Conventional bit-based division property trail search is NP-complete*

Aleksei Udovenko

SnT, University of Luxembourg
aleksei@affine.group

Abstract. This short note shows that the conventional (2-subset) bit-based division property trail search problem is NP-complete.

Keywords: Cryptanalysis · Division property · NP-hardness

1 Introduction

Division property (DP) is a technique introduced by Todo in 2015 [Tod15]. Its goal is finding *integral distinguishers* in symmetric-key primitives such as block ciphers or stream ciphers. This technique was used to cryptanalyze full MISTY1 [BK16, Tod17].

The technique is related to the algebraic representation of the cipher, meaning multivariate polynomials over the binary field $\mathbb{F}_2$, where AND is the multiplication and XOR is the addition. The high-level idea of the technique is to track ways in which a chosen subset of input variables gets multiplied throughout the computation of the primitive. Original division property, also referred to as 2-subset DP or conventional DP, simply decides whether the selected inputs can in principle be multiplied or not; it does not take into account possible XOR-cancellations of the product. Later, [TM16] proposed the 3-subset DP which tracks some of the possible cancellations. Finally, [HLM+20] showed how to track all possible cancellations by counting the parity, and called the technique "3-subset DP without the unknown subset", which was also called *monomial prediction* in [HSWW20].

We focus on the bit-based variant introduced in [TM16], as opposed to word-level/state-level division property [Tod15], which are easier to evaluate due to smaller state size (depending on the word size). Xiang, Zhang, Bao, and Lin showed in [XZBL16] how to use mixed-integer linear programming (MILP) solvers to efficiently apply the conventional bit-based division property application to multiple block ciphers. One of the key ideas here was representing the technique in terms of a *trail search problem* (see Section 2). The surprising efficiency and generality of this method led to many further results, including adaptations to the stronger division property variants: 3-subset DP [WHG+19], monomial prediction [HSWW20]. These adaptations come at a large performance and complexity penalty, since the problem reduces to parity counting of a given Boolean circuit [DRS24] ($\oplus P$-complete problem), which is believed to be hard, although not proven to be NP-hard. Specific optimizations and tricks are needed for applying the perfect DP variant. And still, many large instances are not feasible to be tested.

*The work was supported by the Luxembourg National Research Fund's (FNR) project PQseal (C24/IS/18978392).

Our contribution This note revisits the *conventional* (2-subset) bit-based division property and its computational hardness. Despite its successful performance in cryptanalysis of many primitives, we show that the underlying division trail search/existence problem (defined as in [XZBL16]) is NP-complete, via a simple classical-style reduction from CNF-SAT. This shows that even the basic division property variant cannot be evaluated for arbitrary circuits in polynomial time (unless P=NP), and its success in practice is likely attributed to a specific structure of targets, such as narrow circuit width (block size) and/or slow diffusion.

2 Two-subset division property

Definition 1 (AND-XOR circuit)**.** A circuit with AND and XOR gates is a directed acyclic graph $C = (V, E)$ with nodes $V = I \cup X \cup A \cup O$ corresponding respectively to input gates I (indegree zero), XOR gates X, AND gates A, output gates O (indegree one, outdegree zero).

Definition 2 (Division trail [XZBL16])**.** A *division trail* over an AND-XOR circuit $C = (I \cup X \cup A \cup O,\ E)$ is an assignment $D \in \{0,1\}^{I \cup O \cup E}$ of 0/1 labels to the input and output gates, and to the wires E, with the following rules:

1. for an input gate, the integer sum of the outgoing labels must be equal to the gate's label;
2. for a XOR gate, the integer sum of the incoming labels must equal the integer sum of the outgoing labels, and these sums must equal 0 or 1;
3. for an AND gate, the integer sum of the outgoing wire labels must be equal to 0 if all incoming wire labels are 0, and 1 otherwise;
4. for an output gate, its label must be equal to the incoming edge's label.

Remark 1. The trail rules from [XZBL16] do not allow outdegree larger than 1 for XOR and AND gates, but allow special COPY gates with indegree 1 and arbitrary positive outdegree (or a 1-to-2 COPY gate which can be repeated to achieve the same effect). Our description eliminates them by allowing arbitrary outdegrees and adapting the rules accordingly. It is easy to verify that the definitions are equivalent.

Remark 2. We also note that the trail rules from [XZBL16] are optimized (compared to the original rules from [Tod15, TM16]) by removing the so-called "redundant" transitions, which are derived from non-redundant ones by switching any amount of output labels from 0 to 1.

Division trail rules can be interpreted in the (maximum) network flow framework with 0/1 flows: a XOR gate simply limits the flow to 1 (and can be implemented in the standard network flow by an edge with capacity 1); an AND gate is special: it allows to discard[1] (or send directly to the sink) any flow above 1. As we will show, this special gate makes the problem NP-hard, in contrast to the standard maximum flow problem which can be easily solved in polynomial time.

Problem 1 (Division trail search [XZBL16])**.** *Given a Boolean circuit C with XOR and AND gates and one output gate, decide if there exists a division trail assigning 1 to all input gates and to the output gate.*

[1] In the context of our problem, discarding flow is always beneficial, that is why the AND rule we describe only outputs at most one 1.

Note that an AND gate can propagate any assignment of its incoming labels, while a XOR gate requires that at most one incoming label is set to 1. The nondeterminism in the propagation comes solely from "routing": deciding which of the outgoing edges (if any) needs to be set to 1.

Remark 3. This is the most minimalistic, monotone variant of the problem designed to minimize the search space, following the optimizations from [XZBL16], focusing on all input labels set to 1 and on a single output bit set to 1. One may allow additional 0-labeled inputs and outputs, but these can be eliminated together with relevant parts of the circuit, without changing the solvability of the instance.

3 NP-completeness of the two-subset division property

Theorem 1. *Problem 1 is NP-complete.*

Proof. The problem is clearly in NP since an assignment can be efficiently verified to satisfy the rules. We will reduce the CNF-SAT problem (Satisfiability of a CNF[2] formula) to it to complete the claim. Without loss of generality, we assume that the formula contains at least one clause.

Consider an arbitrary CNF-SAT formula consisting of binary variables $X = \{x_i\}_{1 \le i \le n}$ and clauses $C = \{c_j\}_{1 \le j \le m}$, $c_j \subseteq X \cup \neg X$, where $\neg X = \{\neg x_i\}_{1 \le i \le n}$ are symbolic negations of variables. This represents the CNF formula $c_1 \wedge c_2 \wedge \ldots \wedge c_m$ with

$$c_j = \{x_{i_1}, x_{i_2}, \ldots, \neg x_{k_1}, \neg x_{k_2}, \ldots\}$$

representing the clause

$$(x_{i_1} \vee x_{i_2} \vee \ldots \vee \neg x_{k_1} \vee \neg x_{k_2} \vee \ldots).$$

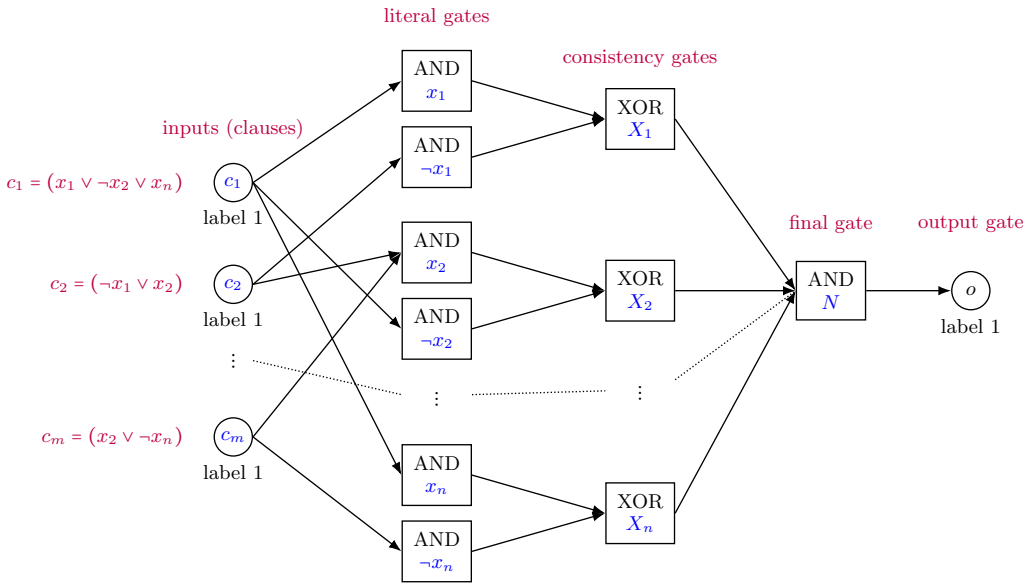

Figure 1: Construction of the circuit corresponding to a CNF-SAT problem $X = \{x_i\}_{1 \le i \le n}$, $C = \{c_j\}_{1 \le j \le m}$, $c_j \subseteq X \cup \neg X$.

[2]Conjunctive normal form.

We will now construct a division trail search instance/circuit (see Figure 1). First, assign an input gate to each of the clauses c_j (m inputs). Second, create an AND node for each variable x_i and for its negation $\neg x_i$ ($2n$ AND gates). We will identify these clauses/variable labels with the corresponding gates. Then, connect each clause $c_j = \{x_{i_1}, \ldots, \neg x_{k_1} \ldots\}$ with each of the variables or negations present in it:

$$c_j \to x_{i_1}, \quad \ldots, \quad c_j \to \neg x_{k_1}, \quad \ldots$$

Third, create a XOR gate X_i for each variable x_i (n XOR gates) and add wires from x_i and $\neg x_i$ to it:

$$x_i \to X_i, \quad \neg x_i \to X_i.$$

Lastly, we add a final AND gate N, and wires $X_i \to N$ for all i, and wire $N \to o$ to the only output gate o. Its purpose is only a formality to ensure a single output. The resulting circuit is thus

$$\begin{aligned} C &= (I \cup X \cup A \cup O,\, E), \quad \text{where} \\ I &= \{c_1, \ldots, c_m\}, \\ X &= \{X_1, \ldots, X_n\}, \\ A &= \{x_1, \ldots, x_n, \neg x_1, \ldots, \neg x_n, N\}, \\ O &= \{o\}, \end{aligned}$$

and the edges E are as described above. The circuit has size linear in n, m, and in the number of literals $\sum_{i=1}^{m} |c_i|$ in the CNF formula: $m + 3n + 2$ nodes and $|E| = 3n + 1 + \sum_{i=1}^{m} |c_i|$ edges.

We now prove that Problem 1 for this circuit has a solution if and only if the original CNF formula is satisfiable.

(CNF satisfiable ⇒ division trail exists) Consider a satisfying assignment for the formula. For each clause c_j, we choose arbitrarily the literal x_i (or $\neg x_i$) which satisfies it in the assignment, and assign label 1 to the corresponding edge; all other edges from c_j are assigned to 0. Since under a fixed satisfying CNF assignment only one of x_i and $\neg x_i$ is true, at most one of the corresponding AND gates in the circuit will receive a positive number of nonzero incoming labels. Therefore, at most one of these two gates may have its single output wire assigned the label 1. Therefore, the XOR gate X_i will have at most one incoming wire labeled with 1, and its output wire is correctly and deterministically assigned. Finally, it is easy to see that the output wire of the AND gate has to be (correctly) set to 1, since no gate can completely eliminate all labels 1 from its input, and the circuit's inputs are labeled with 1.

(division trail exists ⇒ CNF satisfiable) In a division trail, each XOR gate X_i cannot have both incoming wires set to 1. Therefore, at most one of the AND gates x_i, $\neg x_i$ have a positive number of incoming wires and the outgoing wire labeled 1. If x_i does, we set the variable x_i to 1. Otherwise, we set the variable x_i to 0. For every input gate c_j (which is labeled 1 by the problem definition), there must be a single outgoing edge labeled 1 to some gate $x_i/\neg x_i$. This means that this gate was previously selected and the variable was assigned. By the construction of the circuit edges $c_j \to x_i/\neg x_i$, this variable assignment satisfies this clause. We conclude that all the clauses must be satisfied with the assignment, and the assignment is non-contradictory due to the discussion from the beginning of the paragraph. □

Discussion If one takes a hard CNF instance and construct the circuit as in the proof, this should result in a hard division trail search instance (for MILP or any other solver). The circuit in the proof has m inputs and $3n + 1$ computational gates. For hard CNF instances the circuit will have many inputs. The depth of the circuit is constant (three,

counting XOR/AND gates only), although the first and the third layers use arbitrary fan-in gates.

While division property is difficult to evaluate over large binary linear layers [MWWZ22], there are no complicated linear layers in the circuit from the reduction, the combinatorial complexity clearly comes from routing the input labels 1 to the right literal $x_i/\neg x_i$ gates. In the description from [XZW+24], the "routing" is implemented using the COPY gates, and thus they are the main source of hardness in the proof.

It is interesting that the weakest bit-based division property variant (2-subset) is NP-complete, while the strongest one (monomial prediction) is not known to be such (although it is known to be $\oplus P$-complete[3]). At the same time, there is no proof that the $\oplus P$ class contains NP (although probabilistic machines with an $\oplus P$ oracle can solve NP problems [For09]), so the situation with division property variants is similar to the situation with the complexity classes.

4 Conclusion

The main consequence of this result is ruling out a generally efficient (for example, maximum-flow-based) algorithm for the conventional 2-subset bit-based division property. An interesting remaining question is understanding good performance of MILP solvers on real-world instances, as well as their limits.

References

[BK16] Achiya Bar-On and Nathan Keller. A 2^{70} attack on the full MISTY1. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 435–456. Springer, Berlin, Heidelberg, August 2016. doi:10.1007/978-3-662-53018-4_16. 1

[DRS24] Pranjal Dutta, Mahesh Sreekumar Rajasree, and Santanu Sarkar. Complexity of monomial prediction in cryptography and machine learning. In *2024 26th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 118–125, 2024. doi:10.1109/SYNASC65383.2024.00032. 1

[For09] Lance Fortnow. A simple proof of Toda's theorem. *Theory of Computing*, 5(7):135–140, 2009. URL: https://theoryofcomputing.org/articles/v005a007, doi:10.4086/toc.2009.v005a007. 5

[HLM+20] Yonglin Hao, Gregor Leander, Willi Meier, Yosuke Todo, and Qingju Wang. Modeling for three-subset division property without unknown subset - improved cube attacks against Trivium and Grain-128AEAD. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part I*, volume 12105 of *LNCS*, pages 466–495. Springer, Cham, May 2020. doi:10.1007/978-3-030-45721-1_17. 1

[HSWW20] Kai Hu, Siwei Sun, Meiqin Wang, and Qingju Wang. An algebraic formulation of the division property: Revisiting degree evaluations, cube attacks, and key-independent sums. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part I*, volume 12491 of *LNCS*, pages 446–476. Springer, Cham, December 2020. doi:10.1007/978-3-030-64837-4_15. 1

[3] $\oplus P$ is the class of decision problems with odd number of accepting witnesses verifiable in polynomial time. See [PZ82].

[MWWZ22] Yongxia Mao, Wenling Wu, Bolin Wang, and Li Zhang. Improved division property for ciphers with complex linear layers. In Khoa Nguyen, Guomin Yang, Fuchun Guo, and Willy Susilo, editors, *ACISP 22*, volume 13494 of *LNCS*, pages 106–124. Springer, Cham, November 2022. doi:10.1007/978-3-031-22301-3_6. 5

[PZ82] Christos H. Papadimitriou and Stathis K. Zachos. Two remarks on the power of counting. In Armin B. Cremers and Hans-Peter Kriegel, editors, *Theoretical Computer Science*, pages 269–275, Berlin, Heidelberg, 1982. Springer Berlin Heidelberg. 5

[TM16] Yosuke Todo and Masakatu Morii. Bit-based division property and application to simon family. In Thomas Peyrin, editor, *FSE 2016*, volume 9783 of *LNCS*, pages 357–377. Springer, Berlin, Heidelberg, March 2016. doi:10.1007/978-3-662-52993-5_18. 1, 2

[Tod15] Yosuke Todo. Structural evaluation by generalized integral property. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 287–314. Springer, Berlin, Heidelberg, April 2015. doi:10.1007/978-3-662-46800-5_12. 1, 2

[Tod17] Yosuke Todo. Integral cryptanalysis on full MISTY1. *Journal of Cryptology*, 30(3):920–959, July 2017. doi:10.1007/s00145-016-9240-x. 1

[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, *ASIACRYPT 2019, Part III*, volume 11923 of *LNCS*, pages 398–427. Springer, Cham, December 2019. doi:10.1007/978-3-030-34618-8_14. 1

[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, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 648–678. Springer, Berlin, Heidelberg, December 2016. doi:10.1007/978-3-662-53887-6_24. 1, 2, 3

[XZW+24] Binwu Xiang, Jiang Zhang, Kaixing Wang, Yi Deng, and Dengguo Feng. NTRU-based bootstrapping for MK-FHEs without using overstretched parameters. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part I*, volume 15484 of *LNCS*, pages 241–270. Springer, Singapore, December 2024. doi:10.1007/978-981-96-0875-1_8. 5