Improved Automatic Search Tool for Bit-Oriented Block Ciphers and Its Applications

Lingchen Li[1,2](✉), Wenling Wu[1], and Lei Zhang[1]

[1] Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
{lilingchen,wwl}@tca.iscas.ac.cn
[2] University of Chinese Academy of Sciences, Beijing 100049, China

Abstract. The tool based on Mixed-integer Linear Programming (MILP) is simple and effective that frequently used in searching some different types of distinguishers recently. In this paper, we mainly focus on the automatic search method using MILP and the optimizer Gurobi for bit-oriented block ciphers.

We introduce the OPB file format to construct MILP models for the bit-oriented block ciphers. Compared to the LP file format, it is more concise and suitable to deal with boolean variables. And we modify the high-level strategy to reduce the solution time by setting parameter MIPFocus provided by the optimizer Gurobi. Moreover, the new simple linear inequalities of differential pattern propagation of modular addition are given without considering the differential probability in the impossible differential search. As applications, we give the exact lower bounds of the number of differential active s-boxes for 5∼12 rounds LBlock in the related-key model and all of impossible differentials limited the input and output differences to only 1 active bit for the full versions of SPECK.

Keywords: Related-key differentials · Impossible differentials
LBlock · SPECK · MILP

1 Introduction

Finding different types of distinguishers is the key step to evaluate the security of block ciphers. The automatic search methods are the main choices. Most of the early automatic search methods were based on special algorithms implemented from scratch in general purpose programming language. This kinds of methods may be more efficient in some specific cases but they are much more difficult to implement. Recently, the search problem is described as an SAT, MILP, or CP models which can be automatically solved with the corresponding solvers. Among them, the automatic search method based on MILP is simple and practical which has become a popular tool.

The MILP method was first proposed by Mouha *et al.* [1] which used for counting minimum number of differential (or linear) active s-boxes for word-oriented block ciphers. In Asiacrypt 2014, Sun *et al.* [2] proposed a extended

S. Qing et al. (Eds.): ICICS 2017, LNCS 10631, pp. 502–508, 2018.
https://doi.org/10.1007/978-3-319-89500-0_43

framework for bit-oriented block ciphers. The key idea of [2] is to exact the inequalities from the H-representation of the convex hull of all possible differential patterns of the s-box. The linear inequalities describing the differential properties of up to 5-bit s-boxes can be obtained by using the SAGE [3] software and a greedy algorithm. Recently, the tools using the MILP method to searching integral distinguishers [4] based on division property and impossible differentials [5] have also been proposed. Usually, the MILP instances are be described with the LP format and solved with the optimizer Gurobi [6] which is the most efficient commercial solver currently. The MILP method is powerful, but there are some inherent drawbacks. In this paper, we mainly simplify the scale of MILP models and accelerate the search of (related-key) differential characteristics and impossible differentials for bit-oriented block ciphers.

Our Contributions. We proposes the OPB file format to describe the MILP models. Compared to the LP file format, this is more concise and more suitable for constructing models for bit-oriented block ciphers. By setting the parameter MIPFocus of Gurobi reasonably, the solution time can be greatly reduced. For the impossible differentials search, we give the simply linear inequalities of differential propagation of the modular addition without considering the differential probability. This helps reduce the number of variables and constraints in MILP models and speed up searches. As applications, we give the exact lower bounds of the number of related-key differential active s-boxes for LBlock and the impossible differentials for the SPECK family.

Organization. The remainder of the paper is organized as follows. In Sect. 2, we give a brief introduction to the automatic search tools based on MILP and Gurobi for bit-oriented block ciphers. And then we propose some techniques to improve the tools of (related-key) differentials and impossible differentials. As applications, we search the exact lower bounds of the number of related-key differential active s-boxes of LBlock and the impossible differentials of the SPECK family in Sect. 3. We conclude in Sect. 4.

2 The Automatic Search Tool Based on MILP and Gurobi

2.1 The (Related-Key) Differential Automatic Search Method for Bit-Oriented Block Ciphers

In this section, we give a brief introduction of Sun *et al.* [2] framework to find the exact lower bounds of the number of (related-key) differential active s-boxes for bit-oriented block ciphers. The details as follow.

Objective Function. We need introduce a 0–1 variable A_i to mark every s-box in the encryption process and the key schedule algorithm, such that:

$$A_i = \begin{cases} 1, if\ the\ input\ word\ of\ the\ sbox\ is\ nonzero \\ 0, otherwise \end{cases} \tag{1}$$

So, the objective function is $\sum_i A_i$.

Constraints. For the XOR operation, the bit-level input differences are a, b and the bit-level output difference is c. Then the constraints are:

$$\begin{cases} d_{\oplus} \geq a, d_{\oplus} \geq b, d_{\oplus} \geq c \\ a+b+c \geq d_{\oplus} \\ a+b+c \leq 2 \end{cases} \tag{2}$$

where $d_{\oplus}$ is a dummy variable.

For the $w \times v$ s-box marked by A_i, the input difference is $(x_{i0}, x_{i1}, \cdots, x_{i(w-1)})$, the output difference is $(y_{i0}, y_{i1}, \cdots, y_{i(v-1)})$, then:

$$\begin{cases} A_t - x_{ik} \geq 0, k \in \{0, \cdots, w-1\} \\ -A_t + \sum_{j=0}^{w-1} x_{ij} \geq 0 \end{cases} \tag{3}$$

For an bijective s-box, we have:

$$\begin{cases} w \sum_{j=0}^{v-1} y_{ij} - \sum_{j=0}^{w-1} x_{ij} \geq 0 \\ v \sum_{j=0}^{w-1} x_{ij} - \sum_{j=0}^{v-1} y_{ij} \geq 0 \end{cases} \tag{4}$$

In order to make use of the differential distribution table of the s-box, Sun *et al.* used the inequality-generator() function in the sage.geometry.polyhedron class of the SAGE software to obtain the convex hull of all possible differential patterns of the s-box. The number of this linear inequalities can be effectively reduced by using the greedy algorithm. After defining the objective function and the constraints, we need to construct a MILP instances in the LP format. Then we can employ the optimizer Gurobi to solve the MILP instances.

2.2 Improved Tools for Bit-Oriented Block Ciphers

We propose a new file format OPB to describe the MILP models for bit-oriented block ciphers. Gurobi can solve a variety of file format models, such as MPS, LP, OPB and so on. Among them, the MPS format is the most common, the LP format is more readable than MPS. The common method to describe the MILP models is the LP file format using Python or C++ language. The OPB format is used to store pseudo-Boolean satisfaction and pesudo-boolean optimization models which contains only boolean variables 0 or 1. So the OPB file format is more suitable to build MILP models for bit-oriented block ciphers. Compared to the LP file format, the OPB format is more concise, easy to read and write. The key words and contents of the two file formats are different, as shown in Table 1. First, the OPB format does not need to specify the variables and their types in particular because all of them have been default to Boolean variables. In addition, we can easy to describe the constraints of the differential propagation of the XOR operation in this format. As the bit-level input differences are a, $b$and the bit-level output difference is c. Then the constraint is:

$$c - a - b + 2ab = 0 \tag{5}$$

Table 1. The comparison between the LP and OPB file format

*.lp	*.opb
Minimize Subject to (constraints) Binary (variables) End	min:(objection) (constraints)

Many optimization parameters are provided by Gurobi to modify your high-level solution strategy, in which the parameter MIPFocus is one of the most important. MIPFocus = 1 means that you are more interested in good quality feasible solution. If the solver is having no trouble finding the optimal solution, select MIPFocus = 2. If the best objective bound is moving very slowly (or not at all), try MIPFocus = 3. So setting parameters properly can effectively reduce the solution time of the model. When solving the model of the number of (related-key) differential of active s-boxes, we can set MIPFocus = 1 to find a higher quality solution that can effectively shorten the solution time. The experimental results in Sect. 3.1 show that the solution time of the optimized models described with the OPB format are reduced greatly.

In addition, we propose a new simple linear inequalities of differential property of modular addition used in the search of impossible differentials. In [7], Fu *et al.* appended which is used to compute the differential probability to the vector and obtained 13 linear inequalities to describe the differential propagation of modular addition in bit-level. In [8,9], this linear inequalities are used to the impossible differential search for ARX ciphers directly. In fact, it is not necessary to compute the differential probability of modular addition in the impossible differential search. We only need to give the linear inequalities of the 56 possible difference patterns of modular addition. By using the inequality-generator() function in the SAGE and the greedy algorithm, we only need 8 linear inequalities of all possible patterns of modular addition in bit-level which are listed below.

$$\begin{cases} -\alpha[i] - \beta[i] - \gamma[i] + \alpha[i+1] + \beta[i+1] + \gamma[i+1] \geq -2 \\ \alpha[i] + \beta[i] + \gamma[i] - \alpha[i+1] - \beta[i+1] - \gamma[i+1] \geq -2 \\ \alpha[i] + \beta[i] + \gamma[i] + \alpha[i+1] + \beta[i+1] - \gamma[i+1] \geq 0 \\ \alpha[i] + \beta[i] + \gamma[i] + \alpha[i+1] - \beta[i+1] + \gamma[i+1] \geq 0 \\ \alpha[i] + \beta[i] + \gamma[i] - \alpha[i+1] + \beta[i+1] + \gamma[i+1] \geq 0 \\ -\alpha[i] - \beta[i] - \gamma[i] + \alpha[i+1] - \beta[i+1] - \gamma[i+1] \geq -4 \\ -\alpha[i] - \beta[i] - \gamma[i] - \alpha[i+1] + \beta[i+1] - \gamma[i+1] \geq -4 \\ -\alpha[i] - \beta[i] - \gamma[i] - \alpha[i+1] - \beta[i+1] + \gamma[i+1] \geq -4 \end{cases} \quad (6)$$

As application, we use this linear inequalities of differential property of modular addition to construct the models of the impossible differential search for the SPECK family with the OPB format in Sect. 3.2.

3 Applications

3.1 Application to LBlock

Lblock is a lightweight block cipher designed by Wu and Zhang at ACNS [10]. Since the shift operation number of the key schedule of LBlock is 29, we can only use the bit-oriented MILP model in related-key differential search. The models based on the LP format and the OPB format are solved respectively. The size of models described by the OPB format are less than the LP format, so the solution time is more faster usually. By setting the parameter MIPFocus = 1, the solution time of the models is further shortened. The results are shown in Table 2. We improved the results of 9∼11 rounds LBlock which show that 9/10/11 rounds exact lower bounds is 7/9/11 active s-boxes. In [11], Sun *et al.* needed about 4 days to find the 11-round exact lower bounds of the number of differential active s-boxes of LBlock in the related-key model. We only needed about 2 days. To the best of our knowledge, the 12-round exact lower bounds of the number of differential active s-boxes of LBlock is first obtained using about 3 weeks.

The computations are performed on PC (Intel(R) Core(TM) i3-4160 CPU, 3.60 GHz, 4.00 GB RAM, 4 cores, window7) with the optimizer Gurobi7.0.1.

Table 2. The exact lower bounds of the number of differential active s-boxes for round-reduced variants of LBlock in the related-key model

Rounds	The number of active s-boxes			Time (in seconds)	
	This paper	[11]	[2]	LP	OPB&MIPFocus = 1
5	1	1	1	2.5	1.56
6	2	2	2	8.5	8.36
7	4	4	3	70	90
8	6	6	5	2419	745
9	7	8	6	6478	1739
10	9	10	8	56462	13238
11	11	12	10	-	161165
12	13	–	–	-	≈3 weeks

3.2 Application to SPECK

SPECK is a family of lightweight block ciphers publicly released by National Security Agency (NSA) in June 2013 [12]. Cui *et al.* [8] proposed an algorithm for finding impossible differentials for block ciphers and obtained four 17-round impossible differentials for HIGHT. In [9], Lee *et al.* found 157 6-round impossible differentials for SPECK-64 by using the same method. All of them used the same linear inequalities of differential property of modular addition provided by [7]. We applied the improved model to the full versions of the SPECK family and limited the input and output differences to only 1 active bit. The results of the experiments are shown in the Table 3. The input or output difference is expressed by the position of non-zero bit. The position of the leftmost bit is 0.

The computations are performed on PC (Intel(R) Core(TM) i7-7500U CPU, 2.70 GHz, 8.00 GB RAM, 4 cores, window10) with the optimizer Gurobi7.0.1.

Table 3. Summary of impossible differentials on the SPECK family

Version	Rounds	$\Delta_{in} \nrightarrow \Delta_{out}$	# ID	Time (in seconds)
SPECK-32	6	$9 \nrightarrow 16$ $9 \nrightarrow 30$ $9 \nrightarrow 31$	3	166
SPECK-48	6	$16 \nrightarrow 0$ $16 \nrightarrow 2$ ⋮ $29 \nrightarrow 45$	20	450
SPECK-64	6	$0 \nrightarrow 16$ $0 \nrightarrow 64$ ⋮ $45 \nrightarrow 62$	157	918
SPECK-96	7	$40 \nrightarrow 93$ $40 \nrightarrow 94$ ⋮ $53 \nrightarrow 94$	12	3946
SPECK-128	7	$0 \nrightarrow 125$ $2 \nrightarrow 125$ ⋮ $90 \nrightarrow 125$	160	16422

In addition, the modular addition exists a differential with probability 1 that the leftmost bit of one of input differences is active and the leftmost bit of the output difference is active. This difference can be propagate to the round function of SPECK. $56 \nrightarrow 56$ and $57 \nrightarrow 56$ are the 7-round impossible differentials for SPECK-128. We can append one round at the bottom and obtain two 8-round impossible differentials $56 \nrightarrow 0, 64$ and $57 \nrightarrow 0, 64$ for SPECK-128.

4 Conclusion

Our work provides a new OPB file format to describe the MILP models for bit-oriented block ciphers and also through setting the parameter MIPFocus = 1 to accelerate the search. In addition, we give a system of simple linear inequalities of differential patterns propagation of modular addition used in impossible differential search. We applied our techniques to LBlock and SPECK.

Acknowledgments. The authors would like to thank all anonymous referees for their valuable comments that greatly improve the manuscript. This work is supported by National Natural Science Foundation of China (No. 61672509, No. 61232009) and National Cryptography Development Fund (MMJJ20170101).

References

1. Mouha, N., Wang, Q., Gu, D., Preneel, B.: Differential and linear cryptanalysis using mixed-integer linear programming. In: Wu, C.-K., Yung, M., Lin, D. (eds.) Inscrypt 2011. LNCS, vol. 7537, pp. 57–76. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34704-7_5
2. Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L.: Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014, Part I. LNCS, vol. 8873, pp. 158–178. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45611-8_9
3. Stein, W., et al.: Sage: Open source mathematical software (2008)
4. Xiang, Z., Zhang, W., Bao, Z., Lin, D.: Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part I. LNCS, vol. 10031, pp. 648–678. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53887-6_24
5. Sasaki, Y., Todo, Y.: New Impossible differential search tool from design and cryptanalysis aspects. In: Coron, J.-S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part III. LNCS, vol. 10212, pp. 185–215. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56617-7_7
6. Gurobi Optimization: Gurobi optimizer reference manual (2013). http://www.gurobi.com
7. Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L.: MILP-based automatic search algorithms for differential and linear trails for speck. In: Peyrin, T. (ed.) FSE 2016. LNCS, vol. 9783, pp. 268–288. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-52993-5_14
8. Cui, T., Jia, K., Fu, K., et al.: New Automatic Search Tool for Impossible Differentials and Zero-Correlation Linear Approximations. Cryptology ePrint archive, Report 2016/689 (2016). https://eprint.iacr.org/2016/689
9. Lee, H.C., Kang, H.C., Hong, D., et al.: New Impossible Differential Characteristic of SPECK64 using MILP. Cryptology ePrint archive, Report 2016/1137 (2016). https://eprint.iacr.org/2016/1137
10. Wu, W., Zhang, L.: LBlock: a lightweight block cipher. In: Lopez, J., Tsudik, G. (eds.) ACNS 2011. LNCS, vol. 6715, pp. 327–344. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21554-4_19
11. Sun, S., Hu, L., Wang, M., et al.: Towards finding the best characteristics of some bit-oriented block ciphers and automatic enumeration of (related-key) differential and linear characteristics with predefined properties. Cryptology ePrint Archive, Report 2014/747 (2014). https://eprint.iacr.org/2014/747
12. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK famillies of lightweight block ciphers. Cryptology ePrint archive, Report 2013/543 (2013). http://eprint.iacr.org/2013/543