SCIENCE CHINA
Information Sciences

• LETTER •

May 2021, Vol. 64 159103:1–159103:3
https://doi.org/10.1007/s11432-018-9772-0

STP models of optimal differential and linear trail for S-box based ciphers

Yu LIU[1,2], Huicong LIANG[1], Muzhou LI[1], Luning HUANG[1], Kai HU[1], Chenhe YANG[1] & Meiqin WANG[1*]

[1]*Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan* 250100*, China;*
[2]*School of Computer Engineering, Weifang University, Weifang* 261061*, China*

Received 25 September 2018/Revised 12 December 2018/Accepted 18 January 2019/Published online 23 March 2021

Citation Liu Y, Liang H C, Li M Z, et al. STP models of optimal differential and linear trail for S-box based ciphers. Sci China Inf Sci, 2021, 64(5): 159103, https://doi.org/10.1007/s11432-018-9772-0

Dear editor,

SAT solvers, based on heuristic algorithms, are used to solve Boolean satisfiability (SAT) problems. Satisfiability modulo theories (SMT) problem is a decision problem concerned with the satisfiability of a logical formula; it is expressed as a combination of first-order theories. Here, we use the simple theorem prover (STP) SMT solver [1], which is designed to solve constraints involving bit-vectors and arrays; hence, it is suitable for S-box-based ciphers. In this study, we construct STP-based automatic search models for optimal differential and linear trails, which can be used even if a cipher involves a DDT* or LAT*. Here, we call a difference distribution table (DDT) or linear approximation table (LAT) whose entities are not equal to powers of two a DDT* or LAT*, respectively. Further, we apply these models to several ciphers, Table 1 compares the resulting differential/linear trails with previous results.

Models for differential and linear trails with optimal probabilities/correlations. For ciphers with a DDT* or LAT*, Abdelkhalek et al. [2] have already shown how to model the DDT* probabilistically, rounding the negative base-2 logarithm of the probability to one decimal place. However, this method may miss some good differential trails. In contrast, our approach can round the irrational logarithm values involved to sufficient precision to preserve monotonicity. Suppose there are a total of N S-boxes (both active and inactive) in the differential trail, N_j of which have probabilities of $j/2^m$ ($2 \leqslant j \leqslant 2^m$, j is even). Under the Markov cipher assumption, the probability p of this differential trail is

$$\begin{aligned} p =&(2^m)^{N_{2^m}} \times \cdots \times 6^{N_6} \times 4^{N_4} \times 2^{N_2}/(2^m)^N \\ =&f_p(N_{2^m}, N_{2^m-2}, \ldots, N_2). \end{aligned}$$

Next, we build an approximation function G^* to compute p as follows:

$$\begin{aligned} &G^*(N_{2^m}, N_{2^m-2}, \ldots, N_2) \\ &= -\sum_{\substack{2\leqslant j\leqslant 2^m \\ j \text{ is even}}} \sum_{\substack{k=1 \\ N_j\neq 0}}^{N_j} \lceil (\log_2 j - m) \times 10^{n_f} \rceil, \end{aligned}$$

where n_f is a positive integer called the probability precision. We must choose the value of n_f such that the following property holds.

Algorithm 1 first builds a list T_N to store the f_p values, together with their corresponding $(2^{m-1}-1)$-tuples $\{N_{2^m-2}, \ldots, N_4, N_2\}$. Then, it sorts the list T_N according to the keywords f_p, N_{2^m-2}, …, N_4, N_2 in ascending order. Next, it initializes n_f to 1 and computes values for the approximation function G^* based on taking the corresponding tuples in T_N as inputs. If G^* satisfies Property 1, it returns the value of n_f; otherwise, it increments the value of n_f and recomputes G^*. Let the number of non-zero entities (expect 2^m) in the DDT be M and the number of loops needed to compute n_f be b (usually $b < 20$). The time required for the execution of Steps 2–6 is then $\binom{M+N_s-1}{N_s}$ times that for computing f_p, and the time taken for the execution of Steps 8–17 is $2b \times \binom{M+N_s-1}{N_s}$ times that needed to compute G^*.

Property 1. For any two $(2^{m-1}-1)$ tuples $\{N_{2^m}, \ldots, N_2\}$ and $\{N'_{2^m}, \ldots, N'_2\}$, if $f_p(N_{2^m}, \ldots, N_2) > f_p(N'_{2^m}, \ldots, N'_2)$, then $G^*(N_{2^m}, \ldots, N_2) < G^*(N'_{2^m}, \ldots, N'_2)$.

Property 2. Assume that N S-boxes are involved in the differential trail. For the i-th S-box S_i, denote the input and output differences by $\Delta_i^{\text{in}} \in \mathbb{F}_2^m$ and $\Delta_i^{\text{out}} \in \mathbb{F}_2^l$, respectively. In addition, let v_i be a flag variable representing the validity of difference propagation, and define p_i as the probability over S_i. Then, we can represent difference prop-

* Corresponding author (email: mqwang@sdu.edu.cn)

Algorithm 1 Algorithm to calculate n_f, given N_S and the DDTs

Input: number of active S-boxes N_S, DDTs.
Output: $n_f \in \mathbb{Z}^+$.
Data:
V_{count}: number of rows in the list T_N;
$f_p[i]$: the i-th f_p value in the list T_N;
$G^*[i]$: value of the approximation function corresponding to the i-th $(2^{m-1}-1)$-tuple in the list T_N.

1: $V_{\text{count}} \leftarrow 0$;
2: **for** all possible $(2^{m-1}-1)$-tuples $\{N_{2^m-2},\ldots,N_4,N_2\}$ **do**
3: $\quad f_p \leftarrow \frac{1}{2^{N_S \cdot pm}} \cdot \prod_{k=1}^{2^{m-1}-1}(2k)^{N_{2k}}$;
4: $\quad$ Add $\{f_p, N_{2^m-2},\ldots,N_4,N_2\}$ to T_N;
5: $\quad V_{\text{count}}++$;
6: **end for**
7: Sort T_N according to the keywords f_p, $N_{2^m-2},\ldots,N_4$, N_2 in ascending order;
8: $n_f \leftarrow 1$;
9: $G^*[1] \leftarrow -\sum_{k=1}^{2^{m-1}-1}(N_{2k} \times \lceil(\log_2 2k - m) \times 10^{n_f}\rceil)$ corresponding to the first $(2^{m-1}-1)$-tuple in T_N;
10: **for** $i \leftarrow 2$ to V_{count} **do**
11: $\quad G^*[i] \leftarrow -\sum_{k=1}^{2^{m-1}-1}(N_{2k} \times \lceil(\log_2 2k - m) \times 10^{n_f}\rceil)$ corresponding to the i-th $(2^{m-1}-1)$-tuple in T_N;
12: $\quad$ **if** $(G^*[i]>G^*[i-1]) \| (G^*[i]=G^*[i-1]$ && $f_p[i]=f_p[i-1])$ **then**
13: $\quad\quad$ **continue**
14: $\quad$ **else**
15: $\quad\quad n_f++$;
16: $\quad\quad$ **goto** Line 9;
17: $\quad$ **end if**
18: **end for**
19: **return** n_f.

Algorithm 2 Model for generating differential trails with the expected number of active S-boxes

Input: number of rounds r, expected threshold, and flag. (Flags of 0 and 1 indicate related-key and single-key settings, respectively.)
Output: a differential trail with the expected number of active S-boxes.

1: **for** round $\leftarrow 1$ to r **do**
2: $\quad$ List equations for S-Boxes satisfying Property 2.
3: $\quad$ Write equations for the linear layer.
4: **end for**
5: **if** flag $= 1$ **then**
6: $\quad$ Generate equations that set the input differences to non-zero values.
7: **else**
8: $\quad$ Generate equations that set the master key differences to non-zero values.
9: $\quad$ List equations corresponding to the key schedule.
10: **end if**
11: Write equations that set the number of S-boxes to below the expected threshold.
12: Solve the above equations using the STP solver.

agation through S_i via the following equations:

$$v_i = \begin{cases} 0 \text{ (invalid)}, & \text{if } (\Delta_i^{\text{in}}, \Delta_i^{\text{out}}) \in \mathbb{S}_i^0, \\ 1 \text{ (valid)}, & \text{if } (\Delta_i^{\text{in}}, \Delta_i^{\text{out}}) \in \mathbb{S}_i^j,\ 0 < j \leqslant 2^m, \end{cases}$$

$v_i = 1$, which only allows valid input and output differences to remain.

$$p_i = \begin{cases} 0, & \text{if } \Delta_i^{\text{in}} = 0, \\ c_j^*, & \text{if } (\Delta_i^{\text{in}}, \Delta_i^{\text{out}}) \in \mathbb{S}_i^j,\ 0 < j < 2^m, \end{cases}$$

where $c_j^* = -\lceil(\log_2 j - m) \times 10^{n_f}\rceil$ and the set $\mathbb{S}_i^j$ $(1 \leqslant i \leqslant N)$ contains all pairs of input and output differences with probabilities of $j/2^m$.

Algorithm 3 searches for a differential trail with the expected probability, while Algorithm 4 provides a general procedure for finding differential trails with optimal probability. In some cases, we can only obtain improved trails rather than optimal ones because of the impractical runtime (complexity) of Algorithm 4. Due to the duality between differential and linear propagation, the model for finding linear trails is similar.

Algorithm 3 Algorithm for finding differential trails with the expected probability

Input: number of rounds r, expected threshold G^*_{th}, N_S, flag. (Flags of 0 and 1 indicate related-key and single-key settings, respectively.)
Output: a differential trail with a probability of less than G^*_{th}.

1: **for** round $\leftarrow 1$ to r **do**
2: $\quad$ List equations for S-Boxes satisfying Property 2.
3: $\quad$ Write equations for the linear layer.
4: **end for**
5: **if** flag=1 **then**
6: $\quad$ Generate equations that set the input differences to non-zero values.
7: **else**
8: $\quad$ Generate equations that set the master key differences to non-zero values.
9: $\quad$ List equations corresponding to the key schedule.
10: **end if**
11: Write equations that set $G^* < G^*_{\text{th}}$.
12: Write equations that set the number of active S-boxes to N_S.
13: Solve all the above equations with the STP solver.

Algorithm 4 General procedure for finding differential trails with optimal probabilities

Input: number of rounds r, $P_{\max}$.
Output: a differential trail with optimal probability.
Data: $P_{\max}$ represents the maximum probability over all DDTs.

1: Execute Algorithm 2 to obtain the differential trail with the minimum number of active S-boxes N_s.
2: Compute the probability p of this trail.
3: Store this trail in the list L.
4: $t \leftarrow 0$.
5: Set the number of active S-boxes to $N_s + t$, then compute the value of n_f with Algorithm 1.
6: Gradually reduce the value of G^*_{th} and execute Algorithm 3 to find the optimal trail with $N_s + t$ active S-boxes.
7: Compute the probability p' of this trail.
8: **if** $p' > p$ **then**
9: $\quad p \leftarrow p'$;
10: $\quad$ Use this trail update L.
11: **end if**
12: $t++$.
13: **if** $(P_{\max})^{N_S+t} > p$ **then**
14: $\quad$ **goto** Line 5;
15: **else**
16: $\quad$ **return** L and p.
17: **end if**

Applications. We apply our models to several S-box-based ciphers, namely GIFT-128 [3], ICEBERG [4], DES and DESL [5], ARIA [6] and SM4 [7]. These experiments were carried out using a cluster of computers with two Intel Xeon E5-2690 CPUs (2.60 GHz, 128 G memory, 24 cores). As shown in Table 1, we were able to obtain improved/optimal differential and linear trails, in terms of the number of rounds or the probability/correlation. In some

Table 1 Comparison with other S-boxes-based ciphers[a)]

Cipher	Trail	Rounds	Probability/Correlation	Ref.
GIFT-128	Differential	9	$2^{-46.0}$	[3]
		9	$\mathbf{2^{-45.4}}$	This study
		10	$\mathbf{2^{-49.4}}$	This study
		11	$\mathbf{2^{-54.4}}$	This study
		12	$\mathbf{2^{-60.4}}$	This study
		13	$\mathbf{2^{-67.8}}$	This study
		21	$2^{-126.4}$	This study
ICE-BERG	Linear	6	$2^{-30.1}$	[4]
		6	$\mathbf{2^{-30.0}}$	This study
DES	RKD[b)]	6	$2^{-12.9}$	[5]
		6	$\mathbf{2^{-12.2}}$	This study
		7	$2^{-20.4}$	[5]
		7	$2^{-18.3}$	This study
DESL	RKD	7	$2^{-20.0}$	[5]
		7	$\mathbf{2^{-12.2}}$	This study
		11	$< 2^{-31}$	[5]
		11	$2^{-51.7}$	This study
ARIA	Linear	5	2^{-60}	[6]
		5	$\mathbf{2^{-52.6}}$	This study
		6	$\mathbf{2^{-72}}$	This study
SM4	Differential	19	2^{-124}	[7]
		19	2^{-123}	This study

a) The bold indicates that the corresponding differential/linear trail is optimal.

b) RKD: related-key differential.

cases, we obtained differential or linear trails with optimal probabilities/correlations.

Acknowledgements This work was supported by National Cryptography Development Fund (Grant No. MMJJ20170102), National Natural Science Foundation of China (Grant Nos. 61572293, 61502276, 61692276), Natural Science Foundation of Shandong Province (Grant No. ZR2016FM22), and Major Scientific and Technological Innovation Projects of Shandong Province, China (Grant No. 2017CXGC0704).

References

1 Ganesh V, Hansen T, Soos M, et al. STP. 2014. https://stp.github.io/
2 Abdelkhalek A, Sasaki Y, Todo Y, et al. MILP modeling for (Large) S-boxes to optimize probability of differential characteristics. IACR Trans Symmetric Cryptol, 2017, 2017: 99–129
3 Banik S, Pandey S K, Peyrin T, et al. GIFT: a small PRESENT. In: Proceedings of International Conference on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2017. 321–345
4 Sun Y. Linear cryptanalysis of light-weight block cipher ICEBERG. In: Advances in Electronic Commerce, Web Application and Communication. Berlin: Springer, 2012. 529–532
5 Biryukov A, Nikolić I. Search for related-key differential characteristics in DES-like ciphers. In: Proceedings of International Workshop on Fast Software Encryption. Berlin: Springer, 2011. 18–34
6 Abdelkhalek A, Tolba M, Youssef A M. Improved linear cryptanalysis of round-reduced ARIA. In: Proceedings of International Conference on Information Security. Berlin: Springer, 2016. 18–34
7 Su B Z, Wu W L, Zhang W T. Security of the SMS4 block cipher against differential cryptanalysis. J Comput Sci Technol, 2011, 26: 130–138