IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 2019, No. 4, pp. 290–317. DOI:10.13154/tosc.v2019.i4.290-317

Comprehensive security analysis of CRAFT

Hosein Hadipour[1], Sadegh Sadeghi[2,3], Majid M. Niknam[4], Ling Song[5] and Nasour Bagheri[6,7]

[1] Department of Mathematics and Computer Science, University of Tehran, Tehran, Iran, hsn.hadipour@gmail.com

[2] Department of Mathematics, Faculty of Mathematical Sciences and Computer, Kharazmi University, Tehran, Iran, s.sadeghi.khu@gmail.com

[3] LIMOS, University Clermont Auvergne, Clermont-Ferrand, France

[4] Department of Mathematics, Faculty of Mathematical Sciences and Computer, Kharazmi University, Tehran, Iran, std_mmniknam@khu.ac.ir

[5] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing China songling.qs@gmail.com

[6] Electrical Engineering Department, Shahid Rajaee Teacher Training University, Tehran, Iran, Nabgheri@sru.ac.ir

[7] School of Computer Science (SCS), Institute for Research in Fundamental Sciences (IPM), Tehran, Iran

Abstract. `CRAFT` is a lightweight block cipher, designed to provide efficient protection against differential fault attacks. It is a tweakable cipher that includes 32 rounds to produce a ciphertext from a 64-bit plaintext using a 128-bit key and 64-bit public tweak. In this paper, compared to the designers' analysis, we provide a more detailed analysis of `CRAFT` against differential and zero-correlation cryptanalysis, aiming to provide better distinguishers for the reduced rounds of the cipher. Our distinguishers for reduced-round `CRAFT` cover a higher number of rounds compared to the designers' analysis. In our analysis, we observed that, for any number of rounds, the differential effect of `CRAFT` has an extremely higher probability compared to any differential trail. As an example, while the best trail for 11 rounds of the cipher has a probability of at least 2^{-80}, we present a differential with probability $2^{-49.79}$, containing $2^{29.66}$ optimal trails, all with the same optimum probability of 2^{-80}. Next, we use a partitioning technique, based on optimal expandable truncated trails to provide a better estimation of the differential effect on `CRAFT`. Thanks to this technique, we are able to find differential distinguishers for 9, 10, 11, 12, 13, and 14 rounds of the cipher in single tweak model with the probabilities of at least $2^{-40.20}$, $2^{-45.12}$, $2^{-49.79}$, $2^{-54.49}$, $2^{-59.13}$, and $2^{-63.80}$, respectively. These probabilities should be compared with the best distinguishers provided by the designers in the same model for 9 and 10 rounds of the cipher with the probabilities of at least $2^{-54.67}$ and $2^{-62.61}$, respectively.

In addition, we consider the security of `CRAFT` against the new concept of related tweak zero-correlation (ZC) linear cryptanalysis and present a new distinguisher which covers 14 rounds of the cipher, while the best previous ZC distinguisher covered 13 rounds. Thanks to the related tweak ZC distinguisher for 14 rounds of the cipher, we also present 14 rounds integral distinguishers in related tweak mode of the cipher.

Although the provided analysis does not compromise the cipher, we think it provides a better insight into the designing of `CRAFT`.

Keywords: Lightweight block cipher · differential · zero-correlation · tweakable cipher · MILP · SAT · `CRAFT`.

Received: 2019-09-01, Accepted: 2019-11-01, Published: 2020-01-31

1 Introduction

Lightweight cryptography received extensive attention over the last decade, motivated by the emergent growth of resource-constrained devices such as RFID tags and IoT edge devices. To address this demand, several lightweight primitives have been proposed by researchers, to just name some, SKINNY [BJK+16], PRESENT [BKL+07], MIBS [ISSK09], SIMON [BSS+15], SPECK [BSS+15], Quark [AHMN13] and PHOTON [GPP11]. In this direction, recently, the NIST lightweight cryptography competition also announced its second-round candidates. Among lightweight primitives, (tweakable) block ciphers received more attention and many nice designs have already been proposed, each of which targets different applications.

On the other hand, Side-Channel Analysis (SCA) attacks, such as power/time analysis and fault analysis, target implementation of ciphers and protecting a cipher against them requires extra cost, e.g., extra area. Given the constraints of target applications of lightweight block ciphers, it may not be possible to protect them using conventional approaches, e.g., protecting using hardware redundancy for fault analysis which commonly requires double area compared to the unprotected cipher. Hence, several researches have aimed to provide efficient protection against SCA from design. More precisely, they selected a component to design cipher such that they can provide efficient protection against a specific attack,e.g., LS-Designs [GLSV14], FRIT [SBD+18], ZORRO [GGNS13] and Fides [BBK+13].

In this direction, to provide efficient protection against differential fault analysis, Beierle et al. proposed CRAFT [BLMR19], which is a tweakable lightweight block cipher (A tweakable block cipher maps a n-bit plaintext to a n-bit ciphertext using a k-bit secret key and a t-bit tweak). In addition, they supported their design by extensive analysis against known attacks, e.g., differential cryptanalysis, impossible differential cryptanalysis, linear cryptanalysis, zero-correlation cryptanalysis, and so on. Their analysis shows that the cipher provides desired security against these attacks. However, there is still room for third-party analysis. In addition, related tweak zero-correlation [ADG+19] is a new concept which has been proposed after the publication of CRAFT, hence, the security of the cipher against this attack is worth an investigation. Moreover, due to the nature of differential cryptanalysis, which requires to search over a very large space of all possible trails, it should be always possible to improve the previous analysis by using advanced search approaches. Hence, in this paper, we tackle the detailed security analysis of CRAFT against the above-mentioned analyses. The paper's contribution is summarized as follows (also, Table 1 shows a comparison of our results with previous ones for CRAFT):

1. We present 14 rounds zero-correlation distinguishers for the cipher in the related tweak mode. It should be compared with the 13-round distinguisher proposed by the designers, however, in the single tweak mode.

2. Given the related tweak ZC distinguisher for 14 rounds of the cipher and following the connection between zero-correlation and integral distinguisher [SLR+15], we also present 14 rounds integral distinguishers in the related tweak mode of the cipher.

3. Thanks to the advanced automated search models based on CryptoSMT [Köl19] and MILP [MWGP11, SHW+14b, SHW+14a], we are able to improve the designers' lower bounds for differential cryptanalysis. More precisely, while the designers' lower bound on the probability of differential for 9 and 10 rounds of the cipher are least $2^{-54.67}$ and $2^{-62.61}$, respectively, we are able to present 9, 10, 11, 12, 13, and 14 rounds of the differential in single-tweak model with the probabilities of at least $2^{-40.20}$, $2^{-44.89}$, $2^{-49.79}$, $2^{-54.48}$, $2^{-59.13}$, and $2^{-63.80}$, respectively, that improve the previous lower bounds significantly . We make our implementations of the attacks and our modeling of algorithms in MILP and SAT freely available at: https://github.com/hadipourh/craftanalysis.

Table 1: Summary of the main results of attacks on CRAFT. Where ST , RT and RK denotes single tweak mode, related tweak mode and related key mode respectively and RT_i denotes RT mode that is started with TK_i. In addition, D, TD, LH, ID, INT and ZC denote differential effect, truncated differential, linear hull, impossible differential, integral, and zero-correlation cryptanalysis, respectively. For example, RT_0-D denotes differential effect of CRAFT in related tweak mode, starting with TK_0.

Attack	♯ Rounds	Probability	Reference
ST-D	10	$2^{-62.61}$	[BLMR19]
	10	$2^{-44.89}$	this paper
	11	$2^{-49.79}$	
	12	$2^{-54.48}$	
	13	$2^{-59.13}$	
	14	$2^{-63.80}$	
ST-TD	12	2^{-36}	[MA19]
ST-LH	14	$2^{-62.12}$	[BLMR19]
RT_0-D	15	$2^{-55.14}$	[BLMR19]
RT_1-D	16	$2^{-57.18}$	
RT_2-D	17	$2^{-60.14}$	
RT_3-D	16	$2^{-55.14}$	
ST-ID	13	-	
ST-INT	13	-	
ST-ZC	13	-	
RT-ZC	14	-	this paper
RT-INT	14	-	this paper
RK-D	32	2^{-32}	[EY19]

4. We also show some typos in the designers' analysis which could be useful for later studies. For example, we show that two out of 12 zero-correlation masks for 13 rounds of CRAFT, that are provided by the designers, are not valid. We also provide exact trails, with non zero-correlation, for those masks. A similar result is also presented for their proposed impossible differential trails.

The rest of the paper is organized as follows: in Section 2, we present the required preliminaries and also briefly describe CRAFT. In Section 3, we present the zero-correlation analysis in related tweak mode. Differential effect analysis of the cipher is described in Section 4. Section 5 presents our investigation results on some of the designers security claims and points out some of their typos. Finally, we conclude the paper in Section 6.

2 Preliminaries

In this section, we present the required preliminaries and a brief description of CRAFT.

2.1 Notations

The notation used in the paper is summarized in Table 2.

2.2 A brief description of CRAFT

CRAFT is a 64-bit lightweight block cipher which supports 128-bit key and 64-bit tweak and its round function is composed of involutory building blocks. It takes a 64-bit plaintext

Table 2: Notation.

Symbol	Meaning
$\oplus$	XOR operation.
$\|\|$	Concatenation of bits.
%	modulo operation.
T	The 64-bit tweak input.
K	The 128-bit master key.
TK_i	The main tweaks that are made based on the T and K ($i = 0, 1, 2, 3$).
$TK^i_{i\%4}$	The 64-bit round tweakey which is used in round $\mathcal{R}_i$ ($i = 0, \ldots, 31$) and $TK^i_{i\%4}[j]$ represents the j-th cell ($j = 0, \ldots, 15$) of $TK^i_{i\%4}$.
X^i	The internal state before the Mix-Columns (MC) at round $\mathcal{R}_i$ ($i = 0, \ldots, 31$) and $X^i[j]$ represents the j-th cell ($j = 0, \ldots, 15$) of X^i.
Y^i	The internal state before the PermuteNibbles (PN) at round $\mathcal{R}_i$ ($i = 0, \ldots, 31$) and $Y^i[j]$ represents the j-th cell ($j = 0, \ldots, 15$) of Y^i.
Z^i	The internal state before the S-boxes (SB) at round $\mathcal{R}_i$ ($i = 0, \ldots, 31$) and $Z^i[j]$ represents the j-th cell ($j = 0, \ldots, 15$) of Z^i.
ΓS	The linear mask of state S and $\Gamma S[j]$ represents the j-th cell ($j = 0, \ldots, 15$) of ΓS. When the state S is X^i, Y^i or Z^i we denote ΓS with ΓX^i, ΓY^i or ΓZ^i respectively.
ΔS	The differential in state S.
$\langle \cdot, \cdot \rangle$	Inner product.
$\bar{0}$	Zero vector.
$*$	An arbitrary value from $\mathbb{F}_2^4$.
$\mathtt{Y}$	Hexadecimal representation of arbitrary value $Y \in \mathbb{F}_2^4$, where we are using typewriter style.

$m = m_0 \| m_1 \| \cdots \| m_{14} \| m_{15}$ to initiate a 4×4 internal state $IS = I_0 \| I_1 \| \cdots \| I_{14} \| I_{15}$ as follows, where $I_i, m_i \in \mathbb{F}_2^4$:

$$IS = \begin{pmatrix} I_0 & I_1 & I_2 & I_3 \\ I_4 & I_5 & I_6 & I_7 \\ I_8 & I_9 & I_{10} & I_{11} \\ I_{12} & I_{13} & I_{14} & I_{15} \end{pmatrix} = \begin{pmatrix} m_0 & m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 & m_7 \\ m_8 & m_9 & m_{10} & m_{11} \\ m_{12} & m_{13} & m_{14} & m_{15} \end{pmatrix}$$

Then, the internal state is going through 32 rounds $\mathcal{R}_i, i \in 0, \cdots, 31$, to generate a 64-bit ciphertext. As is depicted in Figure 1, each round, excluding the last round, includes five functions, i.e., a binary MixColumn (`MC`), the round dependent combining with round constant AddRoundConstants (`ARC`), the round dependent mixing with the sub-tweakey AddTweakey (`ATK`), a nibble-based permutation PermuteNibbles (`PN`), and the substitution layer S-box (`SB`). The last round only includes `MC`, `ARC` and `ATK`, i.e., $\mathcal{R}_{31} = ATK_{31} \circ ARC_{31} \circ MC$, while for any $0 \leq i \leq 30$, $\mathcal{R}_i = SB \circ PN \circ ATK_i \circ ARC_i \circ MC$.

`MC` is a multiplication of internal state by the following binary matrix:

$$MC = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

After `MC`, in each round i two round dependent constant nibbles $a_i = (a_3^i, a_2^i, a_1^i, a_0^i)$ and $b_i = (b_2^i, b_1^i, b_0^i)$ are XOR-ed with I_4 and I_5 respectively (a_0^i and b_0^i are the least significant bits). A 4-bit LFSR and a 3-bit LFSR are used to update a and b for each round.

Table 3: The S-box used in `CRAFT` in hexadecimal form.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6

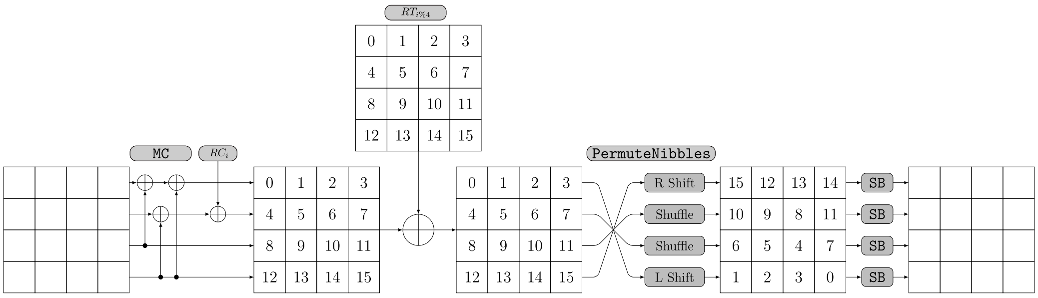


Figure 1: A round of `CRAFT`

Those LFSRs are initialized by values (0001) and (001), respectively and are updated to $a_{i+1} = (a_1^i \oplus a_0^i, a_3^i, a_2^i, a_1^i)$, and $b_{i+1} = (b_1^i \oplus b_0^i, b_2^i, b_1^i)$ from i-th round to $i+1$-th round.

After AddRoundConstants (`ARC`), a 64-bit round tweakey is XOR-ed with IS. The tweakey schedule of `CRAFT` is rather simple. Given the secret key $K = K_0 \| K_1$ and the tweak $T \in \{0,1\}^{64}$, where $K_i \in \{0,1\}^{64}$, four round tweakeys $TK_0 = K_0 \oplus T$, $TK_1 = K_1 \oplus T$, $TK_2 = K_0 \oplus Q(T)$ and $TK_3 = K_1 \oplus Q(T)$ are generated, where given $T = T_0 \| T_1 \| \cdots \| T_{14} \| T_{15}$, $Q(T) = T_{12} \| T_{10} \| T_{15} \| T_5 \| T_{14} \| T_8 \| T_9 \| T_2 \| T_{11} \| T_3 \| T_7 \| T_4 \| T_6 \| T_0 \| T_1 \| T_{13}$. Then at the round $\mathcal{R}_i$, $TK^i_{i\%4}$ is XOR-ed with the IS, where the rounds start from $i = 0$.

The next function is PermuteNibbles (`PN`) which is applying an involutory permutation P over nibbles of IS, where given $IS = I_0 \| I_1 \| \cdots \| I_{14} \| I_{15}$, $P(IS) = I_{15} \| I_{12} \| I_{13} \| I_{14} \| I_{10} \| I_9 \| I_8 \| I_{11} \| I_6 \| I_5 \| I_4 \| I_7 \| I_1 \| I_2 \| I_3 \| I_0$.

The final function is a non-linear 4×4-bit S-box which has been borrowed from `MIDORI` [BBI+15]. The table representation of the S-box is given in Table 3.

3 Related tweak zero-correlation and integral cryptanalysis

In this section, we apply the related tweak zero-correlation attack [ADG+19] to a reduced-round version of `CRAFT`. In the zero-correlation cryptanalysis of a tweakable block cipher $E_K(P, T)$, e.g. `CRAFT`, tweak bits can also be involved into the linear combination of input bits. Hence, in this case, when one looks for a linear hull with zero correlation, input mask consists of two components, one for plaintext, and another one for (master)-tweak. The correlation of a linear approximation with input mask (α_1, α_2), and output mask β, is calculated as follows:

$$corr((\alpha_1, \alpha_2), \beta) = 2 \Pr \left(\langle \alpha_1, P \rangle \oplus \langle \alpha_2, T \rangle \oplus \langle \beta, E_K(P, T) \rangle = 0 \right) - 1,$$

where the probability is taken over the all values of P, and T.

`CRAFT` has a linear twekey-scheduling $L_K : \mathbb{F}_2^{64} \to \left(\mathbb{F}_2^{64}\right)^{32}$, to map the tweak to the sub-tweakeys. The generated sub-tweakeys are then XORed to the internal states of the cipher as depicted in Figure 2. For a linear trail with input-output masks $((\alpha_1, \alpha_2), \beta)$, and internal linear masks $\Gamma = (\Gamma X^0, \Gamma Y^0, \Gamma Z^0, \Gamma X^1, \Gamma Y^1, \Gamma Z^1, \ldots, \Gamma X^{r-1}, \Gamma Y^{r-1}, \Gamma Z^{r-1}, \Gamma X^r)$,

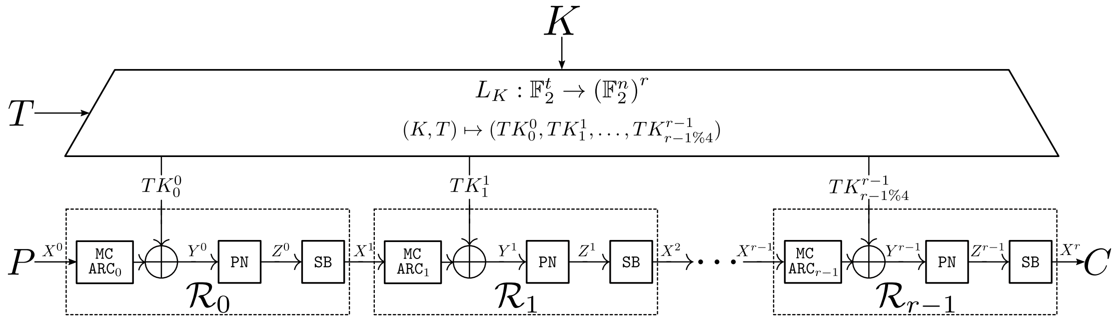


Figure 2: r rounds of CRAFT when $r < 32$

covering r rounds of CRAFT, correlation can be calculated as follows:

$$C_\Gamma = \prod_{i=0}^{r-1} corr((\Gamma X^i, \Gamma TK^i_{i\%4}), \Gamma X^{i+1}),$$

According to the rule of propagation of linear masks through XOR, linear mask ΓY^i must be the same as the linear mask $\Gamma TK^i_{i\%4}$, for all $0 \le i \le r-1$. According to the tweakey-scheduling of CRAFT, which is a linear mapping, the linear masks ΓY^i, for all $0 \le i \le r-1$, should satisfy the following relation:

$$\alpha_2 = \mathcal{L}(\Gamma Y^0, \ldots, \Gamma Y^{r-1}) := \bigoplus_{\substack{i=0,\\ i\%4<2}}^{r-1} \Gamma Y^i \oplus \bigoplus_{\substack{i=0,\\ i\%4\ge 2}}^{r-1} Q^{-1}\left(\Gamma Y^i\right).$$

In other words, there is a linear relation between nibbles of linear masks ΓY^i, for $0 \le i \le r-1$, as follows:

$$\alpha_2[j] = \bigoplus_{\substack{i=0,\\ i\%4<2}}^{r-1} \Gamma Y^i[j] \oplus \bigoplus_{\substack{i=0,\\ i\%4\ge 2}}^{r-1} \Gamma Y^i[Q^{-1}(j)], \text{ for all } 0 \le j \le 15.$$

The correlation of a linear hull, with the input linear masks (α_1, α_2) and the output linear mask β, can be calculated as follows:

$$corr((\alpha_1, \alpha_2), \beta) = \sum_{\substack{\Gamma X^0 = \alpha_1, \Gamma X^r = \beta,\\ (\Gamma X^1, \cdots, \Gamma X^{r-1}) \in (\mathbb{F}_2^{64})^{r-1}\\ \alpha_2 = \mathcal{L}(\Gamma Y^0, \ldots, \Gamma Y^{r-1})}} C_\Gamma.$$

The additional constraint $\alpha_2 = \mathcal{L}(\Gamma Y^0, \ldots, \Gamma Y^{r-1})$, which is induced by the tweakey-scheduling, introduces additional restriction on linear trails that are included in a linear hull. Hence, the probability of achieving a zero-correlation is higher than the single tweak zero-correlation cryptanalysis, where the tweakey-scheduling is not considered.

In the related tweak cases, the zero-correlation linear hull behavior of CRAFT is dependent on the starting round, i.e., the index of RT_i, $(i = 0, 1, 2, 3)$. Hence, we investigated the security of CRAFT against the related tweak zero-correlation attack in RT_0, RT_1, RT_2 and RT_3 modes. To find the related tweak zero-correlation trails, we modeled CRAFT in MILP to find a zero-correlation mask for RT_i and proved it manually. As a result, in the case of RT_0, we found a 14-round zero-correlation linear hull for CRAFT, where the number of forward and backward rounds are both 7. With respect to Figure 3, active linear masks are applied to two cells $X^0[4]$ and $X^0[12]$ at the input, and the active linear mask is applied to cell $X^{14}[4]$ in the state at the output. Then, we focus on the tweak cell labeled 11, where it is depicted by using a red frame in Figure 3. In the following section, based on the given active linear mask in the master tweak T, we present a 14-round related tweak zero-correlation for `CRAFT`:

$$\Gamma T = \bigoplus_{\substack{i=0,\\ i\%4<2}}^{r-1} \Gamma TK^i_{i\%4} \oplus \bigoplus_{\substack{i=0,\\ i\%4\geq 2}}^{r-1} Q^{-1}\left(\Gamma TK^i_{i\%4}\right) = \begin{pmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & \mathtt{8} \\ * & * & * & * \end{pmatrix}.$$

Note that the permutation Q operated on $TK^i_{i\%4}$, when $i = 2, 3, 6, 7, 10, 11$. Based on Figure 3, we have $\Gamma T[11] = \Gamma TK^5_1[11] \oplus \Gamma TK^6_2[8]$ (the XOR of red frames) and so,

$$\Gamma TK^5_1[11] \oplus \Gamma TK^6_2[8] = \mathtt{8}. \tag{1}$$

We denote the Linear Approximation Table of `CRAFT` S-box by LAT and $LAT[i][j]$ is the element of i-th row and j-th column of it and $LAT[i]$ is defined as the set $LAT[i] = \{j \in \mathbb{F}_2^4 | LAT[i][j] \neq 0\}$ (see Table 4). Now, based on the properties of PN and SB operations of 5-th round, we have

$$\Gamma X^6[0] \in LAT[\Gamma Y^5[15]], \tag{2}$$

Due to the `MC` operation on the active cells of column 3 of state X^5 in the input of 5-th round, we have

$$\Gamma Y^5[15] = \Gamma Y^5[11]$$

and so, based on (Equation 2), we have

$$\Gamma X^6[0] \in LAT[\Gamma Y^5[11]] = LAT[\Gamma TK^5_1[11]]. \tag{3}$$

Now, due to the `MC` operation on the active cells of column 0 of state X^6, we have

$$\begin{aligned} \Gamma TK^6_2[8] &= \Gamma X^6[0] \\ &\overset{(Equation\ 3)}{\in} LAT[\Gamma TK^5_1[11]]. \end{aligned} \tag{4}$$

Therefore, based on (Equation 1) and (Equation 4), $\Gamma TK^5_1[11]$ and $\Gamma TK^6_2[8]$ must satisfy the following conditions:

$$\begin{cases} \Gamma TK^5_1[11] \oplus \Gamma TK^6_2[8] = \mathtt{8}, \\ \Gamma TK^6_2[8] \in LAT[\Gamma TK^5_1[11]]. \end{cases}$$

These conditions are equivalent to finding an input mask x ($x = \Gamma TK^5_1[11]$) and an output mask y ($y = \Gamma TK^6_2[8]$), such that:

$$\begin{cases} x \oplus y = \mathtt{8}, \\ LAT[x][y] \neq 0. \end{cases}$$

Note that, by referring to linear approximation table of `CRAFT` S-box, we observe there is no input/output mask that satisfies these conditions (see Table 4).

We also searched the zero-correlation linear hulls for each cases RT_1, RT_2, and RT_3. For RT_1, we could not find a zero-correlation linear hull covering more than 13 rounds, but for both RT_2, and RT_3, we found new zero-correlation linear hulls covering 14 rounds of `CRAFT`.

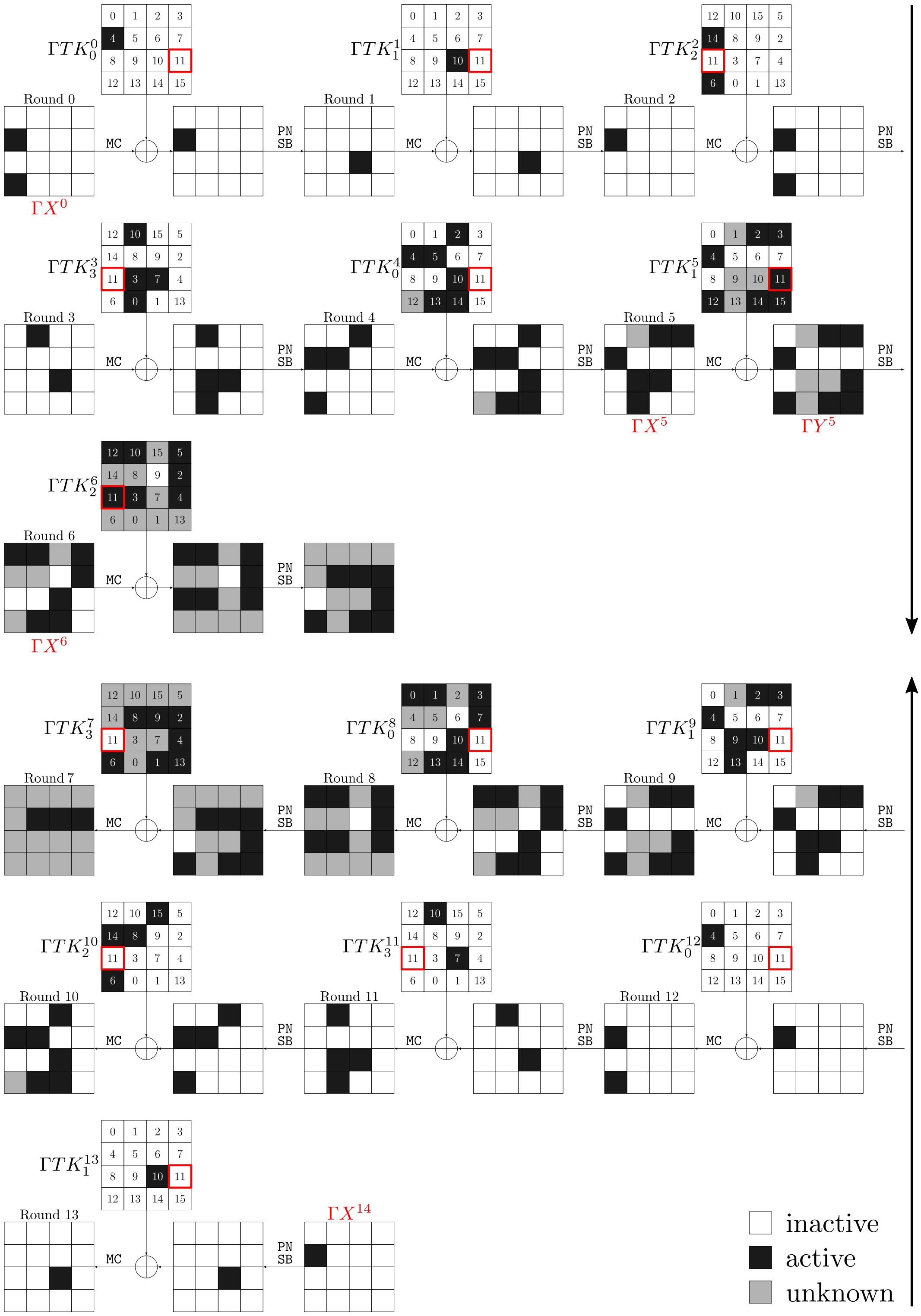


Figure 3: Related tweak zero-correlation of 14-round CRAFT in TK_0 mode

Table 4: Linear approximation table of `CRAFT` S-box.

x/y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	2	4	2	-2	0	2	0	-2	0	2	0	4	-2	0	-2
2	0	4	0	0	4	0	0	0	-4	0	0	0	0	4	0	0
3	0	2	0	2	-2	0	2	4	2	-4	-2	0	0	2	0	2
4	0	-2	4	-2	2	0	-2	0	-2	-4	-2	0	0	-2	0	2
5	0	0	0	0	0	0	0	0	0	0	-4	-4	0	0	4	-4
6	0	2	0	2	-2	0	2	-4	-2	0	-2	0	-4	-2	0	2
7	0	0	0	4	0	0	-4	0	0	0	0	-4	0	0	-4	0
8	0	-2	-4	2	-2	0	-2	0	-4	-2	0	2	2	0	2	0
9	0	0	0	-4	-4	0	0	0	-2	2	-2	-2	2	2	-2	2
A	0	2	0	-2	-2	-4	-2	0	0	-2	4	-2	-2	0	2	0
B	0	0	0	0	0	-4	0	-4	2	-2	-2	2	2	2	-2	-2
C	0	4	0	0	0	0	-4	0	2	2	-2	2	2	-2	2	2
D	0	-2	4	2	-2	0	-2	0	0	2	0	2	-2	4	2	0
E	0	0	0	0	0	4	0	-4	2	-2	2	-2	2	2	2	2
F	0	-2	0	2	2	-4	2	0	0	2	0	-2	2	0	2	4

The activity patterns of linear masks, for the obtained zero-correlation linear hulls in cases RT_2, and RT_3 are as follows:

$$\texttt{0000}\ \ \gamma\texttt{000}\ \ \texttt{0000}\ \ \texttt{0000} \xrightarrow{\text{14-round-}RT_2} \texttt{0000}\ \ \texttt{0}\delta\texttt{00}\ \ \texttt{0000}\ \ \texttt{0000},$$

$$\texttt{0000}\ \ \texttt{0}\gamma\texttt{00}\ \ \texttt{0000}\ \ \texttt{0000} \xrightarrow{\text{14-round-}RT_3} \texttt{0000}\ \ \delta\texttt{000}\ \ \texttt{0000}\ \ \texttt{0000},$$

where in both cases, $\Gamma T = ****\ \ ****\ \ ***\texttt{0}\ \ ****$, and γ, and δ are non-zero elements in $\mathbb{F}_2^4$.

3.1 Linking zero-correlation linear hull to integral

The following theorems show how to convert a zero-correlation linear hull to an integral distinguisher.

Theorem 1. *[SLR+15] Let $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a function, and A be a subspace of $\mathbb{F}_2^n$ and $\beta \in \mathbb{F}_2^n \setminus \{0\}$. Suppose that (α, β) is a zero-correlation linear approximation for any $\alpha \in A$, then for any $\lambda \in \mathbb{F}_2^n$, $\langle \beta, F(x + \lambda) \rangle$ is balanced on the following set*

$$A^{\perp} = \{x \in \mathbb{F}_2^n | \langle \alpha, x \rangle = 0, \alpha \in A\}.$$

The following theorem shows that the input masks should not necessarily form a subspace.

Theorem 2. *[SLR+15] A nontrivial zero-correlation linear hull of a block cipher always implies the existence of an integral distinguisher.*

The number of the required data to verify whether $\langle \beta, F(x + \lambda)$ in Theorem 1, and Theorem 2, is balanced over $A^{\perp}$, is equal to the cardinality of $A^{\perp}$ which is $2^{\dim(A^{\perp})}$. Therefore, if the input-size of F is n bits, and the dimension of the subspace A is m, the data complexity of the corresponding integral distinguisher is 2^{n-m}. Considering the tweak in the zero-correlation linear hull on a general tweakable block cipher may expand the domain space form n to $n + t$, when n, and t, are data-size and tweak-size respectively [ADG+19], but considering tweak in our related-tweak zero-correlation linear hulls for `CRAFT` increases the domain space n only by 4.

The `CRAFT`'s tweakey scheduling algorithm never mixes the different nibbles, and as mentioned above, the tweak, excluding the nibble $T[11]$, is independent of the obtained linear hull in our zero-correlation linear hulls for all cases RT_0, RT_2, and RT_3, and it actually can take any (arbitrary) constants. Therefore, the domain space of our zero-correlation linear hulls is $64 + 4 = 68$ bits instead of 128 bits. In other words, to evaluate the correlation of the obtained linear hull in the online phase, an arbitrary constant is taken for those nibbles labeled by $*$, and the inputs are chosen so that the vector consisting of 17 remaining nibbles, take all the possible values, since the correlation of our linear hulls is equal to zero, independent of those nibble labeled by $*$.

Suppose that we denote the 14 rounds of `CRFAT` starting with RT_0, as follows:

$$E_K : \mathbb{F}_2^{64} \times \mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$$
$$(P, T) \mapsto E_K(P, T),$$

where P and T denote plaintext and tweak, respectively. We also denote the function obtained by fixing 15 nibbles of tweak, excluding the cell 11, by an arbitrary value from $\mathbb{F}_2^{60}$ in function E_k by F, which is actually a function from $\mathbb{F}_2^{64} \times \mathbb{F}_2^4$ to $\mathbb{F}_2^{64}$. Let M be the set of all input masks in our zero-correlation linear hull in case RT_0, as follows:

$$M := \{(\gamma_0, \ldots, \gamma_{15}, \gamma) \in (\mathbb{F}_2^4)^{17} | \gamma_4 = \gamma_{12} \neq 0, \gamma = \mathtt{8}, \gamma_i = 0 \text{ for all } i \neq 4, 12\},$$

where $(\gamma_0, \ldots, \gamma_{15})$ corresponds to input mask for plaintext, and γ corresponds to the input mask for $T[11]$. Although, M is not a subspace of $\mathbb{F}_2^{68}$, for each $\alpha = (\gamma_0, \ldots, \gamma_{15}, \gamma) \in M$, if $A = \{\bar{0}, \alpha\}$, then A is a subspace of dimension 1 of $\mathbb{F}_2^{68}$. Suppose that β is chosen from the set of output masks of our zero-correlation linear hull for 14 rounds of `CRAFT` in case RT_0 which is depicted in Figure 3. Thus, based on Theorem 1, for each $\lambda \in \mathbb{F}_2^{68}$, $\langle \beta, F(x + \lambda) \rangle$ is balanced over $A^{\perp}$. Since $\dim(A^{\perp}) = 67$, the data complexity of the integral distinguisher corresponding to the zero-correlation linear hull covering 14 rounds, in case RT_0 is equal to 2^{67}. For more details, A, $A^{\perp}$ can be displayed as follows:

$$A = \{\bar{0}, (0, \ldots, 0, c_4, 0, \ldots, 0, c_{12}, 0, 0, 0, \mathtt{8})\},$$

where $c_4 = c_{12}$ are non-zero constants from F_2^4, and,

$$A^{\perp} = \{(x_0, \ldots, x_{15}, t_{11}) \in (\mathbb{F}_2^4)^{17} | \langle c_4, x_4 \rangle \oplus \langle c_{12}, x_{12} \rangle \oplus \langle \mathtt{8}, t_{11} \rangle = 0\}.$$

The required data for our integral distinguisher must be taken form $A^{\perp}$, such that $(x_0, \ldots, x_{15})$ corresponds to the plaintext and t_{11} corresponds to cell 11 of tweak. To generate the vectors of $A^{\perp}$, we can choose an arbitrary value for t_{11} at first, and then choose a suitable value for $(x_0, \ldots, x_{15})$, such that vector $(x_0, \ldots, x_{15}, t_{11})$ is in $A^{\perp}$. Since, there are 2^4 possible values for t_{11}, and for each of them there are 2^{63} plaintexts, the total data complexity is 2^{67}.

The zero-correlation linear hulls covering 14 rounds of `CRAFT` in the related-tweak model for cases RT_2, and RT_3 can also be converted to the integral distinguishers in a similar manner. In case RT_2, we apply any same linear mask to two cells 4, and 12, and apply zero linear masks to the remaining 14 nibbles. We also apply linear mask `0` to the cell 11 of tweak. In contrast to case RT_0, the set of all input masks in case RT_2 is a subspace of $\mathbb{F}_2^{68}$ with dimension 4 which is again denoted by A. Thereby, $\dim(A^{\perp}) = 68 - 4 = 64$, and the data complexity of the corresponding integral distinguisher is equal to 2^{64}, or equivalently, 2^4 tweaks, and for each of them 2^{60}, plainetexts are required. The integral distinguishers share the same input linear mask, and the cell 5 of the output is balanced. Due to the high similarity between zero-correlation linear hulls for cases RT_2, and RT_3, the data complexity of the related-tweak integral distinguisher corresponding to case RT_3 is exactly the same as the case RT_2, and has the same input, and output linear masks as the zero-correlation linear hulls obtained for 14 rounds in case RT_3.

4 Differential effect cryptanalysis

The designers of CRAFT provided extensive security analysis against differential and linear cryptanalysis [BLMR19, See Table 5]. They have provided the minimum number of active S-boxes for differential/linear cryptanalysis in single and differential related tweak mode. In addition, they have provided their analysis for differential effect (resp. linear hull) of round reduced CRAFT. In single tweak mode (ST-mode), they presented a differential distinguisher for 9 and 10 rounds of the cipher with the lower bounds of probabilities $2^{-54.67}$ and $2^{-62.61}$, respectively. For related tweak mode (RT-mode), depending on the starting round based on the TK value, they have presented 15, 16, 17, and 16 rounds differential distinguisher when the cipher is started from round 0, 1, 2, and 3, respectively (denoted as RT_0, RT_1, RT_2 and RT_3 respectively). The probability of the presented distinguisher are $2^{-55.14}$, $2^{-57.18}$, $2^{-60.14}$, and $2^{-55.14}$, respectively.

Table 5: Optimum differential/linear trails for reduced CRAFT in different model, where for each model, the upper row determines the minimum number of active S-boxes and the lower row shows the $-\log_2 P$, and also P denotes the probability of the best-found trail.

Model	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Linear	1	2	4	6	10	14	20	26	32	36	40	44	48	52	56	60	64
$-\log_2$	2	4	8	12	20	28	40	52	64	72	80	88	96	104	112	120	128
ST Diff.	1	2	4	6	10	14	20	26	32	36	40	44	48	52	56	60	64
$-\log_2$	2	4	8	12	20	28	40	52	64	72	80	88	96	104	112	120	128
RT_0 Diff.	0	1	2	4	6	12	14	19	22	25	27	32	36	38	40	46	49
$-\log_2$	1	2	4	8	12	24	28	38	44	50	54	64	72	76	80	92	98
RT_1 Diff.	0	1	2	5	7	10	15	18	22	24	28	32	35	38	43	45	46
$-\log_2$	1	2	4	10	14	20	30	36	44	48	56	64	70	76	86	90	92
RT_2 Diff.	0	1	2	4	6	12	16	19	21	24	27	30	34	39	41	42	44
$-\log_2$	1	2	4	8	12	24	32	38	42	48	54	60	68	78	82	84	88
RT_3 Diff.	0	1	2	5	7	10	15	18	21	24	28	31	34	38	39	41	47
$-\log_2$	1	2	4	10	14	20	30	36	42	48	56	62	68	76	78	82	94

To verify their results, first, we developed an automated tool, based on MILP and CryptoSMT. In the ST-mode, we reached the same number of active S-boxes, but an interesting observation was finding trails with optimum probability for any number of round and in any analysis mode, i.e., all S-boxes are activated by the maximum possible probability, i.e., 2^{-2} in differential/linear cryptanalysis (we only found a typo for their report of 17 rounds of RT_1, which was reported to be 44 S-boxes, while it should be 46). Table 5 represents the minimum number of active S-boxes and also the maximum probability of a single trail for the different number of rounds in different mode of analysis.

Next, we evaluated the differential effect of the cipher in ST-mode. To enumerate the differential trails in a differential effect of CRAFT, similar to previous works [LWR16, KLT15], we used the following approach to enumerate all the solutions in a SAT solver:

1. Build the CNF model for the problem, ask the solver to give one solution x if it exists.

2. Add a new condition to the current CNF model in order to remove x.

3. Ask the solver to give a solution, repeat step 2 until the solver returns unsatisfiable.

4.1 Differential effect

In this section, we evaluated the differential effect behavior of CRAFT, by fixing the input and output difference and try to find a better differential probability. We observed that for input/output differences that satisfies a trail with minimum number of active S-boxes, there are many trails with optimum probability and all of them have an identical truncated pattern. While finding an estimation of the real differential behavior of a cipher could be a very time consuming task in general, this observation motivated us to use the following steps to provide a lower bound on the differential probability of `CRAFT` for different number of rounds:

1. Using MILP, find a truncated differential trail with the minimum number of active S-boxes.
2. Verify the correctness of the truncated differential trail by finding at least one trail that matches the found truncated patterns.
3. Based on the found trail, develop the constraints for CryptoSMT, to limit the search to the truncated pattern with fixed input/output in the previous step.

CryptoSMT, supports primitives with S-boxes [AK18], but it uses a naive approach to encode S-boxes. In the SMT model generated by CryptoSMT, input and output differences of each n-bit S-box, are represented by n binary variables $x = (x_0, \ldots, x_{n-1})$, and $y = (y_0, \ldots, y_{n-1})$, respectively. It also introduces additional variables $p = (p_0, \ldots, p_{n-1})$ for each S-box S, representing the probability of the transition $x \xrightarrow{S} y$, which are linked to the $\Pr\{x \xrightarrow{S} y\}$, by the following relation:

$$wt(p_0, \ldots, p_{n-1}) = -\log_2(\Pr\{x \xrightarrow{S} y\}),$$

where $wt(p_0, \ldots, p_{n-1})$ denotes the Hamming weight of binary code $p_0 \ldots p_3$, and is called the weight of the transition $x \to y$. For example, the entries of $\{2^{-3}, 2^{-2}, 2^{-1}, 1\}$ can be encoded as follows:

Pr	2^{-3}	2^{-2}	2^{-1}	1
$p_0p_1p_2p_3$	0111	0011	0001	0000

In order to generate the constraints of each S-box, CryptoSMT first finds the set of all $3n$-tuple $(a_0, \ldots, a_{n-1}, b_0, \ldots, b_{n-1}, c_0, \ldots, c_{n-1}) \in \mathbb{F}_2^{3n}$ corresponding to the non-zero entries of DDT. Therefore, each $3n$-tuple out of the obtained set corresponds to an invalid assignment for $(x_0, \ldots, x_{n-1}, y_0, \ldots, y_{n-1}, p_0, \ldots, p_{n-1})$.

Then CryptoSMT generates a CNF for each S-box, as a constraint which is satisfiable if and only if the assignment corresponds to a valid trail. In order to generate the CNF of each S-box, it considers all invalid assignments. If an assignment $(a_0, \ldots, a_{n-1}, b_0, \ldots, b_{n-1}, c_0, \ldots, c_{n-1})$ is an impossible one, then the following clause is added to the CNF:

$$\begin{aligned} C = & L(a_0, x_0) \vee \cdots \vee L(a_{n-1}, x_{n-1}) \vee \\ & L(b_0, y_0) \vee \cdots \vee L(b_{n-1}, y_{n-1}) \vee \\ & L(c_0, p_0) \vee \cdots \vee L(c_{n-1}, p_{n-1}), \end{aligned}$$

where

$$L(s_i, t_i) = \begin{cases} t_i & if \ s_i = 0 \\ \neg t_i & if \ s_i = 1, \end{cases}$$

to exclude the invalid assignment (a,b,c), from the solution space. For example, if $(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0)$, is an invalid assignment for the variables $(x_0, \ldots, x_3, y_0, \ldots, y_3, p_0, \ldots, p_3)$, then the following clause is added to the CNF of the S-box in the SMT model.

$$(\neg x_0 \vee x_1 \vee \neg x_2 \vee \neg x_3 \vee y_0 \vee y_1 \vee y_2 \vee y_3 \vee p_0 \vee p_1 \vee p_2 \vee p_3).$$

By considering all invalid assignments, the CNF modeling the differential behaviour of a n-bit S-box is as follows:

$$\bigwedge_{i=1}^{m}\left(\bigvee_{j=0}^{n-1} L_j(s_i, t_i)\right).$$

The entries in the DDT of a 4-bit S-box with differential uniformity 4, including `CRAFT`'s S-box, only take four possible values, which are 0, 2, 4, and 16; therefore, the possible differential probabilities are 0, 2^{-3}, 2^{-2}, and 1, respectively. In contrast to the CryptoSMT's encoding, which always uses four variables to encode the probabilities of a given 4-bit S-box, the `CRAFT`'s S-box probabilities can be encoded via only three binary variables denoted as p_0, p_1, p_2, such that $wt(p_0, p_1, p_2) = -\log_2(p)$.

With the aim of optimizing the CryptoSMT's method for encoding the differential behavior of the `CRAFT`'s S-box, we use a different method than the CryptoSMT's original method, which can be easily generalized for an arbitrary n-bit S-box. We first generate the truth table of the following 11-bit boolean function [SWW18]:

$$\begin{aligned}
&f(x,y,p) = 0 && if\ \Pr\{x \to y\} = 0,\\
&f(x,y,p) = \begin{cases} 1 & p = (1,1,1) \\ 0 & o.w \end{cases} && if\ \Pr\{x \to y\} = 2^{-3},\\
&f(x,y,p) = \begin{cases} 1 & p = (0,1,1) \\ 0 & o.w \end{cases} && if\ \Pr\{x \to y\} = 2^{-2},\\
&f(x,y,p) = \begin{cases} 1 & p = (0,0,0) \\ 0 & o.w \end{cases} && if\ \Pr\{x \to y\} = 1,
\end{aligned}$$

where $x = (x_0, \ldots, x_3)$, and $y = (y_0, \ldots y_3)$ denote the input and the output differences, and $p = (p_0, p_1, p_2)$ is used to encode $\Pr\{x \to y\} = 2^{-wt(p)}$. To generate the constraints that model the differential behavior of S-box, we use the minimized product-of-sum representation of the above boolean function, which can be obtained via the Quine-McCluskey[Qui52, Qui55, MJ56], and Espresso algorithm [BHMSV84] implemented at the off-the-shelf program Logic Friday[Log19]. The minimized product-of-sum representation of the above boolean function for the `CRAFT`'s S-box is represented in Appendix A.

Following the above steps, we were able to accelerate the time of differential search for reduced rounds `CRAFT`. For instance, using the un-opimized CryptoSMT, finding a bound for differential of 11 rounds of `CRAFT` costed 86379s on a personal computer (Intel Core (TM)i-5, 8 Gig RAM, running Ubuntu 18.04 LTS), were we reached $2^{-58.7704}$ based on 2458966 trails (all with optimum probability of 2^{-80}). After optimizing CryptoSMT as above, we reached the identical probability much faster. A comparison of the search time to find the best single differential characteristic for reduced rounds variants of `CRAFT` is provided in Table 8, and Table 9 of Appendix A. Based on this approach, for 9 rounds of `CRAFT`, we find the following input/output difference with the differential probability of $2^{-44.37}$, where the least significant nibble appears in the left most position:

$$\texttt{7F0F 7F00 0000 7F00} \xrightarrow{\text{9-round;}\quad \Pr \geq 2^{-44.37}} \texttt{0A00 0000 0000 00DF}.$$

The above differential contains 810592 trails, all with probability 2^{-64} that have been found in 5417s on the above mentioned PC. It has an advantage of $2^{10.3}$ compared to the distinguisher provided by the designers for the same number of rounds. It should be noted that the presented bound is only the lower bound, given that we limited our searches to optimum trails and a specific truncated differential pattern. In addition, given a truncated differential pattern that minimizes the number of active S-boxes for a specific number of rounds, different trails with different input/output can be presented that satisfy the optimum probability. In the above search, we randomly selected one of them (the first optimum trail which is found by the tool) and bounded its lower-bound of differential. However, it may be possible to find a better bound for that number of rounds using another input/output difference or considering other possibilities too, e.g., non-optimum patterns. For example, for 9 rounds, we changed all active nibbles of the input and the output differences of the above-mentioned trail to `A` (it is represented in hexadecimal format) and observed a considerable improvement. To be more precise, for the bellow difference we found 2024500 optimum trails, before interrupting the run due to the RAM limitation:

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{9-round;} \quad \Pr \geq 2^{-43.051}} \texttt{0A00 0000 0000 00AA}.$$

In the case of 10 rounds, with the input difference "`0AAA 00AA 0000 00AA`" and the output difference "`0A00 0000 0000 00AA`", using a `G9 Hp` server with 32 Gig RAM and Windows 10 x64 as the operating system, we were able to observe 3513898 optimal trails in 4 days, before interrupting the run, which provides the probability of the 10-round distinguisher to be at least $2^{-50.2554}$.

Although the above mentioned approach provides advantage over naive search, using the same computer system, to extend this approach to more number of rounds, e.g., 12 rounds and more, it was very time consuming. Hence, we used another approach. We observed that it is possible to come up with expendable truncated trails for even (started from 8) and odd (started from 9) rounds of the cipher. Interestingly, this trails match the optimum number of active S-boxes for $9 \leq r \leq 17$, we did not check for $r > 17$. Figure 4 and Figure 5 represent the details of the construction of those trails. Moreover, setting active nibbles of input and output differences of each trail to `A`, provides us with a valid optimum trail. Hence, denoting the probability of an optimum trail for r-round of the cipher by $pr_c^{o,r}$, the trail bellow is valid for any even round-$r > 8$:

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{r-round;} \quad \Pr_c^{o,r} = 2^{-(56+8(r-8))}} \texttt{0A00 0000 0000 00AA}.$$

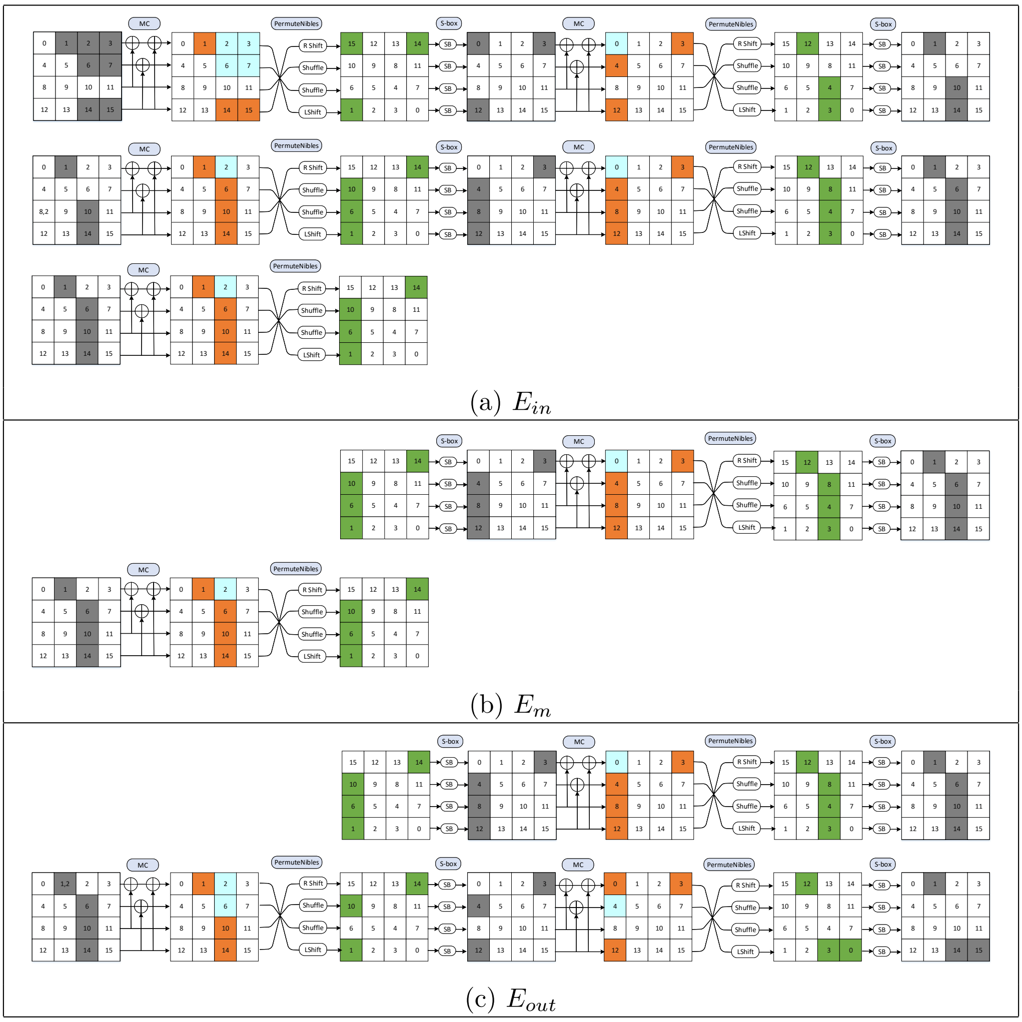
For an odd round-$r > 8$, the differential trail will be as follows:

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{r-round;} \quad \Pr_c^{o,r} = 2^{-(64+8(r-9))}} \texttt{0A00 0000 0000 00AA}.$$

Any of Figure 4 and Figure 5 includes three partitions, denoted by $E_{in,r_{in}}$, E_{m,r_m} and $E_{out,r_{out}}$, where r_x is an integer which is used to indicate the number of rounds in a partition. From now on, $E^{even}_{in,r_{in}}$, E^{even}_{m,r_m} and $E^{even}_{out,r_{out}}$ and E^{odd}_{in}, E^{odd}_{m} and E^{odd}_{out} denote partitions of the cipher in Figure 4 and Figure 5 respectively, while $E^{even/odd}_{in,r_{in}}$, $E^{even/odd}_{m,r_m}$ and $E^{even/odd}_{out,r_{out}}$ denote the their partitions. Hence, to design 10-round and 12-round trails, we respectively can use the structures $E^{even}_{out,4} \circ E^{even}_{m,2} \circ E^{even}_{in,4}$ and $E^{even}_{out,4} \circ E^{even}_{m,2} \circ E^{even}_{m,2} \circ E^{even}_{in,4}$ while to design 9-round and 11-round trails, we can use the structures $E^{odd}_{out,5} \circ E^{odd}_{in,4}$ and $E^{odd}_{out,5} \circ E^{odd}_{m,2} \circ E^{odd}_{in,4}$, respectively. It is also possible to use other combinations. For example, we can construct other 12-round trails as $E^{even}_{out,4} \circ E^{even}_{m,4} \circ E^{even}_{in,4}$, $E^{even}_{out,6} \circ E^{even}_{m,2} \circ E^{even}_{in,4}$ and $E^{even}_{out,6} \circ E^{even}_{in,6}$ also, where $E^{even}_{m,4} \equiv E^{even}_{m,2} \circ E^{even}_{m,2}$, $E^{even}_{in,6} \equiv E^{even}_{m,2} \circ E^{even}_{in,4}$ and $E^{even}_{out,6} \equiv E^{even}_{out,4} \circ E^{even}_{m,2}$.

It is worth noting, for the trails presented in Figure 4 and Figure 5, the output differences (nibbles) of E^{even}_{out} and E^{odd}_{out} are identical and the input differences (nibbles) of E^{even}_{in} can be matched to the input of E^{odd}_{in} by two nibbles rotation to right in each row.

On the other hand, from the DDT of the `CRAFT`'s S-box (Table 6), one can observe that if $x \in \{\texttt{5,7,A,D,F}\}$, then we can find at least one entry $y \in \{\texttt{5,7,A,D,F}\}$ such that $s(x) = y$ with probability 2^{-2} and for any $z \notin \{\texttt{5,7,A,D,F}\}$ the probability of $s(x) = z$ will be upper-bounded by 2^{-3}. Moreover, we observed that any differential includes trails, where each active S-box is activated with the probability 2^{-2}, and we were not even able to count all of them using our computational resources, in the previous approach. These properties motivated us to do semi-truncated differential search for the different parts of



(a) E_{in}

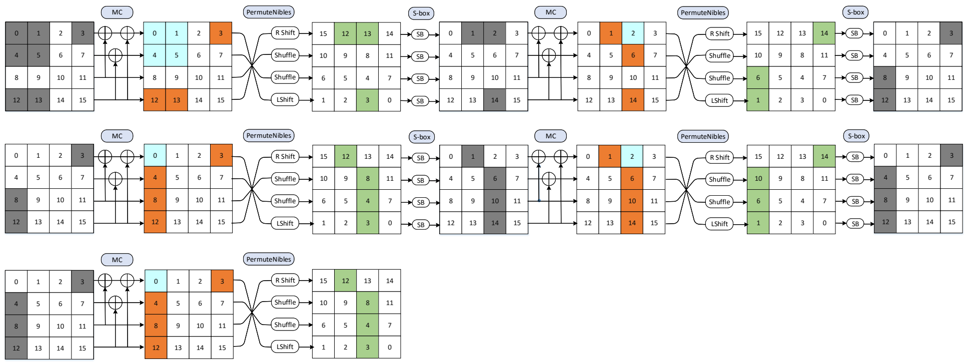
(b) E_m

(c) E_{out}

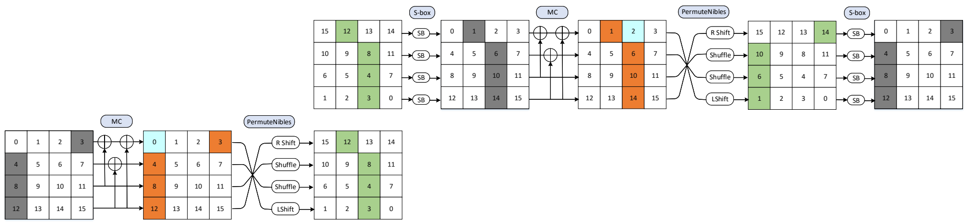
Figure 4: An expendable truncated trail for even rounds, where E_{in} and E_{out} denote the first 4 and the last 4 rounds, respectively and E_m is a repeatable 2-round truncated trail that can be used as much as required. For example, to design a 10-round trail, this stage is repeated once in the current trail. The Cyan-colored cells are inactive due to cancellation after `MC` step, white-colored cells are inactive, and {Gray, Orange, Green} colors are active cells in different stages of the cipher.

our models for even and odd rounds, i.e., $E^{even/odd}_{in,r_{in}}$, $E^{even/odd}_{m,r_m}$ and $E^{even/odd}_{out,r_{out}}$ in Figure 4 and Figure 5. For semi-truncated differential search, we programmed a model with the constraints bellow:

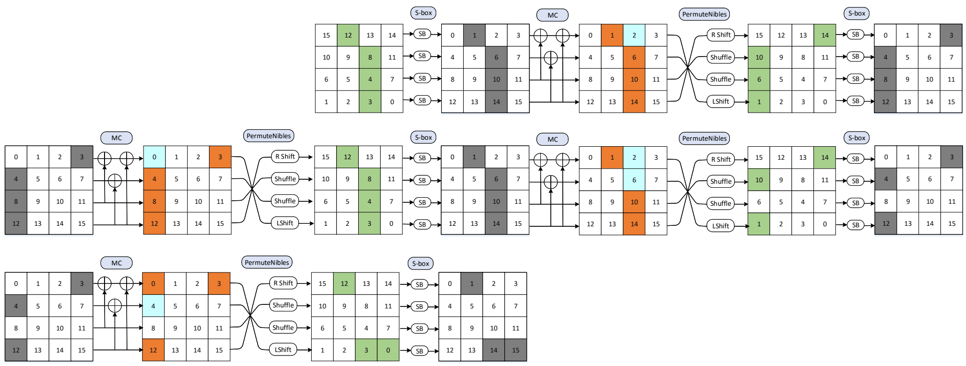
1. We set any active nibbles in the input of $E^{even/odd}_{in,r_{in}}$ and any active nibbles in the output of $E^{even/odd}_{out,r_{out}}$ to be `A`.
2. We limited any active intermediate nibble at the output of $E^{even/odd}_{in,r_{in}}$, the input/output of $E^{even/odd}_{m,r_m}$ and input of $E^{even/odd}_{out,r_{out}}$ to be in the set {`5`,`7`,`A`,`D`,`F`}.
3. We find the differential probability of all possible outputs of $E^{even/odd}_{in,,r_{in}}$, all possible inputs and outputs of $E^{even/odd}_{m,r_m}$ and all possible inputs of $E^{even/odd}_{out,r_{out}}$, concerns the above constraints up to where our programs could compute.



(a) E_{in}



(b) E_m



(c) E_{out}

Figure 5: An expendable truncated trail for odd rounds, where E_{in} and E_{out} denote the first 4 and the last 5 rounds, respectively and E_m is a repeatable 2-round truncated trail that can be used as much as required. For example, to design a 9-round trail, this stage is omitted.

For those constrains, it is trivial that we have only one possible difference for the input of $E^{even/odd}_{in,r_{in}}$ and one possible difference for the output of $E^{even/odd}_{out,r_{out}}$. To determine possible output-differences of $E^{even}_{in,4}$, we should consider the pattern before the last `MC`, i.e., X^4, and after the last `MC`, i.e., Y^4. It can be seen that to satisfy the truncated differential pattern, we should have $X^4[14] = X^4[10] \neq X^4[6]$. Hence, there are only $5 \times 5 \times 4 = 100$ possible values for Y^4 or outputs of $E^{even}_{in,4}$. A similar argument can be provided for the input/output differences of $E^{even/odd}_{m,r_m}$, and the input differences of $E^{even/odd}_{out,r_{out}}$. Therefore, there are only 100×100 possible values for input/output differences of $E^{even/odd}_{m,r_m}$ and 100 possible values for input/output differences of $E^{even/odd}_{out,r_{out}}$. In the next step, we need to determine the differential probability of any possible input/output differences for any

Table 6: Differential distribution table (DDT) of CRAFT S-box.

x/y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	2	4	0	2	2	2	0	2	0	0	0	0	0	2	0
2	0	4	0	0	4	0	0	0	0	4	0	0	4	0	0	0
3	0	0	0	0	2	0	4	2	2	2	0	0	0	2	0	2
4	0	2	4	2	2	2	0	0	2	0	0	2	0	0	0	0
5	0	2	0	0	2	0	0	4	0	2	4	0	2	0	0	0
6	0	2	0	4	0	0	0	2	2	0	0	0	2	2	0	2
7	0	0	0	2	0	4	2	0	0	0	0	2	0	4	2	0
8	0	2	0	2	2	0	2	0	0	2	0	2	2	0	2	0
9	0	0	4	2	0	2	0	0	2	2	0	2	2	0	0	0
A	0	0	0	0	0	4	0	0	0	0	4	0	0	4	0	4
B	0	0	0	0	2	0	0	2	2	2	0	4	0	2	0	2
C	0	0	4	0	0	2	2	0	2	2	0	0	2	0	2	0
D	0	0	0	2	0	0	2	4	0	0	4	2	0	0	2	0
E	0	2	0	0	0	0	0	2	2	0	0	0	2	2	4	2
F	0	0	0	2	0	0	2	0	0	0	4	2	0	0	2	4

partition of the cipher. We provide a horizontal vector containing 100 probabilities for $E^{even}_{in,4}$, a matrix containing 100×100 probabilities for $E^{even/odd}_{m,r_m}$ and a vertical vector containing 100 probabilities for $E^{even/odd}_{out,r_{out}}$. Given those probabilities, we can calculate the differential probability of any trail, it will be just multiplication of those joint probability vectors/matrices, which can be done very efficiently. To this end, we determined the joint probabilities vectors/matrices of all cipher's partitions of Figure 4 and Figure 5. The joint probability horizontal vector of $E^{even}_{in,4}$ includes 76 non-zero entries (out of 100) and it is identical to the joint probability vector derived for $E^{odd}_{in,4}$. The joint probability vertical vectors of both $E^{even}_{out,4}$ and $E^{odd}_{out,5}$ include 92 non-zero entries (each out of 100) and the joint probability matrices derived for $E^{even}_{m,2}$ and $E^{odd}_{m,2}$ also include 2734 non-zero entries (each out of 100×100). For each possible intermediate entry, e.g., an entry in the $E^{even}_{in,4}$ vector, we counted all the possible trails from the fixed input difference of $E^{even}_{in,4}$ to that possible difference of $E^{even}_{in,4}$, which can be directly used to determine the probability related to that entry. Next, we used those joint probabilities vectors/matrices to determine the differential effect of different round reduced variants of CRAFT; in all cases we extended the number of rounds by repeating $E^{even/odd}_{m,2}$ as many times as required:

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{9-round; } \Pr \geq 2^{-40.20}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{10-round; } \Pr \geq 2^{-45.12}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{11-round; } \Pr \geq 2^{-49.79}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{12-round; } \Pr \geq 2^{-54.72}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{13-round; } \Pr \geq 2^{-59.39}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{14-round; } \Pr \geq 2^{-64.32}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{15-round; } \Pr \geq 2^{-68.99}} \texttt{0A00 0000 0000 00AA}.$$

Through our analysis we also investigated the truncated differential behavior of optimum trails of fixed input/output differences. Interestingly, we observed that for any input/output differences with the optimum trails that we have checked (including the input/output differences of Figure 4 and Figure 5) the truncated pattern of all optimum trails of a fixed input/output difference is fixed. To verify this, for a given input/output difference for which there is an optimum trail, we forced the MILP and also SAT tools to finding an optimum trail with different truncated patterns. However, for all input/output differences that we checked, the programs returned infeasible. Hence, for any trail driven from Figure 4 or Figure 5, using our partitioning approach and the way that we have used to determine the probabilities of intermediate entries, we are able to count the exact number of the optimum trails for any number of rounds of `CRAFT`, starting from 9 and for the given differences; also, we can determine a lower bound of non-optimum trails. In the last column of Table 7, we reported the values of the optimum trails for the several numbers of the rounds.

On the other hand, for a fixed input/output difference, changing r_{in}, r_m, and r_{out}, has an influence on the number of non-optimum trails that are considered in the final differential effect. Hence, although the presented distinguishers are the best known distinguishers for the round reduced `CRAFT` in ST-mode, to improve the results more, we also evaluated other values for r_{in}, r_m and r_{out} (it is clear that extending the number of rounds of a partition increases the computational cost of producing the related joint probabilities matrix/vector). As a result, for $r_m = 4$ (for even/odd rounds) and $r_{out} = 6$ (for even rounds) we could improve the above bounds as follows:

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{10-round};\ \ \Pr \geq 2^{-44.89}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{12-round};\ \ \Pr \geq 2^{-54.48}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{13-round};\ \ \Pr \geq 2^{-59.13}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{0AAA 00AA 0000 00AA} \xrightarrow{\text{14-round};\ \ \Pr \geq 2^{-63.80}} \texttt{0A00 0000 0000 00AA},$$

$$\texttt{AA0A AA00 0000 AA00} \xrightarrow{\text{15-round};\ \ \Pr \geq 2^{-68.75}} \texttt{0A00 0000 0000 00AA},$$

where, as it is also depicted in Table 7, for the 14 rounds trail we used the combination $E^{even}_{out,6} \circ E^{even}_{m,4} \circ E^{even}_{in,4}$. The above distinguishers, to the best of our knowledge, are the best-known differential distinguishers for `CRAFT` in ST-model.

It should be noted that we also evaluated the differential effect when $r_{in} = 6$. However, it did not give better results.

Table 7: The values of r_{in}, r_m, and r_{out} of the best differential trails of `CRAFT` that we have found. Pr denotes the probability of the related trail.

♯ Rounds	r_{in}	r_m	r_{out}	Pr	♯ optimum trails
9	4	-	5	$2^{-40.20}$	$2^{23.32}$
10	4	-	6	$2^{-44.89}$	$2^{26.49}$
11	4	2	5	$2^{-49.79}$	$2^{29.66}$
12	4	2	6	$2^{-54.48}$	$2^{32.83}$
13	4	4	5	$2^{-59.13}$	$2^{36.00}$
14	4	4	6	$2^{-63.80}$	$2^{39.18}$

5 Discussion

Through our analysis, we observed some typos in the designers' analysis which reporting them could be useful for later analysis. We already mentioned one of them in Subsection 4.1, i.e., the minimum number of active S-boxes for 17 rounds in the case of RT1. In addition, the designers reported 12 zero-correlation masks for 13 rounds of the cipher. Although we found twelve zero-correlation linear hulls, based on our analysis with both MILP and SAT approaches, 2 of the reported masks are not valid, which are as follows, where γ and δ are non-zero masks in $\mathbb{F}_2^4$:

$$\texttt{0000 00}\gamma\texttt{0 0000 00}\gamma\texttt{0} \xrightarrow{\text{13-round}} \texttt{0000 }\delta\texttt{000 0000 0000},$$

$$\texttt{0000 }\gamma\texttt{000 0000 }\gamma\texttt{000} \xrightarrow{\text{13-round}} \texttt{0000 00}\delta\texttt{0 0000 0000}.$$

In order to verify this claim, for each one of the above linear hulls, a valid linear trail is displayed in Appendix B. We also found the following new zero-correlation linear hulls for 13 rounds of `CRAFT`, in ST-mode:

$$\texttt{0000 00}\gamma\texttt{0 0000 00}\gamma\texttt{0} \xrightarrow{\text{13-round}} \texttt{0000 00}\delta\texttt{0 0000 0000},$$

$$\texttt{0000 }\gamma\texttt{000 0000 }\gamma\texttt{000} \xrightarrow{\text{13-round}} \texttt{0000 }\delta\texttt{000 0000 0000}.$$

We also checked the validity of the reported input/output patterns for the impossible differential covering 13 rounds of `CRAFT`. We observed that two of the input/output patterns are not valid in this case too, which are as follows, where γ and δ are non-zero difference in $\mathbb{F}_2^4$:

$$\texttt{00}\gamma\texttt{0 0000 00}\gamma\texttt{0 0000} \xrightarrow{\text{13-round}} \texttt{0000 0000 }\delta\texttt{000 0000},$$

$$\gamma\texttt{000 0000 }\gamma\texttt{000 0000} \xrightarrow{\text{13-round}} \texttt{0000 0000 00}\delta\texttt{0 0000}.$$

For each one of the input/output patterns above, one possible differential trail is displayed in Appendix C, which proves our claim. We also found the following two new valid impossible differential input/outputs for 13 rounds of `CRAFT`, in ST-mode:

$$\texttt{00}\gamma\texttt{0 0000 00}\gamma\texttt{0 0000} \xrightarrow{\text{13-round}} \texttt{0000 0000 00}\delta\texttt{0 0000},$$

$$\gamma\texttt{000 0000 }\gamma\texttt{000 0000} \xrightarrow{\text{13-round}} \texttt{0000 0000 }\delta\texttt{000 0000}.$$

In the case of RT-model of differential, designers reported the masks bellow:

$$\texttt{0000 A000 0000 0000} \xrightarrow{\text{15-round-}RT_0;\ \ \Pr \geq 2^{-55.14}} \texttt{0000 0000 00A0 A000}$$

$$\texttt{0A0A 00AA 0000 000A} \xrightarrow{\text{16-round-}RT_1;\ \ \Pr \geq 2^{-57.18}} \texttt{0000 000A A000 0000},$$

$$\texttt{0000 0000 0000 0000} \xrightarrow{\text{17-round-}RT_2;\ \ \Pr \geq 2^{-60.14}} \texttt{0000 0000 00A0 A000},$$

$$\texttt{0000 0000 0000 AA00} \xrightarrow{\text{16-round-}RT_3;\ \ \Pr \geq 2^{-55.14}} \texttt{0000 0000 00A0 A000},$$

where in all cases, $\Delta T = \texttt{0000 0000 00A0 0000}$. However, for the provided difference for RT_1, RT_2 and RT_3, there are no trails for those differences with a reasonable number of active S-boxes. In addition, if the difference bellow is valid:

$$\texttt{0000 0000 0000 0000} \xrightarrow{\text{17-round-}RT_2;\ \ \Pr \geq 2^{-60.14}} \texttt{0000 0000 00A0 A000},$$

then, given that the input difference has no active nibble and in backward direction it first goes through S-box layers at the first, with probability 1. It is possible to present an 18 round trail for RT_1 with the same probability, i.e., $2^{-60.14}$. Hence, we also reevaluated the differential effect of `CRAFT` in RT-model, with the same $\Delta T = \texttt{0000 0000 00A0 0000}$. Our best results are as follows:

$$\texttt{0000 A000 0000 0000} \xrightarrow{\text{15-round-}RT_0;\ \ \Pr \geq 2^{-55.14}} \texttt{0000 0000 00A0 A000},$$

$$\texttt{000A 000A 0AA0 000A} \xrightarrow{\text{16-round-}RT_1;\ \ \Pr \geq 2^{-62.68}} \texttt{0000 0000 0000 0000},$$

$$\texttt{A000 0A00 0000 AA00} \xrightarrow{\text{17-round-}RT_2;\ \Pr \geq 2^{-60.14}} \texttt{0000 0000 00A0 A000},$$

$$\texttt{0AA0 0000 00A0 0000} \xrightarrow{\text{16-round-}RT_3;\ \Pr \geq 2^{-55.14}} \texttt{0000 0000 00A0 A000}.$$

It can be seen that we could find other input/output differences for RT_2 and RT_3 that have identical probabilities as the probabilities reported by the designers. In the case of RT_0, we received identical differential effect probability as the designers probability, for the same input/output differences. However, in the case of RT_1 we could not find such differences. This distinction between our result, and those of the designers, in the case of RT_1, motivated us to evaluate the differential effects for 15 and 17 rounds of this mode as follows :

$$\texttt{AAA0 0AA0 000A 0AA0} \xrightarrow{\text{15-round-}RT_1;\ \Pr \geq 2^{-56.31}} \texttt{0000 0000 0000 000A},$$

where $\Delta T =$ `0000 0000 000A 0000`, and,

$$\texttt{050A 000A 0AA0 000A} \xrightarrow{\text{17-round-}RT_1;\ \Pr \geq 2^{-65.34}} \texttt{0000 A000 0000 0000},$$

where $\Delta T =$ `0000 0000 00A0 0000`.

It should be noted we also evaluated the security of `CRAFT` against linear hull, following the same approach as the differential effect. However, we could not beat the designers' claim, which is $2^{-62.12}$ for 14- rounds of `CRAFT` in ST-model.

6 Conclusion

In this work, we provided a detailed analysis of `CRAFT` against differential and related tweak zero-correlation and integral cryptanalysis. Our related tweak zero-correlation and integral cryptanalysis, which cover 14 rounds, are the first analysis of `CRAFT` against this attack, given that the designers analyzed its security against single tweak zero-correlation and integral cryptanalysis. While we found 14-round distinguishers in the related tweak zero-correlation/integral cryptanalysis for cases RT_0, RT_2, and RT_3, we could not find any related tweak zero-correlation/integral distinguisher for case RT_1 for 14-rounds of the cipher.

Our differential analysis improved the designers' results significantly. For example, the designers' report include the lower bound of probability of differential effect for 10 rounds of the cipher in single tweak model to be $2^{-62.61}$ while we improved this bound and presented a differential distinguisher for the same number of rounds with probability $2^{-44.89}$ and a differential distinguisher for up to 14 rounds, with the probabilities beyond 2^{-64}. This analysis shows that there is a huge gap between the differential effect and any differential trails in the round reduced `CRAFT`, similar to some other lightweight block ciphers already mentioned in [AK18].

Through our differential analysis, we observed that for many fixed input/output differentials, `CRAFT` included very strong clusters of high-probable trails that helped us to improve the probability of our differential distinguishers significantly.

In our differential effect analysis of the even/odd number of rounds, we fixed the input/output masks for even/odd number of rounds, and provided extendable truncated differential trails for the cipher and then partitioned those trails to estimate the differential effect of the whole target rounds. This approach helped us estimate the differential effect of the cipher more efficiently (in term of time and the used resources), compared to naive approaches based on counting trails. Thanks to the fixed truncated differential pattern of `CRAFT` for all optimum trails of a fixed input/output mask, partitioning works well to bound its differential effect and we were able to provide the exact number of optimum trails for a given fixed input/output difference; and for any number of rounds, larger than 9, it can be done for any other input/output mask. As a future work, it is worth investigating whether there is any other cipher with the same differential behavior, i.e., fixed truncated differential for dominant trails. If there is, then it should be possible to use the partitioning approach to evaluate its security against differential effect. In addition, while our bound for the number of optimal trails for any fixed input/output mask is tight, we were not able to bound the exact number of non-optimum trails for the used masks. Hence, as another future work, it is possible to improve the reported differential effects considering some missing non-optimum trails in our analysis.

The designer stated [BLMR19, Sec. 5.4] *"For the key recovery the number of rounds that can be appended for an RT_i differential is at most $4+i$ rounds before and 7 rounds after the differential"*. However, given that the focus of this paper was to provide better distinguishers for CRAFT, we have not investigated the key recovery in this paper. Hence, as a future work, it worth to see how many rounds can be attacked based on the provided distinguishers in this paper.

Acknowledgments

The authors gratefully thank the use of School of Computer Science, Institute for Research in Fundamental Science (IPM) as the computations were partially done there. We would like to thank Christof Beierle for his invaluable comments and suggestions in preparing the final draft of this paper. We would like also to thank all the anonymous reviewers of ToSC for their valuable comments which helped us to improve the presentation of the work significantly. We would like also to thank Hadi Soleimany for his invaluable comments and suggestions in preparing this paper. We would like also to thank the designers team of `CRAFT`, specially Shahram Rasoolzadeh, for their valuable feedback during our communications. The work of the second author is supported by the ANR (DeCrypt ANR-18-CE39-0007). The third author was supported in part by the Iran National Science Foundation (INSF) under contract No. 96/53979. The fourth author was supported in part by the National Natural Science Foundation of China (Grants No. 61802399, 61732021). The fifth author was supported in part by the Iran National Science Foundation (INSF) under contract No. 98010674.

References

[ADG+19] Ralph Ankele, Christoph Dobraunig, Jian Guo, Eran Lambooij, Gregor Leander, and Yosuke Todo. Zero-correlation attacks on tweakable block ciphers with linear tweakey expansion. *IACR Trans. Symmetric Cryptol.*, 2019(1):192–235, 2019.

[AHMN13] Jean-Philippe Aumasson, Luca Henzen, Willi Meier, and María Naya-Plasencia. Quark: A lightweight hash. *J. Cryptology*, 26(2):313–339, 2013.

[AK18] Ralph Ankele and Stefan Kölbl. Mind the gap-a closer look at the security of block ciphers against differential cryptanalysis. In *International Conference on Selected Areas in Cryptography*, pages 163–190. Springer, 2018.

[BBI+15] Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni. Midori: A block cipher for low energy. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, volume 9453 of *Lecture Notes in Computer Science*, pages 411–436. Springer, 2015.

[BBK+13] Begül Bilgin, Andrey Bogdanov, Miroslav Knezevic, Florian Mendel, and Qingju Wang. Fides: Lightweight authenticated cipher with side-channel resistance for constrained hardware. In Guido Bertoni and Jean-Sébastien Coron, editors, *Cryptographic Hardware and Embedded Systems - CHES 2013 - 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings*, volume 8086 of *Lecture Notes in Computer Science*, pages 142–158. Springer, 2013.

[BHMSV84] Robert K Brayton, Gary D Hachtel, Curt McMullen, and Alberto Sangiovanni-Vincentelli. *Logic minimization algorithms for VLSI synthesis*, volume 2. Springer Science & Business Media, 1984.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY family of block ciphers and its low-latency variant MANTIS. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.

[BKL+07] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In 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.

[BLMR19] Christof Beierle, Gregor Leander, Amir Moradi, and Shahram Rasoolzadeh. CRAFT: lightweight tweakable block cipher with efficient protection against DFA attacks. *IACR Trans. Symmetric Cryptol.*, 2019(1):5–45, 2019.

[BSS+15] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The SIMON and SPECK lightweight block ciphers. In *Proceedings of the 52nd Annual Design Automation Conference, San Francisco, CA, USA, June 7-11, 2015*, pages 175:1–175:6. ACM, 2015.

[EY19] Muhammad ElSheikh and Amr M. Youssef. Related-key differential cryptanalysis of full round CRAFT. *IACR Cryptology ePrint Archive*, 2019:932, 2019.

[GGNS13] Benoît Gérard, Vincent Grosso, María Naya-Plasencia, and François-Xavier Standaert. Block ciphers that are easier to mask: How far can we go? In Guido Bertoni and Jean-Sébastien Coron, editors, *Cryptographic Hardware and Embedded Systems - CHES 2013 - 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings*, volume 8086 of *Lecture Notes in Computer Science*, pages 383–399. Springer, 2013.

[GLSV14] Vincent Grosso, Gaëtan Leurent, François-Xavier Standaert, and Kerem Varici. Ls-designs: Bitslice encryption for efficient masked software implementations. In Carlos Cid and Christian Rechberger, editors, *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*, volume 8540 of *Lecture Notes in Computer Science*, pages 18–37. Springer, 2014.

[GPP11] Jian Guo, Thomas Peyrin, and Axel Poschmann. The PHOTON family of lightweight hash functions. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 222–239. Springer, 2011.

[HMS] Trevor Hansen, Norbert Manthey, and Mate Soos. Stp in the smtcomp 2019. https://stp.github.io/.

[ISSK09] Maryam Izadi, Babak Sadeghiyan, Seyed Saeed Sadeghian, and Hossein Arabnezhad Khanooki. MIBS: A new lightweight block cipher. In Juan A. Garay, Atsuko Miyaji, and Akira Otsuka, editors, *Cryptology and Network Security, 8th International Conference, CANS 2009, Kanazawa, Japan, December 12-14, 2009. Proceedings*, volume 5888 of *Lecture Notes in Computer Science*, pages 334–348. Springer, 2009.

[KLT15] Stefan Kölbl, Gregor Leander, and Tyge Tiessen. Observations on the simon block cipher family. In *Annual Cryptology Conference*, pages 161–185. Springer, 2015.

[Köl19] Stefan Kölbl. CryptoSMT an easy to use tool for cryptanalysis of symmetric primitives based on SMT/SAT solvers. available on-line, 2019. https://github.com/kste/cryptosmt.

[Log19] Logic Friday. Solve digital logic circuits based on IC packages with logic functions. available on-line, Last access 8/23/2019. https://download.cnet.com/developer/logic-friday.

[LWR16] Yunwen Liu, Qingju Wang, and Vincent Rijmen. Automatic search of linear trails in arx with applications to speck and chaskey. In *International Conference on Applied Cryptography and Network Security*, pages 485–499. Springer, 2016.

[MA19] AmirHossein E. Moghaddam and Zahra Ahmadian. New automatic search method for truncated-differential characteristics: Application to midori, skinny and craft. Cryptology ePrint Archive, Report 2019/126, 2019. https://eprint.iacr.org/2019/126.

[MJ56] Edward J McCluskey Jr. Minimization of boolean functions. *Bell system technical Journal*, 35(6):1417–1444, 1956.

[MWGP11] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In *International Conference on Information Security and Cryptology*, pages 57–76. Springer, 2011.

[NPB15] Aina Niemetz, Mathias Preiner, and Armin Biere. Boolector 2.0. *Journal on Satisfiability, Boolean Modeling and Computation*, 9:53–58, 2015.

[Qui52] Willard V Quine. The problem of simplifying truth functions. *The American mathematical monthly*, 59(8):521–531, 1952.

[Qui55] Willard V Quine. A way to simplify truth functions. *The American Mathematical Monthly*, 62(9):627–631, 1955.

[SBD+18] Thierry Simon, Lejla Batina, Joan Daemen, Vincent Grosso, Pedro Maat Costa Massolino, Kostas Papagiannopoulos, Francesco Regazzoni, and Niels Samwel. Towards lightweight cryptographic primitives with built-in fault-detection. *IACR Cryptology ePrint Archive*, 2018:729, 2018.

[SHW+14a] Siwei Sun, Lei Hu, Meiqin Wang, Peng Wang, Kexin Qiao, Xiaoshuang Ma, Danping Shi, Ling Song, and Kai Fu. 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*, 747:2014, 2014.

[SHW+14b] 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 *International Conference on the Theory and Application of Cryptology and Information Security*, 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 *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part I*, pages 95–115, 2015.

[SWW18] Ling Sun, Wei Wang, and Meiqin Wang. More accurate differential properties of led64 and midori64. *IACR Transactions on Symmetric Cryptology*, pages 93–123, 2018.

A Supplementary information for optimized CryptoSMT

Let $a = (a_3, \ldots, a_0)$, and $b = (b_3, \ldots, b_0)$ denote the input and output differences respectively, where a_0, and b_0 are the least significant bits, and $p = (p_2, p_1, p_0)$ is also used to encode the probability of transition $a \to b$. We use the following CNF to model the differential behavior of `CRAFT`'s S-box in the SMT model:

$$(\neg a_1 \vee a_0 \vee b_2 \vee \neg b_1 \vee \neg p_2) \wedge (a_2 \vee \neg a_1 \vee \neg b_1 \vee b_0 \vee \neg p_2) \wedge (a_3 \vee a_2 \vee a_1 \vee \neg b_3 \vee \neg b_0) \wedge (\neg a_3 \vee \neg a_0 \vee b_3 \vee b_2 \vee b_1) \wedge (a_3 \vee a_1 \vee a_0 \vee \neg b_3 \vee \neg b_2) \wedge (\neg a_3 \vee \neg a_2 \vee b_3 \vee b_1 \vee b_0) \wedge (a_2 \vee \neg a_1 \vee b_2 \vee \neg b_1 \vee \neg p_2) \wedge (\neg a_1 \vee a_0 \vee \neg b_1 \vee b_0 \vee \neg p_2) \wedge (\neg a_2 \vee \neg a_0 \vee \neg b_2 \vee \neg b_0 \vee \neg p_2) \wedge (a_1 \vee \neg b_3 \vee \neg b_2 \vee \neg b_0) \wedge (\neg a_3 \vee \neg a_2 \vee \neg a_0 \vee b_1) \wedge (p_1 \vee \neg p_0) \wedge (\neg p_2 \vee p_0) \wedge (\neg b_1 \vee p_0) \wedge (a_2 \vee \neg a_0 \vee b_1 \vee p_2) \wedge (a_1 \vee \neg b_2 \vee b_0 \vee p_2) \wedge (\neg b_3 \vee p_0) \wedge (b_2 \vee \neg b_1 \vee b_0 \vee \neg p_2) \wedge (a_1 \vee b_1 \vee \neg b_0 \vee p_2) \wedge (\neg a_2 \vee a_0 \vee b_1 \vee p_2) \wedge (\neg a_3 \vee b_1 \vee b_0 \vee p_2) \wedge (\neg a_2 \vee \neg a_0 \vee b_3 \vee b_0 \vee p_2) \wedge (\neg a_1 \vee p_0) \wedge (a_2 \vee a_0 \vee b_2 \vee b_1 \vee b_0 \vee \neg p_1) \wedge (a_2 \vee a_1 \vee a_0 \vee p_2 \vee \neg p_0) \wedge (\neg a_2 \vee a_0 \vee \neg b_2 \vee \neg b_0 \vee p_2) \wedge (a_1 \vee b_2 \vee \neg b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee \neg b_2 \vee \neg b_0 \vee \neg p_2) \wedge (\neg a_2 \vee \neg a_0 \vee b_2 \vee \neg b_0 \vee p_2) \wedge (\neg a_3 \vee a_0 \vee b_2 \vee \neg b_0 \vee p_2) \wedge (a_3 \vee \neg a_1 \vee \neg b_3 \vee \neg b_1 \vee p_2) \wedge (a_2 \vee \neg a_0 \vee \neg b_2 \vee \neg b_0 \vee p_2) \wedge (\neg a_1 \vee b_3 \vee \neg b_2 \vee \neg b_1 \vee \neg b_0 \vee p_2) \wedge (a_2 \vee \neg a_1 \vee b_3 \vee b_2 \vee \neg b_1) \wedge (\neg a_1 \vee a_0 \vee b_3 \vee \neg b_2 \vee b_1 \vee \neg p_2) \wedge (a_2 \vee \neg a_1 \vee b_3 \vee b_1 \vee \neg b_0 \vee \neg p_2) \wedge (a_1 \vee \neg b_2 \vee \neg b_1 \vee \neg b_0 \vee \neg p_2) \wedge (b_3 \vee b_2 \vee b_0 \vee \neg p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee \neg p_2) \wedge (\neg a_2 \vee \neg a_1 \vee \neg a_0 \vee b_1 \vee \neg p_2) \wedge (\neg a_3 \vee a_2 \vee \neg a_0 \vee \neg b_2 \vee \neg b_1 \vee b_0) \wedge (a_2 \vee a_0 \vee \neg b_2 \vee \neg b_1 \vee b_0 \vee p_2) \wedge (\neg a_2 \vee \neg a_1 \vee a_0 \vee b_3 \vee b_0) \wedge (\neg a_2 \vee a_0 \vee \neg b_3 \vee b_2 \vee p_2) \wedge (a_2 \vee \neg a_0 \vee \neg b_3 \vee b_0 \vee p_2) \wedge (\neg a_1 \vee a_0 \vee \neg b_3 \vee b_2 \vee \neg b_0 \vee \neg p_2) \wedge (a_2 \vee \neg a_1 \vee \neg b_3 \vee \neg b_2 \vee b_0 \vee \neg p_2) \wedge (\neg a_0 \vee \neg b_3 \vee \neg b_2 \vee b_0 \vee p_2) \wedge (a_3 \vee \neg a_2 \vee a_1 \vee \neg b_1 \vee b_0 \vee \neg p_2) \wedge (a_3 \vee a_1 \vee \neg a_0 \vee b_2 \vee \neg b_1 \vee \neg p_2) \wedge (\neg a_2 \vee \neg a_0 \vee b_2 \vee b_1 \vee b_0) \wedge (a_3 \vee a_2 \vee \neg b_2 \vee \neg b_0 \vee p_2) \wedge (\neg a_3 \vee \neg a_2 \vee a_1 \vee a_0 \vee b_3 \vee b_2 \vee \neg p_2) \wedge (a_3 \vee a_1 \vee a_0 \vee \neg b_3 \vee b_1 \vee \neg b_0) \wedge (a_3 \vee a_2 \vee a_1 \vee \neg b_3 \vee \neg b_2 \vee b_1) \wedge (\neg a_3 \vee a_1 \vee \neg a_0 \vee b_3 \vee b_1 \vee b_0) \wedge (\neg a_3 \vee \neg a_2 \vee a_0 \vee b_2 \vee \neg b_1 \vee \neg b_0)$$

Similar to the Boolean function which is used to model the differential behaviour of an Sbox, another Boolean function can be constructed to model the linear behaviour of the given Sbox, according to the related table of squared correlations. If $a = (a_3, \ldots, a_0)$, and $b = (b_3, \ldots, b_0)$ denote the input and output linear masks respectively, where a_0, and b_0 are the least significant bits, and $p = (p_3, p_2, p_1, p_0)$ is used to encode the squared correlation of transition $a \to b$, we use the following CNF to model the linear behavior of `CRAFT`'s S-box in the SMT model:

$$
\begin{aligned}
&(p_3 \vee p_2) \wedge (p_1 \vee p_0) \wedge (p_3 \vee p_2) \wedge (p_2 \vee p_0) \wedge (b_1 \vee p_0) \wedge (b_3 \vee b_2 \vee b_1 \vee b_0 \vee p_1) \wedge (b_3 \vee p_0) \wedge \\
&(a_3 \vee a_2 \vee a_1 \vee a_0 \vee p_0) \wedge (a_3 \vee a_1 \vee b_3 \vee b_0 \vee p_2) \wedge (a_1 \vee a_0 \vee b_2 \vee b_1 \vee p_2) \wedge (a_1 \vee p_0) \wedge (a_3 \vee \\
&b_3 \vee b_2 \vee b_1 \vee b_0) \wedge (a_3 \vee a_1 \vee b_2 \vee b_1 \vee p_2) \wedge (a_0 \vee b_2 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee a_0 \vee b_3 \vee b_0 \vee p_2) \wedge \\
&(a_0 \vee b_3 \vee b_2 \vee b_1 \vee p_2) \wedge (a_0 \vee b_2 \vee b_1 \vee b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_3 \vee b_1 \vee p_2) \wedge (a_2 \vee a_1 \vee b_1 \vee b_0 \vee \\
&p_2) \wedge (a_3 \vee p_0) \wedge (a_3 \vee a_1 \vee b_2 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee a_1 \vee a_0 \vee b_2 \vee b_1 \vee p_2) \wedge (a_3 \vee a_1 \vee \\
&b_3 \vee b_0 \vee p_2) \wedge (a_2 \vee a_1 \vee b_3 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_2 \vee b_1 \vee p_2) \wedge \\
&(a_3 \vee a_2 \vee a_1 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee b_2 \vee b_0 \vee p_2) \wedge (a_3 \vee a_1 \vee a_0 \vee b_2 \vee b_1 \vee p_2) \wedge \\
&(a_3 \vee a_1 \vee b_3 \vee b_2 \vee p_2) \wedge (a_3 \vee b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_2 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee \\
&b_3 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee b_3 \vee p_2) \wedge (a_3 \vee a_1 \vee b_3 \vee b_1 \vee p_2) \wedge (a_3 \vee b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (a_2 \vee \\
&b_2 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_2 \vee b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_2 \vee b_0 \vee p_2) \wedge \\
&(a_2 \vee a_0 \vee b_2 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (b_3 \vee b_2 \vee b_0 \vee p_2) \wedge \\
&(a_3 \vee a_2 \vee a_1 \vee a_0 \vee b_3 \vee b_1 \vee p_2) \wedge (a_2 \vee a_1 \vee a_0 \vee b_3 \vee b_1 \vee p_2) \wedge (a_2 \vee a_1 \vee a_0 \vee b_3 \vee b_2 \vee \\
&p_2) \wedge (b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee a_0 \vee b_1) \wedge (a_2 \vee a_1 \vee b_3 \vee b_1 \vee b_0 \vee p_2) \wedge (a_3 \vee a_2 \vee \\
&a_0 \vee p_2) \wedge (a_1 \vee a_0 \vee b_3 \vee b_1 \vee p_2) \wedge (a_3 \vee a_2 \vee a_1 \vee a_0 \vee b_2 \vee b_0) \wedge (a_3 \vee a_2 \vee a_0 \vee b_3 \vee b_2 \vee \\
&b_0) \wedge (a_3 \vee a_2 \vee a_0 \vee b_1 \vee b_0 \vee p_2) \wedge (a_2 \vee a_0 \vee b_3 \vee b_2 \vee b_0 \vee p_2) \wedge (a_1 \vee a_0 \vee b_3 \vee b_2 \vee b_0 \vee p_2)
\end{aligned}
$$

CryptoSMT, uses STP[HMS], as the default SMT solver to solve the obtained SMT problem, but it also supports another SMT solver, called Boolector[NPB15]. Table 8 shows that, our optimization improves the speed of solving the obtained SMT problem, for both SMT solvers used in CryptoSMT. Table 9, also shows the impact of our optimization on the solvers' run-time for finding an optimum differential trail for r rounds of `CRAFT`, where the input, and output differences corresponding to the optimum trail are fixed.

Table 8: The impact of the `CRAFT`'s S-box optimization on the solvers' run-time, where SW and MW respectively denote start-weight and minimum found weight through the search for the best single differential characteristic of r rounds of `CRAFT` in the single tweak model

r	SW	MW	Optimized		Unoptimized	
			STP	Boolector	STP	Boolector
1	0	$2(2^{-2})$	0.36 s	0.52 s	34.2 s	13.53 s
2	2	$4(2^{-4})$	0.80 s	1.07 s	75.07 s	29.69 s
3	4	$8(2^{-8})$	2.25 s	2.79 s	195.67 s	74.78 s
4	8	$12(2^{-12})$	3.25 s	4.18 s	313.44 s	100.83 s
5	12	$20(2^{-20})$	10.84 s	13.92 s	831.86 s	234.95 s
6	20	$28(2^{-28})$	20.55 s	20.14 s	1249.29 s	296.06 s
7	28	$40(2^{-40})$	69.03 s	58.7 s	2703.25 s	512.25 s
8	40	$52(2^{-52})$	179.63 s	100.07 s	5526.63 s	664.21 s
9	52	$64(2^{-64})$	501.68 s	190.35 s	9739.49 s	831.22 s

Table 9: The impact of the CRAFT's S-box optimization on the solvers' run-time, to find an optimum differential trail with fixed input, and output differences for r rounds of CRAFT

r	Optimized		Unoptimized	
	STP	Boolector	STP	Boolector
8	0.40 s	0.11 s	34.2 s	15.53 s
9	1.44 s	0.66 s	116.56 s	42.91 s
10	1.61 s	1.44 s	130.38 s	49.3 s
11	2.11 s	1.84 s	147.54 s	55.32 s
12	2.53 s	2.07 s	204.66 s	60.55 s
13	2.76 s	2.53 s	266.03 s	68.30 s
14	3.99 s	3.75 s	319.30 s	74.72 s
15	4.13 s	3.78 s	313.45 s	81.16 s

B Valid linear trails

Table 10: A valid linear trail with weight 124, corresponding to the linear hull $\texttt{0000 00}\gamma\texttt{0 0000 00}\gamma\texttt{0} \xrightarrow{\text{13-round}} \texttt{0000 }\delta\texttt{000 0000 0000}$

i	ΓX^i	ΓY^i	ΓZ^i	w
0	0000 0050 0000 0050	0000 0050 0000 0000	0000 0000 5000 0000	-2
1	0000 0000 A000 0000	0000 0000 A000 0000	0000 00A0 0000 0000	-2
2	0000 00A0 0000 0000	0000 00A0 0000 00A0	000A 0000 A000 0000	-4
3	0005 0000 A000 0000	0005 0000 A005 0005	5000 00A5 0000 0050	-8
4	A000 00AA 0000 00A0	A000 00AA A000 A00A	AA00 00A0 A00A 000A	-12
5	A500 00A0 500A 0005	A500 00A0 F50A A5A5	5A5A 05FA A000 500A	-20
6	A5A5 0A5A A000 F005	A5A5 0A5A 05A5 5FFA	A5FF A505 5A0A 5A5A	-28
7	AA55 5A0F AA05 F5AA	AA55 5A0F 0050 05F0	005F 5000 0A5F A55A	-20
8	00A5 A000 0AF5 AAA5	00A5 A000 0A50 0A00	00A0 5A00 00A0 0A50	-12
9	00A0 A500 00A0 05A0	00A0 A500 0000 A000	0A00 0000 05A0 0A00	-8
10	0A00 0000 0AA0 0A00	0A00 0000 00A0 0000	0000 A000 0000 A000	-4
11	0000 A000 0000 A000	0000 A000 0000 0000	0000 0000 00A0 0000	-2
12	0000 0000 00A0 0000	0000 0000 00A0 0000	0000 A000 0000 0000	-2
13	0000 5000 0000 0000	none	none	none

Table 11: A valid linear trail with weight 124, corresponding to the linear hull $\texttt{0000}\ \ \gamma\texttt{000}\ \ \texttt{0000}\ \ \gamma\texttt{000} \xrightarrow{\text{13-round}} \texttt{0000}\ \ \texttt{00}\delta\texttt{0}\ \ \texttt{0000}\ \ \texttt{0000}$

i	ΓX^i	ΓY^i	ΓZ^i	w
0	0000 5000 0000 5000	0000 5000 0000 0000	0000 0000 0050 0000	-2
1	0000 0000 00F0 0000	0000 0000 00F0 0000	0000 F000 0000 0000	-2
2	0000 F000 0000 0000	0000 F000 0000 F000	0F00 0000 00F0 0000	-4
3	0500 0000 0050 0000	0500 0000 0550 0500	0050 5500 0000 5000	-8
4	00A0 AF00 0000 A000	00A0 AF00 00A0 0FA0	00FA A000 0FA0 0A00	-12
5	005A 5000 05A0 0500	005A 5000 05FA 555A	A555 F50A 0050 05A0	-20
6	AFBF FA0A 00B0 0A50	AFBF FA0A AF0F 5FE5	55FE 0FAF 0AFA FBFA	-28
7	AAF5 0FA5 0AF5 555A	AAF5 0FA5 A000 F00A	AF00 00A0 AF05 AF5A	-20
8	A500 00A0 550F A5AA	A500 00A0 F00F 000A	A000 00FF A000 500A	-12
9	A000 0055 A000 A005	A000 0055 0000 0050	0005 0000 5005 000A	-8
10	000A 0000 A00A 000A	000A 0000 A000 0000	0000 00A0 0000 00A0	-4
11	0000 0050 0000 0050	0000 0050 0000 0000	0000 0000 5000 0000	-2
12	0000 0000 A000 0000	0000 0000 A000 0000	0000 00A0 0000 0000	-2
13	0000 0050 0000 0000			

C Possible differential trails

Table 12: A possible differential trail with weight 124, corresponding to the input/output pattern $\texttt{00}\gamma\texttt{0}\ \ \texttt{0000}\ \ \texttt{00}\gamma\texttt{0}\ \ \texttt{0000} \xrightarrow{\text{13-round}} \texttt{0000}\ \ \texttt{0000}\ \ \delta\texttt{000}\ \ \texttt{0000}$

i	ΔX^i	ΔY^i	ΔZ^i	w
0	00A0 0000 00A0 0000	0000 0000 00A0 0000	0000 A000 0000 0000	-2
1	0000 F000 0000 0000	0000 F000 0000 0000	0000 0000 00F0 0000	-2
2	0000 0000 00F0 0000	00F0 0000 00F0 0000	0000 F000 0000 0F00	-4
3	0000 F000 0000 0F00	0F00 FF00 0000 0F00	00F0 0000 0FF0 F000	-8
4	00F0 0000 0AF0 A000	AA00 A000 0AF0 A000	0A00 FA00 00A0 A00A	-12
5	0F00 FD00 00A0 500F	5FAF AD0F 00A0 500F	F500 A000 0DAF FAF5	-20
6	AA00 F000 0A5F FDFA	5DA5 0DFA 0A5F FDFA	AFDF 5A0F FD0A DA55	-28
7	5FAA A50A FA0A A57A	00DA 0070 FA0A A57A	AA57 0AFA 7000 0DA0	-20
8	DAAD 0AFA D000 0AA0	000D 005A D000 0AA0	00AA 00D0 500A 00D0	-12
9	00AA 00A0 A00A 00A0	A000 0000 A00A 00A0	000A 00AA 0000 000A	-8
10	0005 00A5 0000 0005	0000 00A0 0000 0005	5000 0000 A000 0000	-4
11	A000 0000 A000 0000	0000 0000 A000 0000	0000 00A0 0000 0000	-2
12	0000 0050 0000 0000	0000 0050 0000 0000	0000 0000 5000 0000	-2
13	0000 0000 A000 0000			

Table 13: A possible differential trail with weight 124, corresponding to the input/output pattern γ000 0000 γ000 0000 $\xrightarrow{\text{13-round}}$ 0000 0000 00δ0 0000

i	ΔX^i	ΔY^i	ΔZ^i	w
0	A000 0000 A000 0000	0000 0000 A000 0000	0000 00A0 0000 0000	-2
1	0000 00A0 0000 0000	0000 00A0 0000 0000	0000 0000 A000 0000	-2
2	0000 0000 D000 0000	D000 0000 D000 0000	0000 00D0 0000 000D	-4
3	0000 0070 0000 000A	000A 007A 0000 000A	A000 0000 700A 00A0	-8
4	D000 0000 D00A 00A0	00AA 00A0 D00A 00A0	000A 00DA A000 0AA0	-12
5	000A 007A A000 05A0	A5AA 05DA A000 05A0	005A 00A0 D50A 5AAA	-20
6	007D 00D0 770D ADDA	DAAA AD0A 770D ADDA	AADD 077D 0DAA AAAD	-28
7	AA77 05DA 0AAD F5DA	5500 F000 0AAD F5DA	AF5D AA0D 00F0 5005	-20
8	AAAA DA0A 00A0 A00A	0A00 7A00 00A0 A00A	AA00 A000 0A70 A000	-12
9	5D00 5000 0DD0 5000	00D0 0000 0DD0 5000	0500 DD00 0000 0D00	-8
10	0700 A700 0000 0700	0000 A000 0000 0700	0070 0000 00A0 0000	-4
11	0050 0000 0050 0000	0000 0000 0050 0000	0000 5000 0000 0000	-2
12	0000 A000 0000 0000	0000 A000 0000 0000	0000 0000 00A0 0000	-2
13	0000 0000 00A0 0000			