For an S-box* $f : \mathbb{F}_2^n \to \mathbb{F}_2^m$*, the differential probability is denoted as* $p(\alpha, \beta)$*, where* $\alpha \in \mathbb{F}_2^n$ *is the input difference and* $\beta \in \mathbb{F}_2^m$ *is the output difference. If the minimal non-zero differential probability of S-box is* 2^{-s}*, where* s *is an integer, we introduce* s *auxiliary variables* $w_0, w_1, \ldots, w_{s-1}$ *satisfying* $w_{i+1} \le w_i, 0 \le i \le s-2$ *to calculate the non-zero differential probability. In order to build the differential SAT model of S-box, we introduce a boolean function as follows:*

$$g(\alpha, \beta, w) = \begin{cases} 1, \text{ if } p(\alpha, \beta) = 2^{-\sum_{i=0}^{s-1} w_i}; \\ 0, \text{ otherwise.}\end{cases}$$

Let A be a set which contains all vectors satisfying $g(a,b,c)=0$ denoted as

$$A=\left\{(a,b,c)\in\mathbb{F}_2^{n+m+s}|g(a,b,c)=0\right\}.$$

Then, the following $|A|$ clauses form a primary differential SAT model of the given S-box

$$\bigvee_{i=0}^{n-1}\left(\alpha_i\oplus a_i^l\right)\vee\bigvee_{j=0}^{m-1}\left(\beta_j\oplus b_j^l\right)\vee\bigvee_{k=0}^{s-1}\left(w_k\oplus c_k^l\right)=1,\left(a^l,b^l,c^l\right)\in A.$$

where $|A|$ is the number of vectors in the set A and $\left(a^l,b^l,c^l\right),0\le l\le|A|-1$ is the l-th vector in the set A.

Note that the solution space of the above $|A|$ clauses about (α,β,γ) is the same as that of the following boolean function:

$$h(\alpha,\beta,\gamma)=\bigwedge_{l=0}^{|A|-1}\left(\bigvee_{i=0}^{n-1}\left(\alpha_i\oplus a_i^l\right)\vee\bigvee_{j=0}^{m-1}\left(\beta_j\oplus b_j^l\right)\vee\bigvee_{k=0}^{s-1}\left(w_k\oplus c_k^l\right)\right)=1.$$

Equivalently, we have

$$h(\alpha,\beta,\gamma)=\bigwedge_{(a,b,c)\in\mathbb{F}_2^{n+m+s}}\left(h(a,b,c)\vee\bigvee_{i=0}^{n-1}\left(\alpha_i\oplus a_i\right)\vee\bigvee_{j=0}^{m-1}\left(\beta_j\oplus b_j\right)\vee\bigvee_{k=0}^{s-1}\left(w_k\oplus c_k\right)\right).$$

This equation is called the product-of-sum representation of h. The issue of reducing the number of clauses is turned into the problem of simplifying the product-of-sum representation of the boolean function. According to [AST+17], we know that this simplification problem can be solved by the Quine-McCluskey (QM) algorithm and Espresso algorithm, theoretically. Although it is also an NP-complete problem, the small-scale problem can be solved by some softwares, such as Logic Friday[3]. After simplification, the SAT model characterising the differential propagation through S-box can be established.

Using the same method of differential SAT model for S-box, the SAT model for linear correlations through S-box can be built easily. Here, we omit it.

2.3 Sequential Encoding Method

When we build SAT model of ciphers, we always aim at getting some cryptographic property such as the number of active S-boxes, the differential probability or the linear correlation. All kinds of these objections can be abstracted as the boolean cardinality constraint $\sum_{i=0}^{n-1}w_i\le m$, where w_i is a boolean variable, and m is a non-negative integer. However, addition over integers is not a natural operation in SAT language, which is not easy to describe with only OR and AND

[3] http://windows.dailydownloaded.com/en/educational-software/student-tools/44924-logic-friday-download-install

operations. The sequential encoding method is one of the best methods which can use relatively small amount of additional variables and a great reduction of clauses to characterise the constraint. Many papers [SWW21,SWW18,LWR16] use the sequential encoding method to convert the constraint into CNF formulas.

When $m = 0$, the cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$ can be translated to n clauses as $\overline{w_i} = 1, 0 \leq i \leq n-1$ which means all variables are zero.

When $m \geq 1$, in order to model constraint $\sum_{i=0}^{n-1} w_i \leq m$, auxiliary boolean variables $u_{i,j}$ $(0 \leq i \leq n-2, 0 \leq j \leq m-1)$ are introduced to return contradiction when the cardinality is larger than m. More specifically, for the partial sum $\sum_{i=0}^{k} w_i = m_k$, the values of the auxiliary boolean variables $u_{k,j}$ $(0 \leq j \leq m-1)$ should satisfy the following equations:

$$u_{k,j} = \begin{cases} 0, \text{ if } m_k \leq j \leq m-1; \\ 1, \text{ if } 0 \leq j \leq m_k - 1. \end{cases}$$

Then, $\sum_{i=0}^{k} w_i = \sum_{j=0}^{m-1} u_{k,j}$, and the sequence $\left\{\sum_{i=0}^{k} w_i | 0 \leq k \leq n-2\right\}$ is non-decreasing. Therefore, the constraint $\sum_{i=0}^{n-1} w_i \leq m$ holds if the following implication predicates are satisfied.

$$\begin{array}{l}
\text{if } w_0 = 1 \text{ then } u_{0,0} = 1 \\
u_{0,j} = 0, 1 \leq j \leq m-1 \\
\left.\begin{array}{l}
\text{if } w_i = 1 \text{ then } u_{i,0} = 1 \\
\text{if } u_{i-1,0} = 1 \text{ then } u_{i,0} = 1 \\
\left.\begin{array}{l}
\text{if } w_i = 1 \text{ and } u_{i-1,j-1} = 1 \text{ then } u_{i,j} = 1 \\
\text{if } u_{i-1,j} = 1 \text{ then } u_{i,j} = 1
\end{array}\right\} 1 \leq j \leq m-1 \\
\text{if } w_i = 1 \text{ then } u_{i-1,m-1} = 0
\end{array}\right\} 1 \leq i \leq n-2 \\
\text{if } w_{n-1} = 1 \text{ then } u_{n-2,m-1} = 0
\end{array}$$

The above predicates can be interpreted as the following $2 \cdot m \cdot n - 3 \cdot m + n - 1$ clauses which are the SAT model for the cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$.

$$\begin{array}{l}
\overline{w_0} \vee u_{0,0} = 1 \\
\overline{u_{0,j}} = 1, 1 \leq j \leq m-1 \\
\left.\begin{array}{l}
\overline{w_i} \vee u_{i,0} = 1 \\
\overline{u_{i-1,0}} \vee u_{i,0} = 1 \\
\left.\begin{array}{l}
\overline{w_i} \vee \overline{u_{i-1,j-1}} \vee u_{i,j} = 1 \\
\overline{u_{i-1,j}} \vee u_{i,j} = 1
\end{array}\right\} 1 \leq j \leq m-1 \\
\overline{w_i} \vee \overline{u_{i-1,m-1}} = 1
\end{array}\right\} 1 \leq i \leq n-2 \\
\overline{w_{n-1}} \vee \overline{u_{n-2,m-1}} = 1
\end{array}$$

2.4 Combining Matsui's Bounding Conditions with Sequential Encoding Method

At EUROCRYPT 1994, Matsui [Mat94] proposed a branch and bound search algorithm which can be used to identify the optimal difference with the maximum probability. Let $P_{ini}(R)$ be the initial estimation for the probability bound achieved by R-round trails. With the knowledge of $P_{opt}(i), 1 \leq i \leq R-1$, where $P_{opt}(i)$ is the maximum probability achieved by i-round trails, a partial trail $\left(\alpha^0, \alpha^1, \ldots, \alpha^r\right)$, $1 \leq r \leq R-1$ covering the first r rounds will never extend to be a better R-round trial if it does not satisfy the following condition:

$$\prod_{i=0}^{r-1} p\left(\alpha^i \rightarrow \alpha^{i+1}\right) \cdot P_{opt}(R-r) \geq P_{ini}(R), \tag{1}$$

where $p\left(\alpha^i \rightarrow \alpha^{i+1}\right)$ is the probability of the i-th round. Therefore, we can give up the partial trail. In this way, the efficiency of search algorithm can be improved greatly.

Let $-\log_2\left(p\left(\alpha^i \rightarrow \alpha^{i+1}\right)\right) = \sum_{j=0}^{\varpi-1} w_j^i$, where $w_j^i, 0 \leq j \leq \varpi-1$ are the boolean variables used to calculate the probability weight of the trail propagation $\alpha^i \rightarrow \alpha^{i+1}$. By define the symbols $n = r \cdot \varpi$ and $w_{(\varpi \times i+j)} = w_j^i$. Then, the Eq. (1) can be rewritten as follows:

$$\sum_{i=0}^{r-1} \sum_{j=0}^{\varpi-1} w_j^i = \sum_{i=0}^{n-1} w_i \leq \log_2\left(P_{opt}(R-r)\right) - \log_2\left(Pr_{ini}(R)\right). \tag{2}$$

Note that the right-hand side of this equation is a constant, and the left-hand side of it matches the probability weight of the trail covering the first r rounds. Generally, all the above bounding conditions can be replaced with an inequality constraint of the following form:

$$\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}, 0 \leq e_1 \leq e_2. \tag{3}$$

Matsui's bounding conditions can be incorporated into automatic search algorithms. In [ZSCH18], Zhang *et al.* incorporated Matsui's bounding conditions into the MILP based automatic search of differential characteristics. Then, Sun *et al.* [SWW21] integrate Matsui's bounding conditions into the SAT method so that the search for optimal differential and linear characteristics can be accelerated. Here, we will introduce the SAT model of combining Matsui's bounding conditions with sequential encoding method.

For the boolean cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$, based on the sequential encoding method, Sun *et al.* realized bounding conditions without claiming any new variables as follows.

Case 1. *Bounding condition* $\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$ *with* $e_1 = 0$ *and* $e_2 < n-1$ *can be modeled by the following* e_2 *clauses:*

$$\overline{w_i} \vee \overline{u_{i-1,m_{e_1,e_2}-1}} = 1, 1 \leq i \leq e_2.$$

Case 2. *Bounding condition* $\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$ *with* $e_1 > 0$ *and* $e_2 < n-1$ *can be modeled by the following* $m - m_{e_1,e_2}$ *clauses:*

$$u_{e_1-1,j} \vee \overline{u_{e_2,j+m_{e_1,e_2}}} = 1, 0 \leq j \leq m - m_{e_1,e_2} - 1.$$

Case 3. *Bounding condition* $\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$ *with* $e_1 > 0$ *and* $e_2 = n-1$ *can be modeled by the following* $2 \cdot (m - m_{e_1,e_2}) + 1$ *clauses:*

$$\begin{cases} u_{e_1-1,j} \vee \overline{u_{n-2,j+m_{e_1,e_2}}} = 1,\ 0 \leq j \leq m - m_{e_1,e_2} - 1; \\ u_{e_1-1,j} \vee \overline{w_{n-1}} \vee \overline{u_{n-2,j+m_{e_1,e_2}-1}} = 1,\ 0 \leq j \leq m - m_{e_1,e_2}. \end{cases}$$

The above method can intermix multiple Matsui's bounding conditions into one SAT problem with an increment on the number of clauses. At the same time, the number of variables remains the same as the original SAT model. According to the experiments, adding all the Matsui's bounding conditions into the SAT model is not the best choice. Thus, Sun *et al.* put forward a strategy on how to organise the sets of bounding conditions that potentially achieve better performance.

3 The Properties of Matsui's Bounding Conditions

We all know that the efficiency of Matsui's algorithm comes from the fact that it can eliminate some impossible solutions and reduce the search space. But, there is no relevant theory which can quantify this effect. In order to make better use of Matsui's bounding conditions, we will researching the properties of them.

3.1 Quantify the Effect of Matsui's Bounding Conditions

With the same mathematical symbols defined in Sect. 2, let $w_i \in \mathbb{F}_2, 0 \leq i \leq n-1$ be the variables which are used to calculate the differential probability or linear correlation of a cipher. Because we want to study the nature of the Matsui's bounding conditions without considering the specific cryptographic algorithm. In order to avoid the influence of the specific cryptographic algorithm, we propose the definition of ideal cryptographic algorithm.

Definition 1. *Let* $W = \{w^i \in \mathbb{F}_2^n, 0 \leq i \leq m-1\}$ *be a cryptographic property vector set and* E *be a cipher. The event that* E *has property* $w^i \in W$ *is denoted as* $E[w^i]$. *And the event that* E *does not has property* $w^i \in W$ *is denoted as* $E[\overline{w^i}]$. *Then,* E *is an ideal cipher of* W *if it satisfies the following conditions:*

(1) For any vector $w^i \in W$, *whether* E *has property* w^i *is random. That is, the probability of* $E[w^i]$ *is* $\frac{1}{2}$, *denoted as* $p\left(E[w^i]\right) = \frac{1}{2}$.

(2) For any two vectors $w^i, w^j \in W, i \neq j$, $E[w^i]$ *is independent with* $E[w^j]$. *That is* $p\left(E[w^i, w^j]\right) = p\left(E[w^i]\right) \times p\left(E[w^j]\right) = \frac{1}{4}$, *where* $E[w^i, w^j]$ *is the event that* E *has the properties* w^i *and* w^j.

If we obtain a Matsui's bounding condition $\sum_{j=e_1}^{e_2} w_j \leq m_{e_1,e_2}$, all the vectors which do not satisfy $\sum_{j=e_1}^{e_2} w_j \leq m_{e_1,e_2}$ are not feasible cryptographic property. Thus, for vector $w^i = \left(w_0^i, w_1^i, \ldots, w_{n-1}^i\right)$ satisfying $\sum_{j=e_1}^{e_2} w_j^i > m_{e_1,e_2}$, we have $p\left(E[w^i]\right) = 0$ and $p\left(E[\overline{w^i}]\right) = 1$. In order to quantify the effect of Matsui's bounding conditions, we introduce the Information Theory of Shannon [Sha48] firstly.

Theorem 1. [Sha48] *For a set of possibilities $P = \{p_0, p_1, \ldots, p_{n-1}\}$, the information produced by P can be measured by $H\left(P\right) = -\sum_{i=0}^{n-1} p_i \log_2^{p_i}$.*

Then, we use this theorem to measure the effect of Matsui's bounding conditions.

Lemma 1. *Let E be an ideal cipher of a cryptographic property vector set $W = \{w^i \in \mathbb{F}_2^n, 0 \leq i \leq m-1\}$ and $C = \{C^0, C^1, \ldots, C^{l-1}\}$ be a bounding conditions set. If there are N vectors of W which do not satisfy all the l conditions in C, the information of $P = \left\{p\left(E[u^0, u^1, \ldots, u^{m-1}]\right) | u^i \in \{w^i, \overline{w^i}\}, 0 \leq i \leq m-1\right\}$ decreased by C is N. And this property is denoted as $H_d\left(P, C\right) = N$.*

Proof. Without considering the bounding conditions, we can apply Definition 1 and Theorem 1 to calculate the information of P as follows:

$$\begin{aligned} H\left(P\right) &= -\sum\nolimits_{u^i \in \{w^i, \overline{w^i}\}, 0 \leq i \leq m-1} p\left(E[u^0, u^1, \ldots, u^{m-1}]\right) \log_2^{p\left(E[u^0, u^1, \ldots, u^{m-1}]\right)} \\ &= -\sum\nolimits_{u^i \in \{w^i, \overline{w^i}\}, 0 \leq i \leq m-1} 2^{-m} \log_2^{2^{-m}} = m. \end{aligned}$$

When considering the l bounding conditions, if a vector w^i doesn't satisfying all the l bounding conditions, it cannot be the feasible cryptographic property. Without losing generality, we denote the N vectors which do not satisfy all the l conditions as $\{w^i | 0 \leq i \leq N-1\}$. Then, we have

$$\begin{cases} p'\left(E[w^i]\right) = 0, & \text{if } 0 \leq i \leq N-1; \\ p'\left(E[\overline{w^i}]\right) = 1, & \text{if } 0 \leq i \leq N-1; \\ p'\left(E[w^i])\right) = \frac{1}{2}, & \text{if } N \leq i \leq m-1; \\ p'\left(E[\overline{w^i}]\right) = \frac{1}{2}, & \text{if } N \leq i \leq m-1. \end{cases} \tag{4}$$

For $P' = \left\{p'\left(E[u^0, u^1, \ldots, u^{m-1}]\right) | u^i \in \{w^i, \overline{w^i}\}, 0 \leq i \leq m-1\right\}$, we have

$$\begin{aligned} H\left(P'\right) &= -\sum\nolimits_{u^i \in \{w^i, \overline{w^i}\}, 0 \leq i \leq m-1} p'\left(E[u^0, u^1, \ldots, u^{m-1}]\right) \log_2^{p'\left(E[u^0, u^1, \ldots, u^{m-1}]\right)} \\ &= -\sum\nolimits_{u^i \in \{w^i, \overline{w^i}\}, N \leq i \leq m-1} 2^{-m+N} \log_2^{2^{-m+N}} = m - N. \end{aligned}$$

The information of P decreased by C is $H_d\left(P, C\right) = H\left(P\right) - H\left(P'\right) = N$. □

When building SAT models, we have to convert the Matsui's bounding conditions into other form of formulas. In the following, we will evaluate the property of the transformation.

Lemma 2. *Let* $P = \left\{ p\left(E[u^0, u^1, \ldots, u^{m-1}] \right) | u^i \in \{w^i, \overline{w^i}\}, 0 \le i \le m-1 \right\}$ *be a cryptographic property possibilities set. If c is a bounding conditions set converted from the bounding conditions set C. Then, we have* $H_d(P, c) \le H_d(P, C)$.

Proof. Let w^i be a vector that satisfies all the bounding conditions in C. Because c is converted from C, w^i should also satisfies all the formulas in c. We have

$$m - H_d(P, C) \le m - H_d(P, c) \Rightarrow H_d(P, c) \le H_d(P, C).$$

□

Corollary 1. *Let c be the bounding conditions set which is converted from the bounding condition set C. When* $H_d(P, c) = H_d(P, C)$*, all the information provided by bounding conditions set C has been fully utilized by c.*

3.2 Further Insights into Matisui's Bounding Conditions

According to Sect. 2.4, Sun *el al.* summarized all the Matsui's bounding conditions as the form of $\sum_{i=e_1}^{e_2} w_i \le m_{e_1,e_2}$. However, when researching the information decreased by the constraints of the form $\sum_{i=e_1}^{e_2} w_i \le m_{e_1,e_2}$, we find that they cannot always utilized all the information provided by Matusui's bounding conditions. We will give an example to show this phenomenon.

For a toy cipher E which has 3 rounds, let $\left(\alpha^0, \alpha^1, \alpha^2, \alpha^3\right)$ be the 3-round trail. By introducing 6 boolean variables $w = \{w_0^{(0)}, w_1^{(0)}, w_0^{(1)}, w_1^{(1)}, w_0^{(2)}, w_1^{(2)}\}$, the probability of round function is calculated as follows:

$$-\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right) = w_0^{(i)} + w_1^{(i)}. \tag{5}$$

Let $P_{opt}(1) = 2^{-1}$, $P_{opt}(2) = 2^{-2}$ and $P_{ini}(3) = 2^{-3}$ be the Matsui's bounding conditions. Then, the vectors satisfying all the above 3 conditions are as follow:

$$\begin{gathered}\{0,1,0,1,0,1\}, \{0,1,0,1,1,0\}, \{0,1,1,0,0,1\}, \{0,1,1,0,1,0\},\\ \{1,0,0,1,0,1\}, \{1,0,0,1,1,0\}, \{1,0,1,0,0,1\}, \{1,0,1,0,1,0\}.\end{gathered}$$

Thus, the information decreased by $\{P_{opt}(1), P_{opt}(2), P_{ini}(3)\}$ is $2^6 - 8 = 56$.

According to Sect. 2.4, all the form of $\sum_{i=e_1}^{e_2} w_i \le m_{e_1,e_2}$ conditions deduced from Matsui's bounding conditions are as follows:

$$\begin{cases} C_0 : -\log_2\left(p\left(\alpha^0 \to \alpha^1\right)\right) \le \log_2\left(P_{opt}(2)\right) - \log_2\left(P_{ini}(3)\right); \\ C_1 : -\sum_{i=0}^{1} \log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right) \le \log_2\left(P_{opt}(1)\right) - \log_2\left(P_{ini}(3)\right); \\ C_2 : -\log_2\left(p\left(\alpha^1 \to \alpha^2\right)\right) \le 2 \cdot \log_2\left(P_{opt}(1)\right) - \log_2\left(Pr_{ini}(3)\right); \\ C_3 : -\sum_{i=1}^{2} \log_2\left(P\left(\alpha^i \to \alpha^{i+1}\right)\right) \le \log_2\left(P_{opt}(1)\right) - \log_2\left(P_{ini}(3)\right); \\ C_4 : -\log_2\left(p\left(\alpha^2 \to \alpha^3\right)\right) \le \log_2\left(P_{opt}(2)\right) - \log_2\left(P_{ini}(3)\right); \\ C_5 : -\sum_{i=0}^{2} \log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right) \le -\log_2\left(P_{ini}(3)\right). \end{cases} \tag{6}$$

Combining Eq. (5) and Eq. (6), we have

$$\begin{cases} C_0' : w_0^{(0)} + w_1^{(0)} \leq 1; \\ C_1' : w_0^{(0)} + w_1^{(0)} + w_0^{(1)} + w_1^{(1)} \leq 2; \\ C_2' : w_0^{(1)} + w_1^{(1)} \leq 1; \\ C_3' : w_0^{(1)} + w_1^{(1)} + w_0^{(2)} + w_1^{(2)} \leq 2; \\ C_4' : w_0^{(2)} + w_1^{(2)} \leq 1; \\ C_5' : w_0^{(0)} + w_1^{(0)} + w_0^{(1)} + w_1^{(1)} + w_0^{(2)} + w_1^{(2)} \leq 3. \end{cases}$$

Then, the 27 vectors that satisfy all the conditions $\{C_0', C_1', C_2', C_3', C_4', C_5'\}$ are as follow:

$$\begin{aligned} &\{0,0,0,0,0,0\}, \{0,0,0,0,0,1\}, \{0,0,0,0,1,0\}, \{0,0,0,1,0,0\}, \{0,0,0,1,0,1\}, \\ &\{0,0,0,1,1,0\}, \{0,0,1,0,0,0\}, \{0,0,1,0,0,1\}, \{0,0,1,0,1,0\}, \{0,1,0,0,0,0\}, \\ &\{0,1,0,0,0,1\}, \{0,1,0,0,1,0\}, \{0,1,0,1,0,0\}, \{0,1,0,1,0,1\}, \{0,1,0,1,1,0\}, \\ &\{0,1,1,0,0,0\}, \{0,1,1,0,0,1\}, \{0,1,1,0,1,0\}, \{1,0,0,0,0,0\}, \{1,0,0,0,0,1\}, \\ &\{1,0,0,0,1,0\}, \{1,0,0,1,0,0\}, \{1,0,0,1,0,1\}, \{1,0,0,1,1,0\}, \{1,0,1,0,0,0\}, \\ &\{1,0,1,0,0,1\}, \{1,0,1,0,1,0\}. \end{aligned}$$

That is, the information decreased by conditions $\{C_0', C_1', C_2', C_3', C_4', C_5'\}$ is $2^6 - 27 = 37$. Therefore, the bounding conditions $\{C_0', C_1', C_2', C_3', C_4', C_5'\}$ do not utilize all the information provided by $\{P_{opt}(1), P_{opt}(2), P_{ini}(3)\}$.

Here, we analyze the reasons for this phenomenon. When using Matsui's branch and bounding algorithm to search R-round optimal trails, we will firstly obtain a partial trail denoted as $\left(\alpha^0, \alpha^1, \ldots, \alpha^r\right)$ covering the first r rounds. Then, we can use Eq. (1) to deduce the bound conditions of the form $\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$. But, it should be noted that all the obtained partial trails are valid. That is, the partial trials should satisfy

$$\sum_{i=0}^{r-1} -\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right) \geq -\log_2\left(P_{opt}(r)\right).$$

Therefore, when combining Matsui's bounding conditions with automatic search algorithm, this kind of bounding conditions should also be considered.

Theorem 2. *For an R-round cipher, the following bounding conditions can utilize all the information provided by $M = \{P_{ini}(R), P_{opt}(i), 1 \leq i \leq R-1\}$.*

$$\left.\begin{aligned} A_{j,r} : \sum_{i=j}^{r} \left(-\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) &\leq \log_2\left(P_{opt}(j)\right) \\ &+ \log_2\left(P_{opt}(R-1-r)\right) - \log_2\left(P_{ini}(R)\right) \\ B_{j,r} : \sum_{i=j}^{r} \left(-\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) &\geq -\log_2\left(P_{opt}(r+1-j)\right) \end{aligned}\right\} \begin{array}{l} 0 \leq j \leq r, \\ r \leq R-1. \end{array}$$

Proof. Let $\left(\alpha^j, \alpha^{j+1}, \ldots, \alpha^{r+1}\right)$ be a feasible partial trail covering $(r+1-j)$ rounds, where $0 \le j \le r \le R-1$. Because of the constraint $P_{opt}\left(r+1-j\right)$, the partial trail should satisfy the following bounding condition:

$$B_{j,r}: \sum_{i=j}^{r} \left(-\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) \ge -\log_2\left(P_{opt}\left(r+1-j\right)\right).$$

Then, due to the constrain of $P_{ini}\left(R\right)$, the partial trail will also not be extended to better R-round trail if the following bounding condition is violated

$$P_{opt}\left(j\right) \cdot \prod_{i=j}^{r} \left(p\left(\alpha^i \to \alpha^{i+1}\right)\right) \cdot P_{opt}\left(R-1-r\right) \le P_{ini}\left(R\right).$$

And the above bounding condition can be converted into

$$\begin{aligned} A_{j,r}: \sum_{i=j}^{r} \left(-\log_2\left(p\left(\alpha^i \to \alpha^{i+1}\right)\right)\right) \le \log_2\left(P_{opt}\left(j\right)\right) + \log_2\left(P_{opt}\left(R-1-r\right)\right) \\ -\log_2\left(P_{ini}\left(R\right)\right). \end{aligned}$$

That is, the bounding conditions $\{A_{j,r}, B_{j,r} | 0 \le j < r \le R-1\}$ is converted from $M = \{P_{ini}\left(R\right), P_{opt}\left(i\right), 1 \le i \le R-1\}$. According to Lemma 2, we have

$$H_d\left(P, \{A_{j,r}, B_{j,r} | 0 \le j \le r \le R-1\}\right) \le H_d\left(P, M\right). \tag{7}$$

Let $\left(\alpha^0, \alpha^1, \ldots, \alpha^R\right)$ be a trail which does not satisfy all the Matsui's bounding conditions in M. If $\left(\alpha^0, \alpha^1, \ldots, \alpha^R\right)$ does not satisfy $P_{ini}\left(R\right)$, it will not satisfy $A_{0,R-1}$. If $\left(\alpha^0, \alpha^1, \ldots, \alpha^R\right)$ satisfies $P_{ini}\left(R\right)$, there is at least a partial trail covering k round that does not satisfy $P_{opt}\left(k\right)$. We denote this partial trail as $\left(\alpha^j, \alpha^{j+1}, \ldots, \alpha^{j+k}\right)$. Then, this partial trail will violate the bounding condition $B_{j,j+k-1}$. So the trail $\left(\alpha^0, \alpha^1, \ldots, \alpha^R\right)$ will not satisfy all the bounding conditions in $\{A_{j,r}, B_{j,r} | 0 \le j < r \le R-1\}$. Therefore, we have

$$H_d\left(P, \{A_{j,r}, B_{j,r} | 0 \le j < r \le R-1\}\right) \ge H_d\left(P, M\right). \tag{8}$$

Combining Eq. (7) and Eq. (8), we have

$$H_d\left(P, \{A_{j,r}, B_{j,r} | 0 \le j < r \le R-1\}\right) = H_d\left(P, M\right).$$

According to Corollary 1, the conditions set $\{A_{j,r}, B_{j,r} | 0 \le j < r \le R-1\}$ utilizes all the information provided by $\{P_{ini}\left(R\right), P_{opt}\left(i\right), 1 \le i \le R-1\}$. □

Using the same mathematical symbols with Eq. (3), we have the following corollary.

Corollary 2. *All the Matsui's bounding conditions can be replaced with inequality constraints of the form* $l_{e_1,e_2} \le \sum_{i=e_1}^{e_2} w_i \le m_{e_1,e_2}$.

4 The Simplest SAT Model of Combining Bounding Conditions with Sequential Encoding Method

Although numerous Matsui's bounding conditions can be obtained, it is not sure which bounding condition can accelerate the solve efficiency of SAT model accurately. With the observations and experiences in the tests, Sun *et al.* [SWW21] put forward a strategy on how to create the sets of bounding conditions that probably achieve extraordinary advances. But this is an experimental and heuristic strategy. It is worth studying how to combine bounding conditions with sequential encoding method in a better way.

4.1 A New Method of Combining Bounding Conditions with Sequential Encoding Method

According to Sec. 2.3, in order to model the cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$, the normal sequential encoding method needs $(n-1)\cdot m$ auxiliary variables, denoted as $u_{i,j}$ $(0 \leq i \leq n-2, 0 \leq j \leq m-1)$. Then, the paper [SWW21] intermix the bounding conditions $\sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$ into the sequential encoding method by adding corresponding clauses. Different from the above strategy, we will propose a new method of intermixing multiple Matsui's bounding conditions into the sequential encoding method by removing some variables and clauses.

From Corollary 2, we know that the more general form of bounding condition is $l_{e_1,e_2} \leq \sum_{i=e_1}^{e_2} w_i \leq m_{e_1,e_2}$. If we get the condition $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$, according to the rules of sequential encoding method, we have

$$u_{e_2,j} = \begin{cases} 0, & \text{if } m_{0,e_2} \leq j \leq m-1, \\ 1, & \text{if } 0 \leq j \leq l_{0,e_2}-1, \\ uncertain, & \text{otherwise.} \end{cases}$$

Therefore, the value of some auxiliary variables are determine. We can omit the variables and clauses which characterise these determined values. Because there are at least $m_{0,e_2} - l_{0,e_2}$ auxiliary variables whose values are uncertain. We have to introduce the boolean variables denoted as $\{u_{e_2,j} | l_{0,e_2} \leq j \leq m_{e_2} - 1\}$ to represent these uncertain values. Then, we can use the following equation to compute the partial sum of $\sum_{i=0}^{e_2} w_i$.

$$\sum_{i=0}^{e_2} w_i = \sum_{j=l_{0,e_2}}^{m_{0,e_2}-1} u_{e_2,j} + l_{0,e_2}.$$

Base on this idea, we propose a new method of combining bounding conditions with sequential encoding method.

Lemma 3. *Let $\sum_{i=0}^{n-1} w_i \leq m, 1 \leq n$ be a cardinality constraint. Based on the sequential encoding method, the following clauses can utilized all the information* *provided by the condition* $l_{0,0} \leq w_0 \leq m_{0,0}$*:*

$$\begin{aligned}
&\textbf{if } l_{0,0} = 0 \textbf{ and } m_{0,0} = 1: \\
&\qquad \overline{w_0} \vee u_{0,0} = 1 \\
&\textbf{if } l_{0,0} = 0 \textbf{ and } m_{0,0} = 0: \\
&\qquad \overline{w_0} = 1 \\
&\textbf{if } l_{0,0} = 1 \textbf{ and } m_{0,0} = 1: \\
&\qquad w_0 = 1
\end{aligned}$$

And this is the simplest model of using the sequential encoding method to characterise the bounding condition $l_{0,0} \leq w_0 \leq m_{0,0}$*.*

Proof. When using original sequential encoding method to model the cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$, we have to introduce m auxiliary boolean variables $u_{0,0}, u_{0,1}, \ldots, u_{0,m-1}$ to represent to the value of partial sum w_0. Different from the method in Sect. 2.4, we can realise the bounding condition $l_{0,0} \leq w_0 \leq m_{0,0}$ by removing variables and clauses as follows.

When $l_{0,0} = 0$ and $m_{0,0} = 1$, only the value of $u_{0,0}$ is uncertain. And all the values of other auxiliary variables $u_{0,1}, u_{0,2}, \ldots, u_{0,m-1}$ are determined. We can remove all these determined variables and related clauses. Then, the value of partial sum w_0 can be represented by the rules of sequential encoding method as $\overline{w_0} \vee u_{0,0} = 1$.

When $l_{0,0} = m_{0,0} = 0$, all the values of auxiliary variables are determined. Thus, no auxiliary variables need to be introduced. And the value of partial sum w_0 can be represented as the clause $\overline{w_0} = 1$.

When $l_{0,0} = m_{0,0} = 1$, all the values of auxiliary variables are determined. Thus, all the auxiliary variables and related clauses can be removed. And the value of partial sum w_0 can be represented as the clause $w_0 = 1$.

In the above three cases, all the introduced auxiliary variables are used to represent the uncertain value and all the clauses are the rules of sequential encoding method to determined the values of variables. They are all necessary which can not be removed. Take $l_{0,0} = m_{0,0} = 1$ as an example, if we remove the clause $w_0 = 1$, the value of w_0 that removed by bounding condition can not be removed. It is contradictory to the state that clauses can use all the information provided by the bounding condition. Therefore, this is the simplest model of using the sequential encoding method to characterise the bounding condition $l_{0,0} \leq w_0 \leq m_{0,0}$. □

Lemma 4. *Let* $\sum_{i=0}^{n-1} w_i \leq m, 3 \leq n$ *be a cardinality constraint. If the bounding condition* $l_{0,e_2-1} \leq \sum_{i=0}^{e_2-1} w_i \leq m_{0,e_2-1}$, $1 \leq e_2 \leq n-2$ *is known, the following* *clauses can utilized all the information provided by* $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$.

$$
\begin{array}{l}
\textbf{if } m_{0,e_2} = 0 : \\
\quad \overline{w_{e_2}} = 1 \\
\textbf{if } m_{0,e_2} > 0 : \\
\quad \textbf{if } l_{0,e_2} = 0 : \\
\quad\quad \overline{w_{e_2}} \vee u_{e_2,0} = 1 \\
\quad\quad \textbf{if } l_{0,e_2-1} < m_{0,e_2-1} : \\
\quad\quad\quad \overline{u_{e_2-1,0}} \vee u_{e_2,0} = 1 \\
\quad \left.\begin{array}{l}
\textbf{if } j = l_{0,e_2-1} : \\
\quad \overline{w_{e_2}} \vee u_{e_2,j} = 1 \\
\textbf{if } j > l_{0,e_2-1} \textbf{ and } j \leq m_{0,e_2-1} : \\
\quad \overline{w_{e_2}} \vee \overline{u_{e_2-1,j-1}} \vee u_{e_2,j} = 1 \\
\textbf{if } j \geq l_{0,e_2-1} \textbf{ and } j \leq m_{0,e_2-1} - 1 : \\
\quad \overline{u_{e_2-1,j}} \vee u_{e_2,j} = 1
\end{array}\right\} max(l_{0,e_2}, 1) \leq j \leq m_{0,e_2} - 1 \\
\quad \textbf{if } m_{0,e_2-1} = m_{0,e_2} \textbf{ and } l_{0,e_2-1} < m_{0,e_2} : \\
\quad\quad \overline{w_{e_2}} \vee \overline{u_{e_2-1,m_{0,e_2}-1}} = 1 \\
\quad \textbf{if } l_{0,e_2-1} = m_{0,e_2} : \\
\quad\quad \overline{w_{e_2}} = 1
\end{array}
\tag{9}
$$

And this is the simplest SAT model of using sequential encoding method to characterise the bounding condition $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$.

Proof. When using original sequential encoding method to model the cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m$, we have to introduce m auxiliary boolean variables $u_{e_2,0}, u_{e_2,1}, \ldots, u_{e_2,m-1}$ to represent to the value of partial sum $\sum_{i=0}^{e_2} w_i$. Different from the method in Sect. 2.4, we can realise the bounding condition $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$ by removing variables and clauses as follows.

When $m_{0,e_2} = 0$, all the values of auxiliary variables are determined. Thus, all the auxiliary variables and related clauses can be removed. And the value of w_{e_2} can be represented as the clauses $\overline{w_{e_2}} = 1$.

When $m_{0,e_2} > 0$, in order to characterise the value of $\sum_{i=0}^{e_2} w_i$, the auxiliary variables $m_{0,e_2} - l_{0,e_2}$ whose values are uncertain must be introduced, denoted as $\{u_{e_2,j} | l_{0,e_2} \leq j \leq m_{0,e_2} - 1\}$. And all the other auxiliary variables whose values are determined can be removed. Then, we use the rules of sequential encoding method to model these variables one by one.

If $l_{0,e_2} = 0$, the value of $u_{e_2,0}$ should satisfy the following rules of sequential encoding method.

$$
\begin{cases}
\text{if } w_{e_2} = 1 \text{ then } u_{e_2,0} = 1; \\
\text{if } u_{e_2-1,0} \text{ is uncertain },\ \text{when } u_{e_2-1,0} = 1 \text{ then } u_{e_2,0} = 1.
\end{cases}
$$

For $max(l_{0,e_2}, 1) \leq j \leq m_{0,e_2} - 1$, the value of $u_{e_2,j}$ should satisfy the following rules of sequential encoding method.

$$\begin{cases} \text{if } u_{e_2-1,j-1} \text{ is determined as 1 and } w_{e_2} = 1 \text{ then } u_{e_2,j} = 1; \\ \text{if } u_{e_2-1,j-1} \text{ is uncertain, when } u_{e_2-1,j-1} = 1 \text{ and } w_{e_2} = 1 \text{ then } u_{e_2,j} = 1; \\ \text{if } u_{e_2-1,j} \text{ is uncertain, when } u_{e_2-1,j} = 1 \text{ then } u_{e_2,j} = 1. \end{cases}$$

Because of the bounding condition $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$ and the rules of sequential encoding method, auxiliary boolean variables $u_{e_2,j}$ will return contradiction when $\sum_{i=0}^{e_2} w_i > m_{0,e_2}$. Thus, the following clauses should be satisfied.

$$\begin{cases} \text{if } m_{0,e_2-1} = m_{0,e_2}, u_{e_2-1,m_{0,e_2}-1} \text{ is uncertain, } w_{e_2} = 1 \text{ then } u_{e_2-1,m_{0,e_2}-1} = 0; \\ \text{if } l_{0,e_2-1} = m_{0,e_2} \text{ then } w_{e_2} = 0. \end{cases}$$

The above predicates can be interpreted as the clauses as Eq. (9). Moreover, because the values of $u_{e_2,j}, l_{0,e_2} \leq j \leq m_{e_2} - 1$ are uncertain. According to the rules of sequential encoding method, all these variables and corresponding clauses should not be omit. Therefore, this is the simplest model of using the sequential encoding method to characterise the bounding condition $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}$. □

Lemma 5. *For cardinality constraint $\sum_{i=0}^{n-1} w_i \leq m, 2 \leq n$, if the bounding condition $l_{0,n-2} \leq \sum_{i=0}^{n-2} w_i \leq m_{0,n-2}$ is known, the following clauses can utilized all the information provided by the condition $l_{0,n-1} \leq \sum_{i=0}^{n-1} w_i \leq m_{0,n-1}$.*

$$\begin{cases} \textbf{if } m_{0,n-1} = 0: \\ \qquad \overline{w_{n-1}} = 1 \\ \textbf{if } m_{0,n-1} > 0: \\ \qquad \textbf{if } m_{0,n-2} = m_{0,n-1} \textbf{ and } l_{0,n-2} < m_{0,n-1}: \\ \qquad\qquad \overline{w_{n-1}} \vee \overline{u_{n-2,m_{0,n-1}-1}} = 1 \\ \qquad \textbf{if } l_{0,n-2} = m_{0,n-1}: \\ \qquad\qquad \overline{w_{n-1}} = 1 \end{cases} \tag{10}$$

And this is the simplest SAT model of using sequential encoding method to characterise the bounding condition $l_{0,n-1} \leq \sum_{i=0}^{n-1} w_i \leq m_{0,n-1}$.

Proof. According to Lemma 3 and 4, we know that the auxiliary variables $u_{n-2,j}, l_{0,n-2} \leq j \leq m_{0,n-2} - 1$ is introduced to describe the value of $\sum_{i=0}^{n-2} w_i$. For the bounding condition $l_{0,n-1} \leq \sum_{i=0}^{n-1} w_i \leq m_{0,n-1}$, we only need to know whether the condition is valid or not. Therefore, no auxiliary variables need to be introduced. Then, the value of w_{n-1} should satisfy the following rules of sequential encoding method.

$$\begin{cases} \text{if } m_{0,n-1} = 0 \text{ then } w_{n-1} = 0; \\ \text{if } l_{0,n-2} < m_{0,n-1} = m_{0,n-2}, w_{n-1} = 1 \text{ then } u_{n-2,m_{0,n-1}-1} = 0; \\ \text{if } m_{0,n-1} > 0, l_{0,n-2} = m_{0,n-1} \text{ then } w_{n-1} = 0. \end{cases}$$

The above predicates can be interpreted as the clauses as Eq. (10). And all these clauses are the rules of sequential encoding method which can not be omit. □

Theorem 3. *Based on the sequential encoding method, the following clauses are the simplest SAT model which can use all the information provide by the bounding conditions* $l_{0,e_2} \leq \sum_{i=0}^{e_2} w_i \leq m_{0,e_2}, 0 \leq e_2 \leq n-1$*:*

$$
\begin{array}{l}
\textbf{if } l_{0,0}=0 \textbf{ and } m_{0,0}=1:\\
\quad \overline{w_0}\vee u_{0,0}=1\\
\textbf{else if } l_{0,0}=m_{0,0}=0:\\
\quad \overline{w_0}=1\\
\textbf{else if } l_{0,0}=1 \textit{ and } m_{0,0}=1:\\
\quad w_0=1\\
\left.\begin{array}{l}
\textbf{if } m_{0,e_2}=0:\\
\quad \overline{w_{e_2}}=1\\
\textbf{if } m_{0,e_2}>0:\\
\quad \textbf{if } l_{0,e_2}=0:\\
\quad\quad \overline{w_{e_2}}\vee u_{e_2,0}=1\\
\quad\quad \textbf{if } l_{0,e_2-1}<m_{0,e_2-1}:\\
\quad\quad\quad \overline{u_{e_2-1,0}}\vee u_{e_2,0}=1\\
\quad\quad\quad \left.\begin{array}{l}
\textbf{if } j=l_{0,e_2-1}\\
\quad \overline{w_{e_2}}\vee u_{e_2,j}=1\\
\textbf{if } j>l_{0,e_2-1} \textbf{ and } j\leq m_{0,e_2-1}\\
\quad \overline{w_{e_2}}\vee\overline{u_{e_2-1,j-1}}\vee u_{e_2,j}=1\\
\textbf{if } j\geq l_{0,e_2-1} \textbf{ and } j\leq m_{0,e_2-1}-1\\
\quad \overline{u_{e_2-1,j}}\vee u_{e_2,j}=1
\end{array}\right\}\begin{array}{l} max(l_{0,e_2},1)\leq j\\ \leq m_{0,e_2}-1\end{array}\\
\quad \textbf{if } m_{0,e_2-1}=m_{0,e_2} \textbf{ and } l_{0,e_2-1}<m_{0,e_2}\\
\quad\quad \overline{w_{e_2}}\vee\overline{u_{e_2-1,m_{0,e_2}-1}}=1\\
\quad \textbf{if } l_{0,e_2-1}=m_{0,e_2}\\
\quad\quad \overline{w_{e_2}}=1
\end{array}\right\}\begin{array}{l}1\leq e_2\\ \leq n-2\end{array}\\
\textbf{if } m_{0,n-1}=0:\\
\quad \overline{w_{n-1}}=0\\
\textbf{if } m_{0,n-1}>0:\\
\quad \textbf{if } m_{0,n-2}=m_{0,n-1} \textbf{ and } l_{0,n-2}<m_{0,n-1}:\\
\quad\quad \overline{w_{n-1}}\vee\overline{u_{n-2,m_{0,n-1}-1}}=1\\
\quad \textbf{if } l_{0,n-2}=m_{0,n-1}:\\
\quad\quad \overline{w_{n-1}}=1
\end{array}
\qquad (11)
$$

Proof. Any bounding condition $l_{0,e_2} \le \sum_{i=0}^{e_2} w_i \le m_{0,e_2}$ belongs to only one case of Lemma 3-5. Therefore, we can integrate them into Eq. (11) which is the simplest SAT model based on sequential encoding method. □

According to Theorem 3, the number of variables and clauses of the simplest SAT model of combining bounding conditions with sequential encoding method is only related to the upper bound and lower bound of partial sum $\sum_{i=0}^{e_2} w_{e_2}, 0 \le e_2 \le n-1$. Specifically, the total number of auxiliary variables needed is $\sum_{i=0}^{n-2} (m_{0,i} - l_{0,i})$. And after checking the generation rules of each clause in Eq. (11), we can easily get the following corollary.

Corollary 3. *For two conditions sets* $\{l_{0,e_2} \le \sum_{i=0}^{e_2} w_i \le m_{0,e_2}, 0 \le e_2 \le n-1\}$ *and* $\{L_{0,e_2} \le \sum_{i=0}^{e_2} w_i \le M_{0,e_2}, 0 \le e_2 \le n-1\}$*, if the inequalities* $l_{0,e_2} \ge L_{0,e_2}$ *and* $m_{0,e_2} \le M_{0,e_2}$ *hold for all* $0 \le e_2 \le n-1$*, when using Theorem 3 to give their SAT models, the numbers of variables and clauses needed to characterise* $\{l_{0,e_2} \le \sum_{i=0}^{e_2} w_i \le m_{0,e_2}, 0 \le e_2 \le n-1\}$ *will not more than those of* $\{L_{0,e_2} \le \sum_{i=0}^{e_2} w_i \le M_{0,e_2}, 0 \le e_2 \le n-1\}$.

4.2 The Algorithm of Building Simplest SAT Model for Matsui's Bounding Conditions

When searching the best trail of R-round ciphers, we know the Matsui's probability bounds $P_{opt}(i), 1 \le i \le R-1$ and the initial estimation for the probability bound of R-round trail $P_{ini}(R)$. According to Theorem 2, we can get a bounding conditions set denoted as C which can utilize all the information provided by $\{P_{ini}(R), P_{opt}(i), 1 \le i \le R-1\}$. According to Corollary 3, if we get all the accurate bounds of partial sum $\sum_{i=0}^{e_2} w_i, 0 \le e_2 \le n-1$ under the constraints of C, then we can get the simplest model of combining Matsui's bounding conditions set with sequential encoding method. In order to get the accurate lower bounds and upper bounds of $\sum_{i=0}^{e_2} w_i, 0 \le e_2 \le n-1$, we will build some MILP models. Here, we give the framework of getting the accurate bounds in Algorithm 1.

For usual ciphers, because the number of variables and constrains in Algorithm 1 is small, the time needed to solve these models is little. Therefore, for all partial sums $\sum_{i=0}^{e_2} w_i, 0 \le e_2 \le n-1$, we can use Algorithm 1 to get their accurate lower and upper bounds. Then, according to Theorem 3, the simplest SAT model of combining Matsui's bounding conditions and sequential encoding method can be obtained. And we can use it to search the best trails of R-round ciphers.

5 Applications to Block Ciphers

In this section, we apply the method for building simplest SAT model of combining Matsui's bounding conditions with sequential encoding method to several block ciphers. And we give a comparison with the primitive method of combining Matsui's bounding conditions with sequential encoding method proposed by Sun *et al.* [SWW21] on the number of variables, clauses and solving time. In order

Algorithm 1 *Bound*$\left(C, w, \sum_{i=0}^{e_2} w_i\right)$

Input: The bounding conditions set C;
The probability weight variables w;
The partial sum $\sum_{i=0}^{e_2} w_i$.
Output: The accurate lower bound l_{0,e_2} and upper bound m_{0,e_2} of $\sum_{i=0}^{e_2} w_i$.

Let $\mathcal{M}_l$ be an empty MILP model
for c in C do
    $\mathcal{M}_l$.addConstr (c)
$\mathcal{M}_l$.setObjective($\sum_{i=0}^{e_2} w_i$,GRB.MINIMIZE)
$l_{0,e_2} = \mathcal{M}_l.\mathit{optimize}()$
Let $\mathcal{M}_m$ be an empty MILP model
for c in C do
    $\mathcal{M}_m$.addConstr (c)
$\mathcal{M}_m$.setObjective($\sum_{i=0}^{e_2} w_i$,GRB.MAXIMIZE)
$m_{0,e_2} = \mathcal{M}_m.\mathit{optimize}()$
return (l_{0,e_2}, m_{0,e_2})

to make the comparison as fair as possible, we implement the two methods on the same platform (a PC with AMD Ryzen 9 5950X 16-Core 3.4G GHz) and the same SAT solver (CaDiCal [Bie19]).

5.1 Description of Some Block Ciphers

SPN Ciphers. PRESENT [BKL$^+$07] has an SPN structure and uses 80- and 128-bit keys with 64-bit blocks through 31 rounds. In order to improve the hardware efficiency, it use a fully wired diffusion layer. RECTANGLE [ZBL$^+$15] is very like PRESENT. It is a 25-round SPN cipher with the 64-bit block size. As an improved version of PRESENT, GIFT [BPP$^+$17] 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.

Feistel Ciphers. LBlock [WZ11] is a lightweight block cipher proposed by Wu and Zhang. The block size is 64 bits and the key size is 80 bits. It employs a variant Feistel structure and consists of 32 rounds. And TWINE [SMMK12] is a 64-bit lightweight block cipher supporting 80- and 128-bit keys. It has the alike structure as LBlock and consists of 36 rounds.

ARX Ciphers. SPECK [BSS$^+$13] is a family of lightweight block ciphers published by National Security Agency (NSA). It adopts ARX structure which takes the modular addition as its nonlinear operation. According to block size, SPECK family of ciphers are composed of SPECK2n, where $n \in \{16, 24, 32, 48, 64\}$.

5.2 The Results of Applications

In order to better illustrate our results, the following notations are introduced.

- M_{sun}: the method proposed by Sun *el al.* [SWW21].

- M_{sim}: the simplest method proposed in Sect. 4.
- $Var,\ Cnf,\ T^{sol}$: the number of variables, clauses and solving time of models.
- $K_{var} = \frac{Var_{sim}}{Var_{sun}}$: The ratio of the total number of variables.
- $K_{cnf} = \frac{Cnf_{sim}}{Cnf_{sun}}$: The ratio of the total number of clauses.
- $K_{sol} = \frac{T^{sol}_{sim}}{T^{sol}_{sun}}$: The ratio of the total solving time of models.
- P_{opt}: the optimal probability of differential trails.
- Cor_{opt}: the optimal correlation of linear trails.

We apply the two methods M_{sun} and M_{sim} to the above SPN, Feistel and ARX ciphers to searching their optimal differential probabilities and linear correlations. The detailed results are shown in Table 4-14 in the Appendix. The comparison of the two methods on the total number of variables, clauses and solving time of models are presented in Table 2. According to the results, our method have greater advantages. Take PRESENT as an example, when searching the optimal differential probabilities of every round from 1 to 31, the total number of variables, clauses and the time of solving SAT models needed by our method is only 7.1%, 11.1% and 36.6% of the method M_{sun}, respectively.

Table 2. The comparison results of the two methods

Cipher	Total round	Property	K_{var}	K_{cnf}	K_{sol}
PRESENT	31 (Full)	differential	7.1%	11.1%	36.6%
		linear	2.0%	4.7%	46.6%
RECTANGLE	25 (Full)	differential	16.2%	20.0%	35.0%
		linear	14.1%	27.4%	94.0%
GIFT64	28 (Full)	differential	8.7%	12.3%	44.8%
		linear	19.0%	24.1%	94.7%
GIFT128	29	differential	19.0%	22.9%	30.7%
	25	linear	24.2%	28.5%	61.2%
LBlock	32 (Full)	differential	18.8%	52.5%	52.0%
		linear	18.0%	31.8%	58.7%
TWINE	36 (Full)	differential	14.4%	19.6%	45.5%
		linear	18.0%	30.8%	60.0%
SPECK32	22 (Full)	differential	23.0%	28.5%	69.0%
		linear	32.8%	43.0%	89.5%
SPECK48	18	differential	22.1%	33.5%	84.0%
	23 (Full)	linear	29.9%	39.5%	67.0%
SPECK64	27 (Full)	differential	18.3%	22.7%	76.5%
		linear	24.9%	34.2%	69.3%
SPECK96	10	differential	49.3%	54.5%	82.7%
	14	linear	47.2%	56.7%	67.8%
SPECK128	9	differential	51.8%	57.8%	90.3%
	10	linear	59.7%	68.3%	71.8%

For PRESENT, RECTANGLE, GIFT64, LBlock, TWINE, SPECK32 and SPECK64, all the optimal differential probabilities and linear correlations of the full-round ciphers have been obtained. For GIFT128, SPECK48, SPECK96 and SPECK128, our method M_{sim} finds some new differential probabilities or linear correlations covering more rounds which are listed Table 3.

Table 3. New optimal differential probabilities and linear correlations

Differential Property					
Cipher	Round	$\log_2^{P_{opt}}$	*Var*	*Cnf*	T^{sol}
GIFT128	30	-193	838882	2119484	1548721.8s
GIFT128	31	-198.415	473100	1176426	137815.9s
GIFT128	32	-204.415	527361	1331711	191841.5s
GIFT128	33	-210.415	523013	1331731	200005.4s
GIFT128	34	-217.415	607170	1550500	242581.9s
GIFT128	35	-224.83	627866	1601828	211591.8s
GIFT128	36	-234.415	947853	2384355	1191166.5s
GIFT128	37	-240.415	642079	1604643	258131.2s
GIFT128	38	-246.415	633699	1596599	313064.2s
GIFT128	39	-253.415	729939	1845704	115049.5s
GIFT128	40	-260.415	644931	1633919	474680.7s
SPECK48	19	-89	68632	177696	1736050.9s
Linear Property					
Cipher	Round	$\log_2^{Cor_{opt}}$	*Var*	*Cnf*	T^{sol}
GIFT128	26	-91	147345	379885	3580030.2s
GIFT128	27	-94	91807	236723	2274569.6s
SPECK96	15	-43	50325	165960	268094.1s
SPECK128	11	-31	55745	175540	939954.9s

6 Conclusion

In this paper, we aim at finding a better way of combining Matsui's bounding conditions with sequential encoding method. By quantifying the effect of bounding conditions, the general form of inequality constraint which can utilized all the information provided by Matsui's bounding conditions are proposed. Because the values of some auxiliary boolean variables in sequential encoding method can be determined, we proposed a new method of integrating bounding conditions into SAT model. Different from the previous methods, our new method can realize the bounding conditions by removing variables and clauses. In order to accelerate the search efficiency, the algorithm for building the simplest SAT model of combining Matsui's bounding conditions with sequential encoding method is proposed. When applying our new method to searching the optimal differential probability and linear correlation of block ciphers, the total number of variables, clauses and solving time of SAT models are decreased. And we find some new differential and linear characteristics covering more round. As a result, we obtain a more efficient search tool.

Because our method of combining bounding condition with sequential encoding method is general, it can be used to search other kinds of distinguishers for ciphers. The wide applications will be done in the future. And for GIFT128, SPECK48, SPECK96 and SPECK128, some optimal differential probabilities or linear correlations of the full round cipher can not be obtained by the existing methods. How to speed up the search of these ciphers is a problem worth studying.

References

[AST+17] Ahmed Abdelkhalek, Yu Sasaki, Yosuke Todo, Mohamed Tolba, and Amr M. Youssef. MILP modeling for (large) s-boxes to optimize probability of differential characteristics. *IACR Trans. Symmetric Cryptol.*, 2017(4):99–129, 2017.

[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.

[Bie19] Armin Biere. Cadical at the sat race 2019. In Marijn Heule, Matti Järvisalo, and Martin Suda, editors, *SAT Race 2019 - Solver and Benchmark Descriptions,Theory and Applications of Satisfiability Testing - SAT 2009*, volume B-2019-1, pages 8–9. University of Helsinki, 2019.

[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 Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 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 Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science*, pages 321–345. Springer, 2017.

[BS90] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. In Alfred Menezes and Scott A. Vanstone, editors, *Advances in Cryptology - CRYPTO '90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*, volume 537 of *Lecture Notes in Computer Science*, pages 2–21. Springer, 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.*, page 404, 2013.

[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.*, page 689, 2016.

[Coo71] Stephen A. Cook. The complexity of theorem-proving procedures. In Michael A. Harrison, Ranan B. Banerji, and Jeffrey D. Ullman, editors, *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*, pages 151–158. ACM, 1971.

[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 Thomas Peyrin, editor, *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, volume 9783 of *Lecture Notes in Computer Science*, pages 268–288. Springer, 2016.

[GRB] Zonghao Gu, Edward Rothberg, and Robert Bixby. Gurobi optimizer. http://www.gurobi.com/.

[KLT15] Stefan Kölbl, Gregor Leander, and Tyge Tiessen. Observations on the SIMON block cipher family. In Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 161–185. Springer, 2015.

[LLL+21] 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. *Sci. China Inf. Sci.*, 64(5), 2021.

[LWR16] Yunwen Liu, Qingju Wang, and Vincent Rijmen. Automatic search of linear trails in ARX with applications to SPECK and chaskey. In Mark Manulis, Ahmad-Reza Sadeghi, and Steve A. Schneider, editors, *Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings*, volume 9696 of *Lecture Notes in Computer Science*, pages 485–499. Springer, 2016.

[Mat93] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

[Mat94] Mitsuru Matsui. On correlation between the order of s-boxes and the strength of DES. In Alfredo De Santis, editor, *Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings*, volume 950 of *Lecture Notes in Computer Science*, pages 366–375. Springer, 1994.

[MP13] Nicky Mouha and Bart Preneel. Towards finding optimal differential characteristics for arx: Application to salsa20. Cryptology ePrint Archive, Report 2013/328, 2013. https://ia.cr/2013/328.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In Chuankun Wu, Moti Yung, and Dongdai Lin, editors, *Information Security and Cryptology - 7th International Conference, Inscrypt 2011, Beijing, China, November 30 - December 3, 2011. Revised Selected Papers*, volume 7537 of *Lecture Notes in Computer Science*, pages 57–76. Springer, 2011.

[Sha48] Claude E. Shannon. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(4):623–656, 1948.

[SHS+13] Siwei Sun, Lei Hu, Ling Song, Yonghong Xie, and Peng Wang. Automatic security evaluation of block ciphers with s-bp structures against related-key differential attacks. In Dongdai Lin, Shouhuai Xu, and Moti Yung, editors, *Information Security and Cryptology - 9th International Conference, Inscrypt 2013, Guangzhou, China, November 27-30, 2013, Revised Selected Papers*, volume 8567 of *Lecture Notes in Computer Science*, pages 39–51. Springer, 2013.

[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 Palash Sarkar and Tetsu Iwata, editors, *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*, volume 8873 of *Lecture Notes in Computer Science*, pages 158–178. Springer, 2014.

[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 Rosario Gennaro and Matthew Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 95–115. Springer, 2015.

[SMMK12] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. Twine: A lightweight block cipher for multiple platforms. In Lars R. Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography, 19th International Conference, SAC 2012, Windsor, ON, Canada, August 15-16, 2012, Revised Selected Papers*, volume 7707 of *Lecture Notes in Computer Science*, pages 339–354. Springer, 2012.

[ST17a] Yu Sasaki and Yosuke Todo. New algorithm for modeling s-box in MILP based differential and division trail search. In Pooya Farshim and Emil Simion, editors, *Innovative Security Solutions for Information Technology and Communications - 10th International Conference, SecITC 2017, Bucharest, Romania, June 8-9, 2017, Revised Selected Papers*, volume 10543 of *Lecture Notes in Computer Science*, pages 150–165. Springer, 2017.

[ST17b] Yu Sasaki and Yosuke Todo. New impossible differential search tool from design and cryptanalysis aspects - revealing structural properties of several ciphers. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *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*, volume 10212 of *Lecture Notes in Computer Science*, pages 185–215, 2017.

[SWW17] Ling Sun, Wei Wang, and Meiqin Wang. Automatic search of bit-based division property for ARX ciphers and word-based division property. In Tsuyoshi Takagi and Thomas Peyrin, editors, *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*, volume 10624 of *Lecture Notes in Computer Science*, pages 128–157. Springer, 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.

[SWW21] Ling Sun, Wei Wang, and Meiqin Wang. Accelerating the search of differential and linear characteristics with the SAT method. *IACR Trans. Symmetric Cryptol.*, 2021(1):269–315, 2021.

[TIHM17] Yosuke Todo, Takanori Isobe, Yonglin Hao, and Willi Meier. Cube attacks on non-blackbox polynomials based on division property. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, volume 10403 of *Lecture Notes in Computer Science*, pages 250–279. Springer, 2017.

[Udo21] Aleksei Udovenko. MILP modeling of boolean functions by minimum number of inequalities. *IACR Cryptol. ePrint Arch.*, page 1099, 2021.

[WHG+19] Senpeng Wang, Bin Hu, Jie Guan, Kai Zhang, and Tairong Shi. Milp-aided method of searching division property using three subsets and applications. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III*, volume 11923 of *Lecture Notes in Computer Science*, pages 398–427. Springer, 2019.

[WW11] Shengbao Wu and Mingsheng Wang. Security evaluation against differential cryptanalysis for block cipher structures. *IACR Cryptol. ePrint Arch.*, page 551, 2011.

[WZ11] Wenling Wu and Lei Zhang. Lblock: A lightweight block cipher. In Javier López and Gene Tsudik, editors, *Applied Cryptography and Network Security - 9th International Conference, ACNS 2011, Nerja, Spain, June 7-10, 2011. Proceedings*, volume 6715 of *Lecture Notes in Computer Science*, 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 Jung Hee Cheon and Tsuyoshi Takagi, editors, *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*, volume 10031 of *Lecture Notes in Computer Science*, 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 Liqun Chen, Mark Manulis, and Steve A. Schneider, editors, *Information Security - 21st International Conference, ISC 2018, Guildford, UK, September 9-12, 2018, Proceedings*, volume 11060 of *Lecture Notes in Computer Science*, pages 101–115. Springer, 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.

Appendix

Table 4. Experimental results of PRESENT

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-2	669	3112	0.1s	667	3059	0.1s
2	-4	668	2659	0.1s	472	2217	0.1s
3	-8	4203	14763	0.2s	2443	10799	0.2s
4	-12	7839	24564	0.3s	3739	15479	0.3s
5	-20	32809	92575	3.7s	14973	53459	2.4s
6	-24	22011	58386	2.2s	8491	29135	1.1s
7	-28	29679	76683	2.4s	9211	32663	1.7s
8	-32	38499	97428	2.8s	9931	36191	1.5s
9	-36	48471	120621	3.0s	10651	39719	1.0s
10	-41	80418	196930	3.9s	8999	31662	1.6s
11	-46	98990	238786	8.1s	14923	52427	2.4s
12	-52	150790	358715	32.4s	28420	97945	9.7s
13	-56	107355	252813	5.4s	18889	64523	3.3s
14	-62	209460	489035	28.9s	35040	118125	16.7s
15	-66	145437	337053	10.0s	22861	76631	3.1s
16	-70	164337	379110	18.8s	22717	78431	2.1s
17	-74	184389	423615	8.3s	22573	80231	2.3s
18	-78	205593	470568	6.4s	22429	82031	2.5s
19	-82	227949	519969	5.1s	8334	29753	1.3s
20	-86	251457	571818	7.1s	8334	30449	1.3s
21	-90	276117	626115	7.6s	8334	31145	1.3s
22	-96	508490	1148645	15.6s	28141	101795	4.0s
23	-100	335511	755283	11.8s	27697	102995	4.6s
24	-106	612280	1374005	33.3s	34129	117935	16.6s
25	-110	400665	896547	17.2s	33397	118559	4.9s
26	-116	725670	1619525	60.0s	40117	134075	36.3s
27	-120	471579	1049907	31.8s	39097	134123	12.5s
28	-124	505167	1123068	20.8s	14034	47405	1.4s
29	-128	539907	1198677	18.2s	13746	47525	2.3s
30	-132	575799	1276734	19.1s	13458	47645	4.9s
31	-136	612843	1357239	18.3s	13170	47765	3.5s
Total		7575051	17154948	403.0s	539417	1895896	147.3s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-1	351	1790	0.6s	351	1758	0.1s
2	-2	382	1977	0.4s	318	1817	0.1s
3	-4	1369	6599	0.7s	983	5634	0.1s
4	-6	2293	9945	0.7s	1391	7754	0.1s
5	-8	3473	13867	0.7s	1799	9874	0.2s
6	-10	4909	18365	1.0s	2207	11994	0.3s
7	-12	6601	23439	1.2s	2615	14114	0.4s
8	-14	8549	29089	1.0s	3023	16234	0.4s
9	-16	10753	35315	1.1s	3431	18354	0.7s
10	-18	13213	42117	1.3s	3839	20474	0.8s
11	-20	15929	49495	1.7s	4247	22594	0.6s
12	-22	18901	57449	2.1s	4655	24714	1.1s
13	-24	22129	65979	2.2s	5063	26834	0.8s
14	-26	25613	75085	2.5s	5471	28954	0.9s
15	-28	29353	84767	2.8s	5879	31074	1.1s
16	-30	33349	95025	2.7s	6287	33194	1.6s
17	-32	37601	105859	5.0s	6695	35314	1.9s
18	-34	42109	117269	3.5s	7103	37434	2.1s
19	-36	46873	129255	5.3s	7511	39554	1.6s
20	-38	51893	141817	5.5s	7919	41674	1.7s
21	-40	57169	154955	3.4s	8327	43794	2.2s
22	-42	62701	168669	6.0s	8735	45914	2.2s
23	-44	68489	182959	6.3s	9143	48034	3.0s
24	-45	74533	197825	7.7s	9551	50154	3.3s
25	-48	80833	213267	8.0s	9959	52274	3.6s
26	-50	87389	229285	8.8s	10367	54394	3.7s
27	-52	94201	245879	8.9s	10775	56514	4.6s
28	-54	101269	263049	8.5s	11183	58634	5.1s
29	-56	108593	280795	9.3s	11591	60754	3.7s
30	-58	116173	299117	10.0s	11999	62874	4.9s
31	-60	124009	318015	14.1s	12407	64994	9.5s
Total		9731820	22048710	133.3s	194824	1027681	62.1s

Table 5. Experimental results of RECTANGLE

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-2	669	2392	2.9s	667	2339	1.1s
2	-4	668	2179	0.4s	472	1737	0.3s
3	-7	2659	8117	0.8s	1491	5486	0.7s
4	-10	4653	13313	1.2s	2129	7678	0.7s
5	-14	11193	30351	1.3s	4501	15503	1.1s
6	-18	16845	43752	1.7s	6085	20039	1.1s
7	-25	50313	125223	7.6s	18281	55018	5.0s
8	-31	60335	145130	15.8s	21455	60545	9.9s
9	-36	63766	150466	18.8s	20654	57228	14.1s
10	-41	80418	187330	23.0s	23402	64540	16.6s
11	-46	98990	228226	70.5s	26150	71852	42.8s
12	-51	119482	273154	103.0s	28898	79164	27.1s
13	-56	141894	322114	227.8s	31646	86476	52.7s
14	-61	166226	375106	140.7s	34394	93788	57.1s
15	-66	192478	432130	256.9s	37142	101100	58.8s
16	-71	220650	493186	203.8s	39890	108412	75.2s
17	-76	250742	558274	354.1s	42638	115724	76.6s
18	-81	282754	627394	242.8s	45386	123036	98.5s
19	-86	316686	700546	287.3s	48134	130348	132.7s
20	-91	352538	777730	406.6s	50882	137660	137.9s
21	-96	390310	858946	479.1s	53630	144972	106.8s
22	-101	430002	944194	497.5s	56378	152284	111.5s
23	-106	471614	1033474	335.0s	59126	159596	175.3s
24	-111	515146	1126786	560.1s	61874	166908	170.5s
25	-116	560598	1224130	621.7s	64622	174220	324.8s
Total		4801629	10683643	4860.6s	779927	2135653	1698.9s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-1	367	1246	1.6s	351	1214	0.9s
2	-2	446	1433	0.7s	318	1273	0.4s
3	-4	1705	4967	1.4s	983	4002	0.7s
4	-6	2997	7769	1.2s	1391	5578	0.8s
5	-8	4673	11147	1.3s	1799	7154	0.7s
6	-10	6733	15101	1.3s	2207	8730	1.0s
7	-13	14268	30114	3.6s	4252	16115	2.5s
8	-16	19731	39396	6.6s	5473	19691	4.5s
9	-19	26058	49926	9.8s	6694	23267	10.8s
10	-22	33249	61704	20.9s	7915	26843	21.6s
11	-25	41304	74730	48.2s	9136	30419	44.1s
12	-28	50223	89004	104.5s	10357	33995	74.6s
13	-31	60006	104526	234.6s	11578	37571	220.5s
14	-34	70653	121296	292.6s	12799	41147	271.6s
15	-37	82164	139314	380.6s	14020	44723	429.5s
16	-40	94539	158580	1073.8s	15241	48299	778.5s
17	-42	71037	118311	368.5s	10435	33506	205.9s
18	-45	119292	197409	507.8s	16162	52415	875.7s
19	-48	134115	220227	1286.6s	17479	56183	1150.2s
20	-51	149802	244293	1312.7s	18796	59951	1081.4s
21	-54	166353	269607	1214.7s	20113	63719	1265.1s
22	-57	183768	296169	1467.1s	21430	67487	1363.2s
23	-60	202047	323979	1781.8s	22747	71255	1745.2s
24	-63	221190	353037	1884.8s	24064	75023	1888.6s
25	-66	241197	383343	5480.7s	25381	78791	5008.5s
Total		1997917	3316628	17487.4s	281121	908351	16446.6s

Table 6. Experimental results of GIFT64

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-1.415	590	2747	0.3s	590	2699	0.2s
2	-3.415	1560	6677	0.3s	1268	5947	0.2s
3	-7	4554	16630	0.5s	2990	12916	0.3s
4	-11.415	11663	36670	3.2s	6281	24437	0.5s
5	-17	28744	81820	15.5s	13678	48259	2.4s
6	-22.415	38950	103956	33.8s	16090	53830	19.4s
7	-28.415	65899	168535	110.9s	24275	78099	66.7s
8	-38	136625	334925	433.1s	49795	147570	343.9s
9	-42	73534	175738	74.6s	23962	69556	25.8s
10	-48	136911	323127	191.0s	38249	112630	62.1s
11	-52	110934	259130	33.0s	26634	79812	43.5s
12	-58	198771	460311	189.2s	42257	128014	54.8s
13	-62	156014	358650	56.6s	29306	90068	20.7s
14	-68	272151	621687	70.7s	46265	143398	60.1s
15	-72	208774	474298	46.8s	31978	100324	5.1s
16	-78	357051	807255	107.8s	28561	86231	38.6s
17	-82	269214	606074	51.2s	27205	85367	13.7s
18	-88	453471	1017015	119.7s	30997	94787	56.1s
19	-92	337334	753978	59.5s	29353	93347	34.6s
20	-98	561411	1250967	133.5s	33433	103343	59.6s
21	-102	413134	918010	82.6s	31501	101327	16.2s
22	-108	680871	1509111	125.7s	35869	111899	75.3s
23	-112	496614	1098170	87.5s	33649	109307	35.5s
24	-118	811851	1791447	239.1s	38305	120455	142.2s
25	-122	587774	1294458	120.8s	35797	117287	40.4s
26	-128	954351	2097975	251.9s	40741	129011	137.8s
27	-132	686614	1506874	155.6s	37945	125267	11.8s
28	-138	1108371	2428695	365.3s	43177	137567	100.2s
Total		9163735	20504930	3160.9s	800151	2512754	1416.4s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	-1	351	1150	1.1s	351	1118	0.8s
2	-2	382	1337	0.3s	318	1177	0.4s
3	-3	637	2245	0.4s	445	1765	0.4s
4	-5	2039	6879	0.8s	1269	4954	0.7s
5	-7	3155	10033	0.8s	1741	6562	0.8s
6	-10	7077	21216	1.5s	3601	12815	1.5s
7	-13	10236	29106	2.3s	4822	16247	2.2s
8	-16	13971	38244	4.5s	6043	19679	3.7s
9	-20	24950	65986	27.2s	10250	31940	18.8s
10	-25	41805	106810	218.3s	16955	49845	182.2s
11	-29	43090	107342	592.1s	16742	47540	460.1s
12	-31	25795	63539	175.1s	8893	25474	166.5s
13	-34	45021	110115	218.2s	13705	39935	215.0s
14	-37	52500	127317	250.5s	14638	42791	208.2s
15	-40	60555	145767	500.8s	15571	45647	345.1s
16	-43	69186	165465	462.0s	16504	48503	344.2s
17	-46	78393	186411	351.7s	17437	51359	357.0s
18	-49	88176	208605	256.1s	18370	54215	221.0s
19	-52	98535	232047	241.0s	19303	57071	330.8s
20	-55	109470	256737	227.0s	20236	59927	214.9s
21	-58	120981	282675	266.9s	21169	62783	338.5s
22	-61	133068	309861	253.0s	22102	65639	307.0s
23	-64	145731	338295	309.1s	23035	68495	310.4s
24	-67	158970	367977	271.8s	23968	71351	225.8s
25	-70	172785	398907	264.5s	24901	74207	456.5s
26	-73	187176	431085	283.2s	25834	77063	260.3s
27	-76	202143	464511	285.6s	26767	79919	262.8s
28	-79	217686	499185	311.7s	27700	82775	237.5s
Total		2113864	4978847	5777.5s	402670	1200796	5473.2s

Table 7. Experimental results of GIFT128

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	*Cnf*	T^{sol}	*Var*	*Cnf*	T^{sol}
1	-1.415	1182	5499	0.2s	1182	5403	0.2s
2	-3.415	3128	13381	0.2s	2548	11931	0.2s
3	-7	11939	42911	0.7s	8057	33693	0.5s
4	-11.415	23375	73502	1.5s	12713	49269	1.4s
5	-17	48201	137955	7.9s	22631	80998	6.9s
6	-22.415	78022	208308	19.7s	32698	108934	17.8s
7	-28.415	131979	337655	98.1s	49363	158179	83.3s
8	-39	305162	746449	3832.1s	115588	337447	2553.6s
9	-45.415	272180	645604	2657.7s	98536	273887	1867.7s
10	-49.415	239761	562598	542.7s	72419	206125	201.9s
11	-54.415	345062	802966	726.5s	87710	256334	115.0s
12	-60.415	483563	1114804	2172.2s	110573	324151	229.8s
13	-67.83	664028	1515923	7202.8s	145314	418180	1015.5s
14	-79	1218318	2747022	154725.1s	316984	856761	29013.6s
15	-85.415	856156	1912402	82353.6s	204874	538803	16675.4s
16	-90.415	833262	1854320	23703.7s	176946	472134	2261.1s
17	-96.415	1095855	2430141	28299.3s	209023	564547	6249.6s
18	-103.415	1416604	3128587	98258.3s	255346	687908	10032.7s
19	-110.83	1597380	3513947	153129.3s	277578	742308	10794.1s
20	-121.415	2729099	5973181	2679475.9s	495133	1285212	544635.4s
21	-126.415	1528822	3334794	128549.4s	272002	699574	29560.2s
22	-132.415	1950067	4246118	87235.3s	314263	818523	19879.2s
23	-139.415	2444925	5311943	159346.3s	272403	971688	48047.9s
24	-146.83	2680964	5811667	222371.8s	394602	1026020	95098.1s
25	-157.415	4447707	9611825	2680211.5s	680957	1731196	1021543.7s
26	-162.415	2431742	5244388	138927.1s	367058	927014	72698.5s
27	-168.415	3046199	6562735	284765.3s	419503	1072499	128264.8s
28	-174.415	3271885	7041002	302579.7s	419187	1080583	143142.4s
29	-181.83	4018764	8637027	454797.7s	490994	1268484	202086.2s
Total		38175331	83568654	7695991.6s	7265067	19127269	2366197.5s
30	-193	-	-	-	838882	2119484	1548721.8s
31	-198.415	-	-	-	464358	1158942	137815.9s
32	-204.415	-	-	-	527361	1331711	191841.5s
33	-210.415	-	-	-	523013	1331731	200005.4s
34	-217.415	-	-	-	607170	1550500	242581.9s
35	-224.83	-	-	-	627866	1601828	211591.8s
36	234.415	-	-	-	947853	2384355	1191166.5s
37	240.415	-	-	-	642079	1604643	258131.2s
38	246.415	-	-	-	633699	1596599	313064.2s
39	253.415	-	-	-	729939	1845704	115049.5s
40	260.415	-	-	-	644931	1633919	474680.7s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	*Cnf*	T^{sol}	*Var*	*Cnf*	T^{sol}
1	-1	703	2302	1.0s	703	2238	0.8s
2	-2	766	2681	0.4s	638	2361	0.5s
3	-3	1277	4501	0.4s	893	3541	0.4s
4	-5	4087	13791	0.9s	2549	9946	1.0s
5	-7	6323	20113	1.0s	3501	13186	1.3s
6	-10	14181	42528	2.1s	7249	25775	1.9s
7	-13	20508	58338	4.8s	9718	32711	4.6s
8	-17	38338	104234	24.0s	17262	54884	25.9s
9	-22	66780	173900	234.0s	29480	87725	224.1s
10	-26	70814	178870	640.3s	29642	84948	721.0s
11	-31	113135	279355	4804.3s	44955	125305	5587.3s
12	-36	142550	345035	28270.0s	54430	147565	25064.7s
13	-38	67573	161991	5045.4s	23083	62978	1329.7s
14	-41	115848	276465	10202.9s	24510	96239	7672.0s
15	-45	178898	423742	15362.0s	49422	137796	15227.4s
16	-48	153843	342028	10751.9s	39427	110063	3818.8s
17	-51	173226	405870	4591.6s	40360	113927	4207.0s
18	-56	328690	765185	19648.9s	74550	207765	20826.5s
19	-59	222738	515616	9483.1s	48706	134603	13455.1s
20	-64	416330	958975	80615.3	88460	242225	63578.4s
21	-68	373878	856594	148642.6s	78746	212388	86316.8s
22	-74	629715	1434747	1931535.4s	134681	355678	1278924.8s
23	-79	589055	1334575	1208961.7s	129035	333305	691225.2s
24	-82	387213	874722	206139.3s	80821	208775	89751.8s
25	-86	560174	1262890	584729.2s	109634	284772	305487.9s
Total		4676643	10859048	4272597.4s	1132455	3090699	2613454.2s
26	-91	-	-	-	147345	379885	3580030.2s
27	-94	-	-	-	91807	236723	2274569.6s

Table 8. Experimental results of LBlock

Differential Property

Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	184	546	0.1s	184	522	0.1s
2	-2	1053	3524	0.2s	1051	3401	0.2s
3	-4	1911	6169	0.2s	1615	5360	0.2s
4	-6	3057	9511	0.2s	2179	7319	0.2s
5	-8	4491	13501	0.3s	2743	9278	0.2s
6	-12	11070	31656	0.5s	6210	20115	0.5s
7	-16	16210	44036	0.7s	8410	25880	0.5s
8	-22	32571	84505	1.8s	16149	46879	1.2s
9	-28	45633	113891	2.8s	21609	59682	1.8s
10	-36	80208	193906	5.0s	36876	97323	3.4s
11	-44	107136	252370	8.5s	47748	121452	5.8s
12	-48	73530	170916	4.0s	29770	75305	2.3s
13	-56	160164	368326	13.3s	60420	151638	9.2s
14	-62	150563	342553	14.1s	53837	133497	10.0s
15	-66	124200	281046	9.2s	40110	100020	6.2s
16	-72	198877	447903	13.4s	58849	147315	11.4s
17	-76	161110	361336	11.7s	43690	109890	7.7s
18	-82	253911	567365	19.0s	63861	161133	12.8s
19	-86	202820	451706	20.8s	47270	119760	13.6s
20	-92	315665	700939	20.7s	68873	174951	14.7s
21	-96	249330	552156	11.7s	50850	129630	6.5s
22	-102	384139	848625	18.2s	73885	188769	11.6s
23	-106	300640	662686	20.5s	54430	139500	9.7s
24	-112	459333	1010423	21.8s	78897	202587	9.7s
25	-115	284202	624243	10.4s	45218	117120	5.7s
26	-121	536886	1177618	22.3s	79926	208453	12.1s
27	-126	499251	1092904	36.3s	72563	188404	16.5s
28	-131	537885	1175710	26.5s	74789	194482	10.8s
29	-135	479895	1047811	17.3s	62455	163690	8.4s
30	-141	720202	1570430	34.3s	90300	236789	9.6s
31	-146	662427	1442272	51.5s	81743	213268	18.5s
32	-151	706821	1537174	39.2s	83969	219346	16.3s
Total		7765375	7187757	456.3s	1460479	3772758	237.3s

Linear Property

Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	176	481	0.1s	176	465	0.1s
2	-1	623	1981	0.1s	607	1918	0.1s
3	-2	1013	3156	0.1s	877	2934	0.1s
4	-3	1499	4524	0.1s	1147	3950	0.1s
5	-4	2081	6052	0.1s	1417	4966	0.1s
6	-6	4353	11893	0.2s	2671	9251	0.2s
7	-8	6051	15376	0.3s	3331	11279	0.3s
8	-11	11098	26227	0.5s	5570	18236	0.5s
9	-14	15038	33227	0.8s	6910	21852	0.8s
10	-18	25040	52116	1.4s	10700	32605	1.3s
11	-22	32780	64771	2.6s	13100	38565	2.2s
12	-24	24027	46129	1.3s	8737	25595	1.2s
13	-27	37802	71199	3.2s	12554	36876	2.3s
14	-30	44718	82487	2.8s	13830	40364	1.9s
15	-33	52210	94607	3.7s	15106	43852	3.5s
16	-36	60278	107559	7.8s	16382	47340	3.7s
17	-37	33647	59590	2.5s	8291	24342	1.7s
18	-40	74694	131375	4.2s	16918	50300	2.4s
19	-42	62541	109018	3.4s	13291	39635	2.4s
20	-45	92562	160043	4.3s	18594	55532	3.1s
21	-47	76662	131575	4.1s	14548	43559	2.3s
22	-50	112350	191527	5.1s	20270	60764	3.1s
23	-52	92223	156244	4.5s	15805	47483	2.4s
24	-55	134058	225827	5.5s	21946	65996	3.6s
25	-56	72217	121220	2.8s	10977	33478	1.8s
26	-59	155194	259627	6.7s	22098	68188	2.1s
27	-62	168926	280835	9.3s	23822	72572	6.9s
28	-65	183234	302875	16.1s	25546	76956	5.2s
29	-66	97669	161024	4.3s	12713	38830	3.4s
30	-69	207826	341795	6.3s	25442	78636	5.7s
31	-72	223670	366075	16.2s	27294	83276	5.7s
32	-74	178917	291859	10.2s	21097	64415	6.2s
Total		2285177	3912294	130.4s	411767	1244010	76.5s

Table 9. Experimental results of TWINE

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	*Cnf*	T^{sol}	*Var*	*Cnf*	T^{sol}
1	0	184	761	0.6s	184	737	0.4s
2	-2	1053	4814	1.0s	1051	4691	1.1s
3	-4	1911	8104	1.1s	1615	7295	1.2s
4	-6	3057	12091	1.1s	2179	9899	1.3s
5	-8	4491	16726	1.1s	2743	12503	1.1s
6	-12	11070	38106	2.0s	6210	26565	1.9s
7	-16	16210	51561	2.1s	8410	33405	2.5s
8	-22	32571	96545	3.6s	16149	58919	3.3s
9	-28	45633	127436	4.1s	21609	73227	4.0s
10	-38	100661	265893	10.9s	47575	147587	8.6s
11	-46	111870	283105	15.2s	51312	149829	11.3s
12	-51	92541	229174	11.0s	38657	111682	7.9s
13	-58	148588	362181	22.8s	56940	163576	20.9s
14	-64	155253	372989	30.1s	55307	157479	15.8s
15	-68	127790	304341	14.3s	40920	117745	9.4s
16	-74	204239	482693	39.5s	59647	172963	28.9s
17	-77	131330	308567	15.0s	34410	101436	7.6s
18	-83	256928	600482	32.3s	61348	183183	17.8s
19	-88	247479	574738	35.2s	55775	166306	27.4s
20	-94	322371	744437	60.4s	68985	205247	21.8s
21	-97	202482	465815	14.0s	39554	119500	7.8s
22	-103	387828	889106	26.3s	70014	214123	12.6s
23	-107	303395	692916	10.5s	51545	158445	5.6s
24	-113	463358	1054586	24.9s	74690	230279	13.5s
25	-116	286598	650531	11.1s	42718	133612	4.6s
26	-122	541247	1225463	17.2s	75383	238483	7.9s
27	-126	417660	943011	18.5s	55500	176085	5.8s
28	-132	629881	1418495	28.5s	60760	189025	6.6s
29	-136	483370	1085931	21.8s	59080	188105	9.2s
30	-142	725235	1625639	54.7s	64580	201525	12.6s
31	-146	553880	1238931	28.3s	62660	200125	12.0s
32	-152	827309	1846895	41.3s	68400	214025	15.1s
33	-155	501770	1118447	22.8s	51418	166572	7.6s
34	-161	930398	2070860	39.1s	56350	178372	6.8s
35	-166	848643	1885174	68.0s	70310	225145	23.7s
36	-172	1051617	2331743	74.8s	76510	239965	21.4s
Total		11169901	25428287	805.3s	1610498	4977660	366.8s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	*Cnf*	T^{sol}	*Var*	*Cnf*	T^{sol}
1	0	176	777	0.6s	176	761	0.3s
2	-1	607	3165	0.7s	607	3102	0.7s
3	-2	941	4932	0.7s	877	4710	0.7s
4	-3	1339	6892	0.8s	1147	6318	0.8s
5	-4	1801	9012	0.7s	1417	7926	0.7s
6	-6	3633	17221	1.2s	2671	14579	1.1s
7	-8	4875	21592	1.4s	3331	17495	1.4s
8	-11	8666	35699	2.3s	5570	27708	1.9s
9	-14	11438	43883	2.4s	6910	32508	2.3s
10	-18	18640	66916	4.0s	10700	47405	3.0s
11	-22	23980	81051	4.7s	13100	54845	3.6s
12	-24	17403	56785	2.7s	8737	36251	2.3s
13	-27	27194	86591	4.7s	12554	52268	3.6s
14	-30	31950	99063	5.0s	13830	56940	4.3s
15	-32	27459	83560	4.0s	10975	45503	2.8s
16	-35	41594	124467	6.2s	15506	64540	4.3s
17	-36	23177	68572	2.9s	7885	33598	1.6s
18	-39	51370	150395	5.3s	16170	70124	3.1s
19	-41	42936	124075	4.5s	12778	55487	3.0s
20	-44	63446	181175	6.4s	17974	77980	4.1s
21	-45	34647	98142	3.1s	9087	40254	1.2s
22	-48	75398	211967	5.2s	18510	83308	3.1s
23	-50	61869	172270	4.1s	14581	65471	2.2s
24	-53	89906	248123	5.8s	20442	91420	3.9s
25	-54	48421	132832	3.4s	10289	46910	2.0s
26	-57	104034	283779	5.6s	20850	96492	2.6s
27	-59	84258	228145	5.0s	16384	75455	3.2s
28	-62	120974	325311	8.0s	17851	79979	3.5s
29	-63	64499	172642	3.7s	11491	53566	2.2s
30	-66	137278	365831	7.8s	12549	56742	3.4s
31	-68	110103	291700	5.4s	12619	57946	2.5s
32	-71	156650	412739	7.0s	13707	61182	3.7s
33	-72	82881	217572	4.4s	12693	60222	2.3s
34	-75	175130	458123	7.4s	13847	63590	3.7s
35	-77	139404	362935	5.8s	13885	64730	2.9s
36	-80	196934	510407	9.4s	15069	68158	3.2s
Total		2085011	5758341	152.1s	396769	1775473	91.2s

Table 10. Experimental results of SPECK32

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	79	294	0.5s	79	279	0.1s
2	-1	281	1229	1.9s	281	1170	0.1s
3	-3	783	3154	2.1s	691	2837	0.2s
4	-5	1368	5002	1.7s	1000	3995	0.2s
5	-9	3925	12826	2.6s	2535	9285	0.6s
6	-13	6465	19176	3.4s	3665	12425	1.8s
7	-18	11838	32782	9.3s	6050	19264	6.7s
8	-24	20349	53299	55.2s	9653	28875	41.9s
9	-30	28511	71702	417.5s	12565	35903	299.9s
10	-34	26350	64751	484.3s	10340	29245	248.0s
11	-38	32265	78226	805.1s	11095	31635	764.8s
12	-42	38780	92976	1211.5s	11850	34025	852.1s
13	-45	36328	86427	680.1s	9704	28376	292.8s
14	-49	52565	124216	1071.1s	12495	37085	698.4s
15	-54	73638	172510	2213.9s	16646	48856	878.3s
16	-58	70840	164726	1368.0s	15160	44165	690.1s
17	-63	97188	224542	4808.5s	19844	57352	3472.7s
18	-69	130424	299069	32243.4s	26796	75411	20902.7s
19	-74	127386	290218	101072.8s	25982	71704	58801.5s
20	-77	94186	213859	20315.6s	17642	49148	15312.9s
21	-81	129125	292506	35272.9s	21855	61925	36305.5s
22	-85	141865	320456	31015.1s	22385	63865	21320.6s
Total		1124539	2623946	233056.3s	258313	746825	160891.4s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	111	455	0.1s	111	440	0.1s
2	0	190	924	0.1s	190	879	0.1s
3	-1	582	2855	0.1s	582	2722	0.1s
4	-3	1398	6232	0.2s	1306	5783	0.2s
5	-5	2169	8788	0.2s	1801	7604	0.3s
6	-7	3120	11749	0.5s	2296	9425	0.5s
7	-9	4251	15115	1.1s	2791	11246	0.8s
8	-12	7654	25655	3.8s	4614	17884	3.9s
9	-14	7455	23863	10.8s	4081	15482	6.1s
10	-17	12526	38639	46.1s	6334	23532	28.8s
11	-19	11559	34591	48.4s	5371	19718	37.6s
12	-20	8941	26418	17.0s	3673	13886	30.0s
13	-22	15399	44977	41.7s	5695	22034	25.9s
14	-24	17835	51268	12.8s	6145	23765	26.4s
15	-26	20451	57964	15.8s	6595	25496	23.2s
16	-28	23247	65065	38.9s	7045	27227	35.7s
17	-30	26223	72571	62.2s	7495	28958	31.7s
18	-34	50310	136821	1315.2s	14570	53795	622.0s
19	-36	34419	92200	1346.1s	9889	35396	1578.2s
20	-38	38025	101101	2133.8s	10249	36947	1549.7s
21	-40	41811	110407	1227.7s	10609	38498	1518.5s
22	-42	45777	120118	1185.8s	10969	40049	1199.1s
Total		373453	1047776	7508.3s	122411	460766	6718.7s

Table 11. Experimental results of SPECK48

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	119	446	0.1s	119	423	0.1s
2	-1	425	1869	0.3s	425	1778	0.1s
3	-3	1191	4810	0.5s	1051	4325	0.2s
4	-6	2966	10551	0.8s	2214	8492	0.3s
5	-10	6575	20761	1.9s	4215	14875	1.3s
6	-14	10590	30741	6.4s	5870	19545	2.9s
7	-19	19110	52168	23.2s	9494	29980	18.4s
8	-26	37868	97805	174.1s	17836	52472	155.2s
9	-33	54112	133941	1764.7s	24176	67280	2170.1s
10	-40	72932	175413	15030.6s	30516	82088	15476.6s
11	-45	69234	163648	18668.9s	26174	69748	19057.5s
12	-49	69125	161871	11095.6s	22805	61465	9322.1s
13	-54	97908	227464	20309.8s	28712	78076	16776.3s
14	-58	95090	219421	4787.1s	24920	68405	3966.3s
15	-63	131550	301768	22354.6s	31250	86404	14627.8 s
16	-68	151335	345052	31069.3s	33527	92578	17658.7s
17	-75	233120	527877	214052.9s	50800	137776	198543.5s
18	-82	269972	606885	692164.7s	59716	157736	568723.1s
Total		1323222	3082491	1031574.8s	373820	1033446	866500.5s
19	-89	-	-	-	68632	177696	1736050.9s
Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	167	695	0.1s	167	672	0.1s
2	0	286	1412	0.2s	286	1343	0.1s
3	-1	878	4367	0.4s	878	4162	0.2s
4	-3	2118	9544	0.4s	1978	8855	0.3s
5	-6	4624	18411	0.6s	3872	15988	0.5s
6	-8	5163	18832	1.4s	3757	14981	1.0s
7	-12	12405	41821	21.8s	8195	30970	10.4s
8	-15	13882	43731	79.5s	8266	29900	63.8s
9	-19	23105	69231	1190.9s	12595	44030	1303.9s
10	-22	23730	68419	3425.5s	11786	40348	3016.2s
11	-25	29116	81827	13328.0s	13100	44684	12381.6s
12	-28	35054	96431	23989.9s	14414	49020	21814.9s
13	-30	30711	83281	11245.4s	11353	39134	8996.6s
14	-33	47302	126663	36999.4s	15958	55532	28682.4s
15	-37	69365	182556	144397.5s	22555	76760	131326.2s
16	-39	47694	124006	105626.1s	14476	49223	90098.0s
17	-43	90305	232291	449659.4s	25945	87810	382129.5s
18	-45	61086	155641	310300.9s	16510	55853	154346.7s
19	-48	90332	228663	205234.8s	22604	77364	139713.7s
20	-51	100594	252651	184329.7s	24010	81884	62696.2s
21	-54	111408	277835	782536.3s	25416	86404	543774.3s
22	-57	122774	304215	1208767.8s	26822	90924	806042.5s
23	-59	100227	247261	189832.5s	20383	70010	74138.0s
Total		1022326	2669784	3670966.4s	305326	1055851	2460536.3s

Table 12. Experimental results of SPECK64

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	159	598	0.1s	159	567	0.1s
2	-1	569	2509	0.3s	569	2386	0.1s
3	-3	1599	6466	0.4s	1411	5813	0.2s
4	-6	3990	14199	1.0s	2982	11436	0.5s
5	-10	8855	27961	2.7s	5695	20075	2.4s
6	-15	17679	50812	15.8s	10079	32782	11.0s
7	-21	32319	86556	78.0s	16779	50841	78.6s
8	-29	62991	159427	1414.9s	30945	87369	1382.8s
9	-34	58056	142108	1954.5s	25640	70444	1665.8s
10	-38	60690	146291	1266.6s	23050	63905	1971.9s
11	-42	73545	175406	518.8s	24065	67775	551.8s
12	-46	87640	207156	524.9s	25080	71645	333.0s
13	-50	102975	241541	685.1s	26095	75515	508.7s
14	-56	170401	396040	1793.7s	40943	117103	1458.5s
15	-62	202055	464969	7300.5s	48083	133931	7970.2s
16	-70	308286	702316	171274.8s	75378	202569	124237.1s
17	-73	157152	355875	3821.9s	36120	96728	3618.4s
18	-76	173082	391331	2644.1s	33922	93812	1200.3s
19	-81	288162	649648	2777.5s	51086	143332	1894.5s
20	-85	266705	599311	2356.3s	43945	124025	1623.7s
21	-89	293045	656946	1274.9s	43875	125725	1064.8s
22	-94	386793	864742	1874.5s	54593	156958	1809.3s
23	-99	425742	948952	4186.1s	58454	166876	3068.2s
24	-107	709857	1575649	53855.4s	103395	285009	43906.8s
25	-112	523152	1156936	40157.5s	78776	211876	36288.7s
26	-116	471520	1040961	13957.4s	66400	179905	9877.2s
27	-121	610170	1344904	62226.7s	80786	220300	43099.3s
Total		5497189	12409610	375954.0s	1008305	2818702	287617.3s

Linear Property							
Round	$\log_2 Cor_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	223	935	0.1s	223	904	0.1s
2	0	382	1900	0.2s	382	1807	0.2s
3	-1	1174	5879	0.3s	1174	5602	0.2s
4	-3	2838	12856	0.4s	2650	11927	0.3s
5	-6	6208	24811	1.4s	5200	21556	1.0s
6	-9	9622	34583	3.9s	7102	27676	3.2s
7	-13	17765	58536	55.2s	11785	43300	40.1s
8	-17	25205	77401	452.1s	15135	52885	440.0s
9	-19	19497	57676	787.2s	10267	35840	417.7s
10	-21	23502	68269	161.9s	10732	38513	231.5s
11	-24	37852	107623	570.3s	15604	56260	377.1s
12	-27	45730	127067	742.3s	17506	62380	577.2s
13	-30	54352	148123	2064.8s	19408	68500	1918.9s
14	-33	63718	170791	2508.2s	21310	74620	2327.5s
15	-37	93445	246156	58259.1s	30165	103220	23275.5s
16	-41	109565	283621	246564.5s	34755	115285	168923.4s
17	-43	74577	191080	15661.2s	22039	73280	12542.8s
18	-45	82302	209857	2447.7s	21760	74465	1987.3s
19	-47	90399	229471	2184.9s	21481	75650	2085.4s
20	-49	98868	249922	549.0s	21202	76835	643.7s
21	-52	144912	364211	108.0s	29192	106612	96.8s
22	-54	118965	297418	51.0s	22489	82889	32.2s
23	-59	263694	653938	2745.4s	50606	180502	2440.3s
24	-63	246015	603951	136591.4s	50065	169085	106329.4s
25	-66	215848	526530	151105.9s	43424	144324	112890.8s
26	-68	174399	423994	55103.8s	32791	110429	30689.2s
27	-70	186123	451606	1843.6s	31861	110312	1640.3s
Total		2207180	5628206	677816.7s	550308	1924658	469908.9s

Table 13. Experimental results of SPECK96

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	239	902	0.8s	239	855	0.1s
2	-1	857	3789	1.9s	857	3602	0.1s
3	-3	2415	9778	2.6s	2131	8789	0.2s
4	-6	6038	21495	4.2s	4518	17324	0.7s
5	-10	13415	42361	6.4s	8655	30475	3.4s
6	-15	26799	77020	24.4s	15359	49870	22.7s
7	-21	49007	131244	163.8s	25627	77497	230.4s
8	-30	108025	272406	5512.4s	54445	151910	5358.7s
9	-39	159420	384536	149122.6s	76920	202360	145998.0s
10	-49	243782	570615	1628937.4s	111848	283107	1323894.2s
Total		609997	1514146	1783776.5s	300599	825789	1475508.5s
Linear Property							
Round	$\log_2^{Cor_{opt}}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	335	1415	0.1s	335	1368	0.1 s
2	0	574	2876	0.1s	574	2735	0.1 s
3	-1	1766	8903	0.2s	1766	8482	0.2s
4	-3	4278	19480	0.2s	3994	18071	0.2s
5	-6	9376	37611	1.3s	7856	32692	1.1 s
6	-9	14550	52439	12.6s	10750	42012	10.5 s
7	-13	26885	88776	200.6s	17865	65780	180.4 s
8	-18	46923	143128	4483.5s	28679	98698	4025.3 s
9	-22	53435	154236	36875.5s	29685	98220	25305.9s
10	-27	83859	232396	457549.1s	42863	137626	387357.7s
11	-31	88445	237556	936813.7s	41505	130660	624957.1s
12	-33	62940	166486	129785.2s	26008	83255	50454.2s
13	-36	96992	253923	158613.9s	35328	115844	87626.0s
14	-39	112318	290559	161359.4s	37094	122908	97441.6s
Total		602676	1689784	1885695.4s	284302	958351	1277360.4s
15	-43	-	-	-	50325	165960	268094.1s

Table 14. Experimental results of SPECK128

Differential Property							
Round	$\log_2 P_{opt}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	319	1206	0.1s	319	1143	0.1s
2	-1	1145	5069	0.1s	1145	4818	0.1s
3	-3	3231	13090	0.3s	2851	11765	0.3
4	-6	8086	28791	1.0s	6054	23212	0.7s
5	-10	17975	56761	3.5s	11615	40875	4.2
6	-15	35919	103228	36.7s	20639	66958	30.3
7	-21	65695	175932	343.8s	34475	104153	286.3
8	-30	144825	36520	9874.4s	73325	204390	9773.7s
9	-39	213740	365206	274980.4s	103720	272600	247510.6s
Total		490935	1264859	285240.3s	254143	729914	257606.3s
Linear Property							
Round	$\log_2^{Cor_{opt}}$	M_{sun}			M_{sim}		
		Var	Cnf	T^{sol}	Var	Cnf	T^{sol}
1	0	447	1895	0.1s	447	1832	0.1s
2	0	766	3852	0.2s	766	3663	0.1s
3	-1	2358	11927	0.2s	2358	11362	0.2s
4	-3	5718	26104	0.4s	5338	24215	0.3s
5	-6	12544	50411	3.6s	10512	43828	2.9s
6	-9	19478	70295	23.2s	14398	56348	18.1s
7	-13	36005	119016	463.5s	23945	88260	308.5s
8	-18	62859	191896	10263.7s	38471	132490	8422.5s
9	-22	71595	206796	11079.5s	39845	131900	8468.6s
10	-27	112371	311596	355623.7s	57551	184858	253834.6s
Total		324141	993788	377458.1s	193631	678756	271055.9s
11	-31	-	-	-	55745	175540	939954.9s