Links Between Truncated Differential and Multidimensional Linear Properties of Block Ciphers and Underlying Attack Complexities

Céline Blondeau and Kaisa Nyberg

Department of Information and Computer Science,
Aalto University School of Science, Finland
{celine.blondeau, kaisa.nyberg}@aalto.fi

Abstract. The mere number of various apparently different statistical attacks on block ciphers has raised the question about their relationships which would allow to classify them and determine those that give essentially complementary information about the security of block ciphers. While mathematical links between some statistical attacks have been derived in the last couple of years, the important link between general truncated differential and multidimensional linear attacks has been missing. In this work we close this gap. The new link is then exploited to relate the complexities of chosen-plaintext and known-plaintext distinguishing attacks of differential and linear types, and further, to explore the relations between the key-recovery attacks. Our analysis shows that a statistical saturation attack is the same as a truncated differential attack, which allows us, for the first time, to provide a justifiable analysis of the complexity of the statistical saturation attack and discuss its validity on 24 rounds of the PRESENT block cipher. By studying the data, time and memory complexities of a multidimensional linear key-recovery attack and its relation with a truncated differential one, we also show that in most cases a known-plaintext attack can be transformed into a less costly chosen-plaintext attack. In particular, we show that there is a differential attack in the chosen-plaintext model on 26 rounds of PRESENT with less memory complexity than the best previous attack, which assumes known plaintext. The links between the statistical attacks discussed in this paper give further examples of attacks where the method used to sample the data required by the statistical test is more differentiating than the method used for finding the distinguishing property.

Keywords: statistical cryptanalysis, block cipher, chosen plaintext, known plaintext, differential cryptanalysis, truncated differential cryptanalysis, linear cryptanalysis, multidimensional linear cryptanalysis, statistical saturation, integral, zero-correlation, impossible differential.

1 Introduction

After the invention of the differential and linear cryptanalysis several extensions and related statistical cryptanalysis methods for block ciphers have been presented. The need for a common framework for statistical attacks that would facilitate their comparison has been raised in the literature at least by Vaudenay [1] and Wagner [2], who also put forward such frameworks. While the former aims at providing provable security against all statistical attacks, the latter takes a high level view on the iterated Markov ciphers. In this paper, we propose a more pragmatic approach to show that, no matter whether we use a linear or differential characteristic to identify some non-random behavior, it can be exploited for a known-plaintext (KP) and a chosen-plaintext attack (CP).

Previously, many mathematical relationships between statistical cryptanalysis methods have been established. They concern the computation of the main statistic of the cryptanalysis method under consideration. In [3], Leander studied relations between the statistical saturation (SS) attack [4] and the multidimensional linear (ML) cryptanalysis using the χ^2 statistical test [5]. In the former the strength of the distinguishing property is measured by the non-uniformity of the distribution of partial ciphertext values when part of the plaintext is fixed. In the latter, the non-uniformity of the joint distribution of plaintext parts and ciphertext parts is under consideration. Leander showed that the non-uniformities computed in the SS attack are on average equal to the non-uniformity of the distribution considered in the ML attack.

Later more links were established in [6] and also applied in practice. For example, an efficient zero-correlation (ZC) property was found on a variant of Skipjack. Using the mathematical link it was then transformed to an integral property to launch an efficient CP attack on 31 rounds of this cipher. The question arises, whether it would have been possible to use the ZC property directly. Or would it have consumed essentially more data, time, or memory to exploit the ZC property directly in this attack? The purpose of this paper is to give an exhaustive answer to such questions in the more general setting of truncated differential (TD) [7] and ML attacks.

Building on the link proposed by Chabaud and Vaudenay [8] and applied by Blondeau and Nyberg [9], we establish now a more general mathematical link between differential and linear statistical properties of block ciphers. This link provides a unified view on statistical distinguishers of block ciphers that measure the uniformity of a distribution of pairs of partial plaintext and ciphertext values and covers any TD and ML distinguishers. It allows for examination and comparison of the corresponding KP and CP distinguishing attacks and the related statistical models. In this paper, we will make a detailed comparison between the data, time and memory complexities of the CP TD and KP ML distinguishing attacks. Also the SS distinguisher will be considered and shown to be essentially identical to a TD distinguisher.

One of the main results given in this paper is that for any KP ML distinguishing attack there is a stronger CP TD distinguishing attack, where the strength is measured in terms of the data, time and memory complexities of the distinguishers. We will see that the main advantage of the CP TD distinguishers over the KP ML distinguishers is due to the better organisation of the chosen data and allows some data, time and memory savings. Overall, the results obtained in this paper show that the method used for finding the distinguishing property, differential or linear, may be quite irrelevant when performing the attack. We show cases where, for the same distinguisher, the data, time and memory complexities of the attack are essentially different depending on what kind of data sampling methods are used for the statistical test.

The knowledge of the relationships between the different distinguishers and their complexities will then be applied to the outstanding key-recovery attacks. In particular, we will compare the KP and CP scenarios and their effects on the complexities of the key-recovery attacks based on different but mathematically equivalent distinguishing properties of the cipher. The cost difference of using KP instead of CP can be quite small. When such a case occurs in practice, the cryptanalyst can choose whether to use KP data with small additional cost instead of CP data to perform the key-recovery attack.

The resistance of PRESENT [6] against TD attacks has been a longstanding open problem. Since the strong differentials of the Sbox diffuse faster than the strong linear approximations as the number of rounds increases, it has been very difficult to achieve accurate estimates of differential probabilities directly. In [9], linear approximations were used to evaluate some differential probabilities. While the obtained estimates were accurate, no differentials were found that would essentially improve the best known differential attack, which can break 19 rounds of PRESENT [9]. Using the results obtained in this paper, we convert the 24-round ML distinguisher of [10] to a TD distinguisher and use it to present a TD key-recovery attack on a 26-round reduced version of PRESENT. This attack which reach the same number of rounds than the KP ML attack of [10] illustrates than one can make used of linear properties to conduct a differential attack.

The rest of the paper is organized as follows. In Sect. 2, we present a general link between the TD and ML properties. In Sect. 3, we study the data, time and memory complexities of the CP TD, CP SS and KP ML distinguishing attacks, which depending on the parameters of the underlying properties suggest different time-memory tradeoffs. By showing that the SS attacks correspond to TD attacks, in Sect. 4, we provide improved complexity estimates of the SS attacks. Sect. 5 is dedicated to the link between the TD and ML key-recovery attack. We show how to convert a CP attack to a KP attack and analyze the cost of this conversion. On the other hand, we show the existence of a TD attack on 26 rounds of PRESENT, which requires less memory than the best known ML attack on PRESENT. In Sect. 6, we analyze other known statistical attacks on block cipher and discuss their relations. Sect. 7 summarizes the results on these different links.

2 Preliminaries

2.1 ML and TD Setting and Notation

In differential cryptanalysis [11], the attacker is interested in finding and exploiting non-uniformity in occurrences of plaintext and ciphertext differences. Given a vectorial Boolean function $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$, a differential is a pair $(\delta, \varDelta)$ where $\delta \in \mathbb{F}_2^n$ and $\varDelta \in \mathbb{F}_2^n$ and its probability is defined as

$$\mathbf{P}[\delta \xrightarrow{F} \varDelta] = 2^{-n} \#\{x \in \mathbb{F}_2^n \,|\, F(x) \oplus F(x \oplus \delta) = \varDelta\}.$$

Linear cryptanalysis [12] uses a linear relation between bits from plaintexts, corresponding ciphertexts and encryption key. The strength of the linear relation is measured by its correlation. The correlation of a Boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2$ is defined as

$$\mathbf{cor}(f) = \mathbf{cor}(f(x)) = 2^{-n} \Big[\# \left\{ x \in \mathbb{F}_2^n | f(x) = 0 \right\} - \# \left\{ x \in \mathbb{F}_2^n | f(x) = 1 \right\} \Big],$$

where the quantity within brackets can be computed as the Walsh transform of f evaluated at zero, see e.g. [13].

In block ciphers, the data is usually represented as vectors in some basis over $\mathbb{F}_2$. For the purposes of our analysis, we also present the input and output data as vectors over $\mathbb{F}_2$. The selection of the basis we use is determined by the linear or differential properties of the cipher. Hence the basis we use may or may not be the same as used for the description of the cipher. The input and output spaces are divided into two orthogonal spaces as follows

$$F : \mathbb{F}_2^s \times \mathbb{F}_2^t \to \mathbb{F}_2^q \times \mathbb{F}_2^r : \; (x_s, x_t) \mapsto (y_q, y_r) = F(x_s, x_t), \text{where } s + t = q + r = n.$$

In this study, we focus on ML approximations composed of 2^s input masks $(a_s, 0) \in \mathbb{F}_2^s \times \{0\}$, and 2^q output masks $(b_q, 0) \in \mathbb{F}_2^q \times \{0\}$, which makes in total 2^{s+q} linear approximations over F. The correlation of a linear approximation determined by a mask pair $(a_s, 0), (b_q, 0)$ is then $\mathbf{cor}\,(a_s \cdot x_s + b_q \cdot y_q)$, where $x = (x_s, x_t) \in \mathbb{F}_2^s \times \mathbb{F}_2^t$ and $F(x_s, x_t) = (y_q, y_r) \in \mathbb{F}_2^q \times \mathbb{F}_2^r$.

The strength of the ML approximation $[(a_s, 0), (b_q, 0)]_{a_s \in \mathbb{F}_2^s,\, b_q \in \mathbb{F}_2^q}$ is measured by its capacity C defined as follows

$$C = \sum_{(a_s, b_q) \neq (0,0)} \mathbf{cor}^2 \left(a_s \cdot x_s \oplus b_q \cdot y_q \right). \tag{1}$$

The capacity can also be computed as an L^2-distance between the probability distribution of the pairs (x_s, y_q) of partial plaintext and ciphertext values and the uniform distribution over $\mathbb{F}_2^s \times \mathbb{F}_2^q$. As we will show in this paper, this ML approximation is related to a certain TD. This TD is composed of 2^t input differences $(0, \delta_t) \in \{0\} \times \mathbb{F}_2^t$, and 2^r output differences $(0, \varDelta_r) \in \{0\} \times \mathbb{F}_2^r$, which makes in total 2^{t+r} differentials over the cipher F. The probability of a differential determined by the input and output differences $(0, \delta_t)$ and $(0, \varDelta_r)$ is then $\mathbf{P}[(0, \delta_t) \xrightarrow{F} (0, \varDelta_r)] = 2^{-n} \#\{x \in \mathbb{F}_2^n \,|\, F(x) \oplus F\left(x \oplus (0, \delta_t)\right) = (0, \varDelta_r)\}$.

Then the probability p of the TD $[(0, \delta_t), (0, \varDelta_r)]_{\delta_t \in \mathbb{F}_2^t, \varDelta_r \in \mathbb{F}_2^r}$ is defined as the average probability that the output difference is in the set $\{(0, \varDelta_r) \,|\, \varDelta_r \in \mathbb{F}_2^r\}$

taken over the input differences $(0, \delta_t)$, $\delta_t \in \mathbb{F}_2^t$, which are assumed to be equally likely. Hence

$$p = 2^{-t} \sum_{\delta_t \in \mathbb{F}_2^t, \varDelta_r \in \mathbb{F}_2^r} P[(0, \delta_t) \xrightarrow{F} (0, \varDelta_r)]. \tag{2}$$

Note that this definition of TD probability includes the zero input difference.

2.2 Mathematical Link

Chabaud and Vaudenay [8] provide a link between the differential probabilities and the squared correlations of linear approximations of vectorial Boolean functions. In the context of this paper, this one can be written as

$$\mathbf{P}[\delta \xrightarrow{F} \varDelta] = 2^{-n} \sum_{a \in \mathbb{F}_2^n} \sum_{b \in \mathbb{F}_2^n} (-1)^{a \cdot \delta \oplus b \cdot \varDelta} \mathbf{cor}^2 \left(a \cdot x \oplus b \cdot F(x)\right),$$

where $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a vectorial Boolean function. By applying this link to the splitted spaces defined above and summing up over all $\delta_t \in \mathbb{F}_2^t$ and $\varDelta_r \in \mathbb{F}_2^r$, the following expression for the probability of a TD is given in [9].

Theorem 1 ([9]). *For all $\delta_s \in \mathbb{F}_2^s$ and $\varDelta_q \in \mathbb{F}_2^q$ it holds that*

$$2^{-t} \sum_{\delta_t \in \mathbb{F}_2^t, \varDelta_r \in \mathbb{F}_2^r} \mathbf{P}[(\delta_s, \delta_t) \xrightarrow{F} (\varDelta_q, \varDelta_r)] =$$
$$2^{-q} \sum_{a_s \in \mathbb{F}_2^s, b_q \in \mathbb{F}_2^q} (-1)^{a_s \cdot \delta_s \oplus b_q \cdot \varDelta_q} \mathbf{cor}^2 \left((a_s, 0) \cdot x \oplus (b_q, 0) \cdot F(x)\right).$$

In [9] this result was used in the case when $q = t$ and all the nontrivial correlations and differential probabilities are equal to zero, to provide a link between zero-correlation linear (ZC) cryptanalysis [14, 6] and impossible differential (ID) cryptanalysis [15]. In this paper, we focus on the case where $\delta_s = 0$ and $\varDelta_q = 0$, but no other assumptions are made about the correlations and differential probabilities. If $\delta_s = 0$ and $\varDelta_q = 0$, then $a_s \cdot \delta_s \oplus b_q \cdot \varDelta_q = 0$, $a_s \in \mathbb{F}_2^s$ and $b_q \in \mathbb{F}_2^q$. By using the notations of (1) and (2) we get the following corollary of Th. 1.

Corollary 1. *Let the TD probability p be defined as in (2) and the ML capacity C as in (1). Then*

$$p = 2^{-q}(C + 1). \tag{3}$$

In the ML context, we evaluate the non-uniformity of the distribution of partial plaintext and ciphertext pairs (x_s, y_q) in terms of the L^2-distance. By Cor. 1 this non-uniformity can be measured in terms of probability of coincidences in the observed values (x_s, y_q). As a special case of Cor. 1, we get the method of Index of Coincidence [16] over some binary alphabet by taking $s = 0$ and $q = n$. Notice that the link given in (3) holds for a block cipher with a fixed-key as well as on average over the keys.

Next we examine the different statistical models developed for ML and TD types of distinguishers and derive relationships between their data, time and memory complexities.

2.3 Complexity of an Attack

While the most powerful statistical attacks aim at recovering some information on the secret key, they are often derived from a distinguishing attack consisting of identifying if a cipher is drawn at random or not. Given some statistical distribution, the data complexity of an attack corresponds to the number of plaintexts necessary to successfully perform this distinguishing operation.

When the distinguishing attack is turned to a key-recovery attack, it is common to separate the process into a *distillation phase* consisting of the extraction of some statistics for all subkey candidates from the available data, an *analysis phase* consisting of the computation of the likelihood of each of the key candidates and a *search phase* for the exhaustive search of the corresponding master key from the list of kept candidates. In the following, we denote by K the set of key candidates.

Throughout this paper, to facilitate the comparison of attacks, we assume that the success probability of finding the key is fixed to 50%. For a key-recovery attack the probability of false positives determines the time complexity of the search phase. The notion of advantage a defined in [17] corresponds to a probability of false positives of 2^{-a}. For simplicity, in the statistical derivations of this paper, we denote by φ_a the quantity $\Phi^{-1}(1-2^{-a})$ where Φ is the cumulative function of the standard normal distribution $\mathcal{N}(0,1)$.

3 Complexity of a Distinguishing Attack

Having established the link (3) between the ML and TD properties of a vector-valued Boolean function, we now examine the distinguishers derived from these properties for block ciphers and their complexities. We use the most commonly accepted statistical models for the distinguishing attacks. The major difference between the distinguishing attacks based on ML and TD is that the former is a KP attack and the latter a CP attack. In this section we analyze this difference in more detail and discuss how it affects the complexities of the distinguishing attacks.

3.1 ML Distinguishing Attacks

For ML attacks both LLR and χ^2 statistical tests have been used in the literature. In this paper, we restrict our analysis to the χ^2 test, which first, according to the results discussed in the following of this section seems to be in good accordance with the common statistical test for a TD distinguishing attack, and secondly, is more applied in practice since it does not require having accurate prediction of the distributions derived from the cipher data.

The data complexity of an ML attack has been studied in [5], and can be computed similarly than for a classical linear attack modelled in [17].

Proposition 1. *For a success probability of 50% and an advantage of a bits, the data complexity N^{ML} of an ML distinguishing attack using 2^{s+q} linear approximations with capacity C as defined in (1) is*

$$N^{ML} = \frac{2^{(s+q+1)/2}}{C}\varphi_a. \tag{4}$$

Given a set of 2^{s+q} linear approximations, the general algorithm presented in Alg. 1, for an ML distinguisher using the χ^2 statistical test requiring N plaintexts can be performed using 2^{q+s} simple operations[1].

Alg. 1 Multidimensional linear distinguisher

```
Set a counter $D$ to 0
Create a table $T$ of size $2^{q+s}$
**for** $N$ plaintexts **do**
    $(y_q, y_r) = E((x_s, x_t))$
    $T[(x_s, y_q)] += 1$
**for** all $(x_s, y_q)$ **do**
    $D += (T[(x_s, y_q)] - N/2^{q+s})^2$
```

ML distinguishing attacks typically require 2^{s+q} counters, either for evaluating the correlations of the 2^{s+q} linear approximations or evaluating the distributions over the 2^{s+q} values. We observe that in the general ML setting it is possible that 2^{s+q} is larger than the data complexity N^{ML}, in which case the memory requirement can be reduced to N^{ML}. Then it is enough for such an algorithm to deal with a sorted list of maximum size N^{ML}. Using a binary search the time complexity of this KP ML distinguisher is $N^{ML}\log(N^{ML})$.

3.2 TD and SS Distinguishing Attacks

The probability of a TD as given in (2) is computed as an average probability over the input differences. By ordering the plaintexts into structures we can efficiently handle an evaluation of the TD probability for multiple input differences.

In the following, let us assume that all structures are of equal size, and let us denote by S the size of the structures and by M the number of structures used in the attack. Then the total amount of data N^{TD} used for the TD attack is equal to $M \cdot S$. For a further comparison with the complexity derived for the ML attack, we express the relation between the data complexity and the advantage of the TD attack using the framework of [17]. In the context where $p = 2^{-q} + 2^{-q}C$ is close to the uniform probability 2^{-q} ($C \ll 1$) this model is in accordance with the more general model presented in [18].

Proposition 2. *For a success probability of 50% and an advantage of a bits, the data complexity of a TD distinguishing attack using 2^t input differences and*

[1] In some cases this complexity can be reduced using a FFT.

2^r *output differences with probability* p *as defined in (2) is*

$$N^{TD} = \frac{2^{-q+1}}{S \cdot (p - 2^{-q})^2} \cdot \varphi_a^2, \textit{ where } S \leq 2^t. \tag{5}$$

Proof. According to the framework of [17], the number of pairs N_S required for such a TD distinguisher is $N_S = \frac{2^{-q}}{(p-2^{-q})^2}\varphi_a^2$. By using M structures of S plaintexts, we can generate $N_S = M \cdot (S-1)S/2$ pairs, which we obtain if the amount of available CP data is $N^{TD} = M \cdot S \approx 2N_S/S$.

Alg. 2 TD and SS distinguishers

```
Set a counter $D$ to 0
for $M$ values of $x_s \in \mathbb{F}_2^s$ do
    Create a table $T$ of size $S$
    for $S$ values of $x_t \in \mathbb{F}_2^t$ do
        $(y_q, y_r) = E((x_s, x_t))$
        $T[x_t] = y_q$
    for all pairs $(x_t, x_t')$ do
        if $T[x_t] = T[x_t']$ then
            $D+ = 1$
```

(2a) Generic TD distinguisher

```
Set a counter $D$ ($D'$) to 0
for $M$ values of $x_s \in \mathbb{F}_2^s$ do
    Initialize to 0 a table $T$ of size $2^q$
    for $S$ values of $x_t \in \mathbb{F}_2^t$ do
        $(y_q, y_r) = E((x_s, x_t))$
        $T[y_q]+ = 1$
    for all $y_q \in \mathbb{F}_2^q$ do
        $D+ = T[y_q] \cdot (T[y_q] - 1)/2$
        $(D'+ = T[y_q]^2)$
```

(2b) Improved TD distinguisher (SS distinguisher)

When in the context of TD cryptanalysis, the number of considered input differences t is relatively small, the cryptanalyst usually runs a distinguisher of the type given in Alg. 2a. Each structure is handled separately. To minimize the number of encryptions, the partial ciphertexts y_q are stored. The time complexity of the CP TD distinguisher represented in Alg. 2a corresponds to N^{TD} encryptions and $M \cdot S^2/2$ simple operations. The time taken by the comparison between all ciphertext pairs is then often considered as the limiting factor for the attack. We observe that by using the memory differently meaning that instead of storing all partial ciphertexts y_q, soring only their distribution, we can also reduce the time complexity. Indeed, if ℓ partial ciphertexts y_q are equal, then $\ell(\ell-1)/2$ ciphertext pairs have difference zero in these q bits. The TD distinguishing algorithm modified in this manner is presented in Alg. 2b. At a memory cost of 2^q counters, its time complexity is $M \cdot 2^q$ simple operations and N^{TD} encryptions.

The SS attack has been proposed by Collard and Standaert [4] and applied on the cipher PRESENT [19]. It exploits the non-uniformity of the distribution of the partial ciphertexts $y_q \in \mathbb{F}_2^q$ obtained by encryption of plaintexts (x_s, x_t) by keeping x_s fixed. The non-uniformity is measured using the L^2-distance. This is exactly what Alg. 2b computes using the score D'. By noticing that the score D of the TD distinguisher satisfies $D = \sum_M \sum_{y_q \mathbb{F}_2^q} T[y_q] \cdot (T[y_q]-1)/2 = D' - M \cdot S/2$, we conclude that the CP TD distinguisher as described in Alg. 2b is identical to the CP SS distinguisher of [4].

In 2011, Leander [3] observed a mathematical relation between the expected values of the SS score D' computed in Alg. 2b and the ML score D in Alg 1 .

But this link has not been used for developing a statistical model for SS attacks. The statistical model developed in this paper, allows for the first time to derive accurate estimates of the data complexities for the last-rounds SS key-recovery attack on PRESENT proposed in [4]. This key recovery attack will be explained and analyzed in Sect. 4.

3.3 Comparison Between ML and TD Distinguishers

Recalling Cor. 1 we can summarize the results from (4) and (5) and get the following relationship between the data complexities N^{ML} and N^{TD} of the ML distinguisher and the TD distinguisher.

Corollary 2. *Consider an ML distinguisher and a TD distinguisher based on the ML and TD properties defined in Sect. 2.1. Then*

$$N^{TD} = \frac{(N^{ML})^2}{S \cdot 2^s} = \frac{2^{q+1}}{S \cdot C^2} \cdot \varphi_a^2.$$

In Table 1, we summarize the complexities of the KP ML and CP TD distinguishers presented in this section. Given the splitted input and output spaces, $\mathbb{F}_2^n = \mathbb{F}_2^s \times \mathbb{F}_2^t = \mathbb{F}_2^q \times \mathbb{F}_2^r$, a TD distinguisher as presented in Alg. 2b is less memory demanding than a ML distinguisher as presented in Alg. 1. According to a commonly adopted practice in differential cryptanalysis, the structure size is maximized to minimize the time complexity. If $S = 2^t$ we obtain by Cor. 2 that

Table 1: Complexities of the ML and TD distinguishing algorithms.

Alg.	Data	Time	Memory	Condition
ML, Alg. 1	N^{ML}	N^{ML}	2^{s+q}	$2^{s+q} < N^{ML}$
TD, Alg. 2a	N^{TD}	$N^{TD} + N^{TD}S$	$S(\leq 2^t)$	$N^{TD}S < 2^n$
TD, Alg. 2b	N^{TD}	$N^{TD} + M \cdot 2^q$	$\min(S, 2^q)$	-

$$N^{TD} = 2^{-n}(N^{ML})^2.$$

This means that also the data and the time complexities of the TD distinguisher is smaller than the ones of the corresponding ML distinguisher.

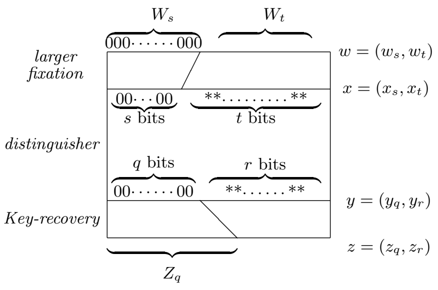
In the remaining sections of this paper, we focus on the TD and ML key-recovery attacks. In particular, we investigate whether a CP attack is always less costly than a KP attack, and extract links with other statistical key-recovery attacks on block ciphers.

4 TD and SS Key-recovery Attacks

4.1 Last-rounds TD and SS Key-recovery Attack

For the results described in this section, we use the notation of Alg. 3. The s bits of the TD distinguisher on which the input difference is fixed to 0 is called a fixation. As suggested by [4], if the size of the fixation is small, we can increase the number of rounds of the distinguisher. Given the fixation on s bits, we denote by W_s the larger fixation after adding some rounds at the beginning of the distinguisher. By choosing structures such that the part w_s of the plaintext $w = (w_s, w_t) \in W_s \times W_t$ is fixed we remain certain that the s-bits part x_s of $x = (x_s, x_t)$ is fixed and ensures zero difference in these bits between two plaintexts in a same structure. The space Z_q, described in Alg. 3 corresponds to the minimal space needed for checking the distribution of the q bits y_q after partial decryption of the ciphertexts $z = (z_q, z_r)$. The resulting CP TD last-rounds key-recovery attack is depicted in Alg. 3.

Alg. 3 Last-rounds CP TD and SS key-recovery attack



```
For all $|K|$ key candidates $k$, set a counter $D_k$ to 0
for $M$ values of $w_s \in W_s$ do
    Initialize a vector $V$ of size $|Z_q|$ to 0
    for $S$ values of $w_t \in W_t$ do
        $(z_q, z_r) = E((w_s, w_t))$
        $V[z_q]+ = 1$
    for all key candidates $k$ do
        Initialize a vector $T$ of size $2^q$ to 0
        for all $z_q \in Z_q$ do
            Partially decrypt to obtain $y_q$ from $z_q$
            $T[y_q]+ = V[z_q]$
        for all $y_q \in \mathbb{F}_2^q$ do
            $D_k+ = T[y_q] \cdot (T[y_q] - 1)/2$
Sort the counters $D_k$ to obtain the most likely keys.
```

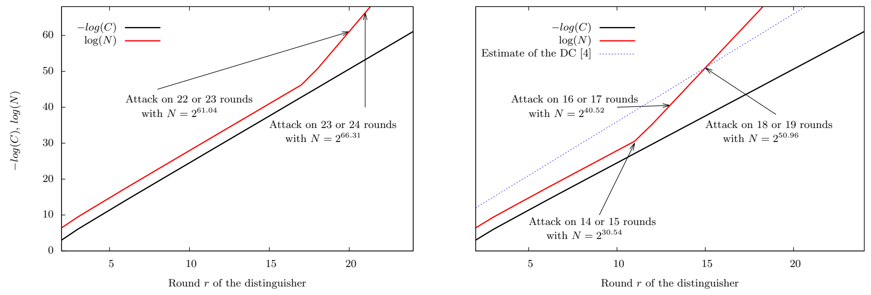
Implementing Alg. 3 requires storing $|K| + |Z_q|$ counters. This algorithm which runs in a time corresponding to $M \cdot S$ encryptions and $M \cdot |K| \cdot |Z_q|$ partial inversions requires $M \cdot S$ chosen plaintexts. Note that by increasing the size of the fixation, the size S of a structure is limited to $S \le |W_t| \le 2^t$. Then according to (5) the increasing data complexity constitutes a major limiting factor to this process which consists of adding rounds at the beginning of the distinguisher without guessing any key-bits on these rounds.

4.2 Using the Link Between TD and SS Attacks to Analyze the SS Attack on 24 Rounds of PRESENT

From the complexity of the last-rounds TD key-recovery attack given in Alg. 3 and the relation between TD and SS described in Sect. 3.2 , we analyze in this section the SS key-recovery attack of [4] on 24 rounds of PRESENT.

When linking the statistic computed in the SS attack with the capacity of a ML approximation, Leander [3] also confirms that the capacity of the ML distinguisher can be estimated as suggested in [4] by multiplying by a factor close to 2^{-3} when adding a round to the distinguisher (see Fig. 1). In ML attacks the data complexity is inversely proportional to the capacity, and the same was assumed to hold for the SS distinguisher used in [4]. While the experiments of [20] confirm this hypothesis when the attack is limited to one structure, a gap was observed in [20] starting from rounds 18 (or 19). Next we present an explanation of this behavior based on the statistical model of the SS distinguisher we derived using the TD model.

From Cor. 2 we know that the number of samples of a TD attack so also of a SS attack is a multiple of $\frac{2^q}{C^2}$. As long as only one structure is used, with N plaintexts we can generate $N^2/2$ plaintext pairs and the data complexity is $N = \frac{2^{(q+1)/2}}{C}\varphi_a$. But when more than one structure is used the data complexity is proportional to the square of the inverse of the capacity: $N = \frac{2^{q+1}}{|W_t| \cdot C^2}\varphi_a^2$. This phenomenon is illustrated in Fig. 1. Given a distinguisher on r rounds, the data complexity computed from Cor. 2 of the SS attack on $r+3$ rounds, meaning with a fixation of $log(|W_s|) = 16$ bits, and on $r+4$ rounds, meaning with a fixation of $log(|W_s|) = 32$ bits, is given in Fig. 1. In particular, the computed values on the right figure are, for the first time, in accordance with the experiments done in [20]. From Fig. 1, one can see that by fixing 32 bits only 21 rounds can be attacked. By fixing 16 bits, one can compute that an attack on 24 rounds will require more than the full codebook. From these observations, we conclude that the SS attack described in [4] only works for 23 rounds of PRESENT instead of 24 rounds as originally claimed.



(a) $|W_s| = 2^{16}$: Attack on $r+3$ rounds (b) $|W_s| = 2^{32}$: Attack on$r+4$ rounds

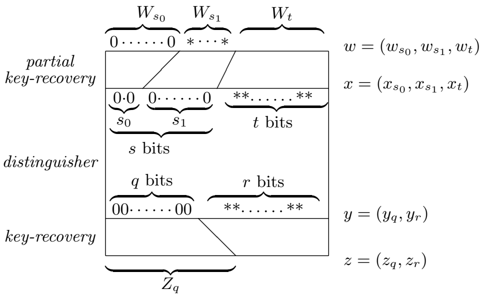
Fig. 1: Capacity C of the r-round ML of [4] as computed in [3] and data complexity N (computed using Cor. 2) of the underlined attacks on $r+r'$-rounds for $a = 8$.

5 Comparison of TD and ML Key-recovery Attacks

5.1 Partial Key-recovery Attack on the First Rounds

In the previous section, we discussed the limitation of adding rounds at the beginning of the distinguisher without guessing any key-bits on these rounds. In this section, we develop a TD key-recovery attack which allow to find the key of the first rounds. For more generality and to illustrate some data, time and memory trade-offs, we assume that the aim is to guess only part of the possible key bits in the first rounds. In Alg. 4, we describe this TD key-recovery attack. From the fixation of s bits, we want to keep a fixation on s_0 bits, and we define a space W_{s_0} such that given a fixation on W_{s_0} after partial encryption we have a fixation on these s_0 bits. We then take advantage of the non-fixed $s_1 + t$ bits to find some information on the first rounds subkey. In Alg. 4, the space $W_{s_1} \times W_t$ corresponds to the non-fixed bits of $w = (w_{s_0}, w_{s_1}, w_t)$.

Alg. 4 First-rounds and last-rounds TD key-recovery attack



```
For all $|K|$ key candidates $k$, set a counter $D_k$ to 0
**for** $M$ values of $w_{s_0} \in W_{s_0}$ **do**
    Initialize a vector $V$ of size $|W_{s_1}| \cdot |Z_q|$ to 0
    **for** $S$ values of $(w_{s_1}, w_t) \in W_{s_1} \cup W_t$ **do**
        $(z_q, z_r) = E((w_{s_0}, w_{s_1}, w_t))$
        $V[(w_{s_1}, z_q)] += 1$
    **for** all key candidates $k$ **do**
        Initialize a vector $T$ of size $2^{s_1+q}$ to 0
        **for** all $(w_{s_1}, z_q)$ **do**
            Partially decrypt to obtain $y_q$ from $z_q$
            Partially encrypt to obtain $x_{s_1}$ from $w_{s_1}$
            $T[(x_{s_1}, y_q)] += V[(w_{s_1}, z_q)]$
        **for** all $(x_{s_1}, y_q) \in 2^{s_1+q}$ **do**
            $D_k += T[(x_{s_1}, y_q)] \cdot (T[(x_{s_1}, z_q)] - 1)/2$
Sort the counters $D_k$ to obtain the most likely keys.
```

The partial first rounds TD key-recovery attack of Alg. 4 can be done in a time corresponding to N encryptions and $|M| \cdot |K| \cdot |W_{s_1}| \cdot |Z_q|$ partial encryptions using $|W_{s_1}| \cdot |Z_q| + |K|$ counters. The data complexity of the attack can be computed as follows. Here, as we need to check if the difference in the s_1 input bits of the distinguishers are equal to 0, the probability of the TD is $p = 2^{-(s_1+q)}(C+1)$ and need to be compared to the uniform probability $2^{-(s_1+q)}$. In this case the number of required samples is $N_S = \frac{2^{q+s_1}}{C^2}\varphi_a^2$. As with $M \cdot S$ plaintexts we can generate $M \cdot S^2/2$ pairs, the data complexity is $N = \frac{2^{q+s_1+1}}{S \cdot C^2}\varphi_a^2$, where the size S of a structure can be up to $|W_{s_1} \cup W_t|$.

5.2 Chosen-Plaintext Versus Known-Plaintext Attack

When setting $s_1 = s$ and $s_0 = 0$ in Alg. 4, we transform a CP TD attack to a known KP TD attack. In this case with N plaintexts we can generate $N(N-1)/2 \approx N^2/2$ pairs. As in the TD setting the uniform probability is equal to 2^{-q-s}, the number of required samples is $N_S = \frac{2^{q+s}}{C^2} \cdot \varphi_a^2$. The data complexity of a CP TD attack is then equal to the one of a KP ML key-recovery attack: $N^{TD} = N^{ML} = \frac{2^{(s+q+1)/2}}{C}\varphi_a$. The time and memory complexities are then also similar. While all KP attacks can be converted to a CP attack, by this result we show, that in some cases, we can also, with small data, time and memory complexity overhead, convert a CP key-recovery attack to a KP one.

5.3 A Differential Attack on 26 Rounds of PRESENT

In [10], Cho proposed a KP ML attack on 26 rounds of PRESENT. This attack which is based on a combination of 9 ML approximations can be converted to a TD attack in the KP model as presented in the previous section. In this particular case the data complexity is $N^{TD} = N^{ML} = \frac{\sqrt{9 \cdot 2^{8+1}}}{C}\varphi_a$. The time and memory complexities of the TD key-recovery attack are similar to the ones of the ML key-recovery attack.

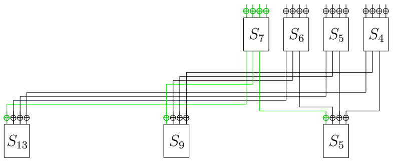
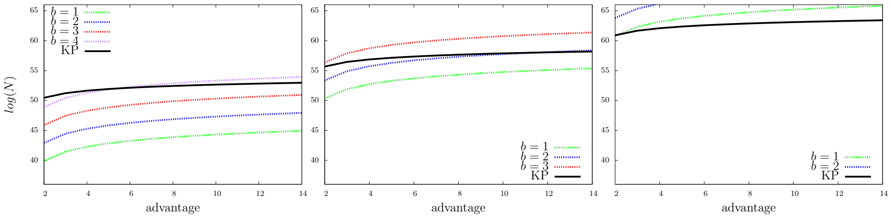


Fig. 2: Partial key recovery on the first round of PRESENT

More importantly, inspired by the KP attack of Cho [10] on PRESENT, we illustrate that, in many cases, when changing from the KP model to the CP model with only a partial key recovery on the first rounds data, time and memory complexity can be reduced.

In the KP ML attack of Cho, 16 bits of keys corresponding to the ones at the input of S_4, S_5, S_6, S_7 (see Fig. 2) are guessed for partial encryption on the first round. If we are in the CP model and we want to only guess part of these 16 key bits, we can specify that the input differences of some Sboxes are equal to 0. We assume that out of the 4 Sboxes S_4, S_5, S_6, S_7, the input of b of them are fixed (see Fig. 2). In this case, $|W_{s_0}| = 2^{4b}$ and we can use structure of size $|W_{s_1}| \cdot |W_t| = 2^{64-4b}$. The data complexity of the attack is then $N = \frac{9 \cdot 2^{4+(4-b)+1}}{2^{64-4b}C^2}\varphi_a^2$. In Fig. 3, we illustrate that depending of the size $|W_{s_0}|$ of the fixation, the data complexity of the key-recovery attack in the CP model can be smaller than in the KP model.



$20 + 2$ rounds, $C = 2^{-44.94}$ $22 + 2$ rounds, $C = 2^{-50.16}$ $24 + 2$ rounds, $C = 2^{-55.38}$

Fig. 3: Evolution of the data complexity of a CP key-recovery attack depending of the size of the fixation and comparison with a KP attack for different number of rounds of PRESENT.

Given the same advantage, if the data complexity of the KP model is smaller than the one in the CP model, it is the same for the time complexity of the distillation phase. By storing only a vector of size $|Z_q| \cdot |W_{s_1}|$ instead of $|Z_q| \cdot |W_s|$, the memory complexity of the CP attack is always smaller than the one of the KP attack. Assuming a fixation of $4b$ bits and independent keys in the first and last rounds the time complexity of the distillation phase (Time_1 in Table 2) corresponds to $N \cdot 2^{-4b}$ partial encryptions and N encryptions. In the CP model 2^{33-4b} counters are necessary for the attack. The time complexity to recover the 80 bit master key (Time_2 in Table 2) is 2^{80-a}. For illustration, we compare in Table 2 the complexities of some KP attacks and CP attacks with $b = 1$ on PRESENT.

Out of the proposed attack on 24 rounds of PRESENT, with complexities summarized in Table 2a we illustrate a case where data, time and memory complexity of the CP are smaller than the ones of the KP attack. Out of the 26-round attack summarized in Table 2b, we show that even when close to the full codebook, the proposed CP attack required less memory than the KP attack.

Table 2: Complexity of attacks on PRESENT for a success probability of 50%. Time_1: Complexity of the distillation phase. Time_2: Complexity of the search phase.

(a) Attacks on 24 rounds of PRESENT, with different complexities

Model	a	Data	Memory	Time_1	Time_2
CP	10	$2^{54.75}$	2^{29}	$2^{54.75}$	2^{70}
KP	5	$2^{57.14}$	2^{33}	$2^{57.14}$	2^{75}

(b) Attacks on 26 rounds of PRESENT, with same advantage

Model	a	Data	Memory	Time_1	Time_2
CP	4	$2^{63.16}$	2^{29}	$2^{63.16}$	2^{76}
KP	4	$2^{62.08}$	2^{33}	$2^{62.08}$	2^{76}

While it has always been assumed that the security of PRESENT in regards to differential cryptanalysis was always better than the one in regards to linear cryptanalysis, these examples illustrate the fact that we can build a CP TD attack on 26 rounds of PRESENT with less memory complexity than the best KP ML attack of [10] done with 2^{64} KP and in time 2^{72}.

6 Links Between Other Statistical Attacks

6.1 Integral, Zero-Correlation and Uniform TD Attacks

Integral cryptanalysis was introduced in [21], and has been used in the literature under the names square, integral or saturation attack. Integral distinguishers mainly make use of the observation that it is possible to fix some parts of the plaintext such that specific parts of the ciphertext are balanced, i.e. each possible partial value occurs the exact same number of times in the output. In practice, the condition of balancedness is typically verified by summing up all partial ciphertexts. In [6], however, in the attack called as zero-correlation integral attack, the authors suggest to store the partial ciphertexts and to verify the proper balancedness condition.

ZC distinguishers [14, 6] are built out of linear approximations with zero bias. In that case the expected capacity of the ML approximation is $C = 0$, and it has been shown in [6] that distinguishing from random can be successful only if no repetition of the plaintexts is allowed. In [6], the authors present a mathematical link between integral and ZC distinguishers. Using this link, the authors of [6] convert a ZC distinguisher on 30 rounds of Skipjack-BABABABA to an integral attack on 31 rounds.

By observing that the output distribution is balanced exactly when the counters $T[y_q]$ of Alg. 3 are all equal, we show that the integral attack is a TD attack where the TD probability of having zero difference in the q bits corresponds to the uniform probability $p = 2^{-q}$. If $q \neq t = n - s$, a ZC distinguisher gives a *uniform truncated differential* distinguisher where the attacker takes advantage of differences which occur uniformly for the cipher.

While the CP integral attack of [6] requires 2^{48} plaintexts and a memory of 2^{32} counters, the same attack on 31 rounds of Skipjack-BABABABA in the KP (without repetition) ZC model would have required roughly the same data and time complexities but a memory of 2^{48} counters. Indeed, the use of structures as in the TD case allows to reduce the memory complexity of the attack as described in Sect. 4.

6.2 Impossible Differential and ML Attacks

Impossible differential cryptanalysis(ID) [15] takes advantage of differentials that never occur. From (2), in the ID case we have $p = 2^{-t}$ and from the formula $p = 2^{-q}(C + 1)$, we deduce that $C = 2^{q-t} - 1$. This formula was used directly in [9] to show the equivalence between the ID and the ZC distinguisher in the case where $t = q$. Nevertheless, in many concrete applications [15, 22, 23], t is small in comparison to n and q is close to n.

It is often assumed that the data complexity N^{ID} of an ID is of order of magnitude $N^{ID} = \mathcal{O}(2^{q-t})$. As the corresponding KP ML distinguisher will require a data complexity of $N^{ML} = \mathcal{O}(\frac{2^{(q+s)/2}}{2^{q-t}-1})$, we discuss in this section the limitations of converting a CP ID distinguisher to a KP ML one.

In practice, ID distinguisher are defined for small $r = n - q$ and $t = n - s$. From $C = 2^{q-t} - 1$, one can note that the ID property occurs only if $q \geq t$. In that case an ID distinguishing attack can be performed using 2^{q-t} plaintexts, in time 2^{q+t} by storing 2^t counters. On the other hand, a KP ML distinguishing attack would require to analyse a distribution of size $2^{(n+q-t)}$ using $2^{(n-q+t)/2}$ known plaintexts. Nevertheless, when the size of the ML distribution is much larger than the data requirement given be the statistical model, the data complexity needs to be adjusted to approximately to $2^{(n+q-t)/2}$ for the χ^2 test to give meaningful results.

While it is possible to find practical ID distinguishers [24] where the data complexity of the ML and ID distinguishers are similar, the time and memory complexity of the ML distinguisher constitutes a limiting factor for this transformation. Nevertheless as the data complexity of a TD or an ID attack is modified when it comes to a key-recovery on the first rounds, it remains an open question to see if we can transform a CP ID key-recovery attack to a KP ML one.

6.3 Classical Differential and Linear Cryptanalysis

As a special case, we see that any classical KP linear distinguishing attack ($s = q = 1$) can be seen as a CP TD distinguishing attack described in Alg. 2b. As summarized in Table 1, both distinguishing attacks have similar complexities. While a last-rounds key-recovery attack remain similar for the CP TD and the KP ML attacks, due to the small fixation $s = 1$, a CP TD key-recovery attack on the first rounds will be equivalent to a KP one. This link has been used previously, although in an implicit manner, for the attack on Salsa and ChaCha [25], where a TD distinguisher with probability $1/2 + \varepsilon$ was extracted.

A transformation from the CP classical differential ($s = q = n - 1$) to a KP ML is does not work in a similar way. While the classical differential distinguisher is memoryless, the ML one required huge memory as shown in Table 1. As in practical security considerations the data complexity typically is close to the full codebook, this KP ML distinguisher has also too high time complexity. A far comparison of the data complexity between these attacks will require more investigation since in that case this one can not be computed from (5).

7 Conclusion

In this paper, we have been investigating many statistical single-key key-recovery attacks on block ciphers both in the KP and CP models. We have shown that many of them are equivalent or that a data-time-memory tradeoffs allow for conversion from a CP to a KP attack. While as shown in Table 3, it is always possible to convert a last-round KP attack of linear type to a CP attack that requires less data, time and/or memory complexity, converting a CP attack to a KP attack is often less profitable. As illustrated by the key-recovery attacks on PRESENT in Sect. 5 on the first rounds, it is not always straightforward to make comparison between KP and CP key-recovery cryptanalysis methods.

The results and links presented in this paper allow to achieve a better understanding of the statistical models of a large number of statistical attacks. For instance, by showing the equivalence between the SS and TD attack, we have been able to compute the data requirement of the SS key-recovery attack.

The attacks are usually called after the method used to derive the distinguisher. For instance, the distinguishers for the differential-linear cryptanalysis are found by combining a truncated differential and a linear approximation. Nevertheless, the attack itself can be treated as a TD attack, see e.g. [26]. In this paper we also presented a concrete example of a distinguisher originally found as a linear property but now used to launch a CP differential attack. It has been a common belief that PRESENT was more secure against differential than linear cryptanalysis, since it is easy to derive linear properties, but practically impossible to compute the probabilities of differential trails. In this paper, we have shown how to derive from the known ML distinguisher a CP differential key-recovery attack on 26 rounds of PRESENT that uses less memory than the previously known KP attack.

We have focused on the most basic ML and TD attacks on block ciphers. We do not claim to have covered them all and many variants and refinements remain to be studied. More generally, it would interesting to analyze our approach in more detail in the context of decorrelation theory [1] which provides a unified framework for all statistical attacks on block ciphers in the single-key model.

Table 3: Links between last-rounds key-recovery attacks

Linear context		Differential context
ML	$\xrightarrow{2^{s+q}<2^n}$	TD = Statistical Saturation
ML	$\xrightarrow{q>t \text{ and } C=2^{-t+q}-1}$	ID (TD with $p_* = 0$)
ZC (ML with $C = 0$)	$\xrightarrow{t>q}$	Integral (TD with $p = 2^{-q}$)
ZC (ML with $C = 0$)	$\xrightarrow{q=t}$	ID (TD with $p_* = 0$)
Linear (ML with $s = q = 1$)	$\longrightarrow$	TD

Acknowledgments

We wish to thank Gregor Leander for useful discussions. We also wish to thank the anonymous reviewers for helpful comments which help us to improve the quality of this paper.

References

1. Vaudenay, S.: Decorrelation: A Theory for Block Cipher Security. J. Cryptology **16** (2003) 249–286
2. Wagner, D.: Towards a Unifying View of Block Cipher Cryptanalysis. In Roy, B.K., Meier, W., eds.: FSE. Volume 3017 of LNCS., Springer (2004) 16–33
3. Leander, G.: On Linear Hulls, Statistical Saturation Attacks, PRESENT and a Cryptanalysis of PUFFIN. In Paterson, K.G., ed.: EUROCRYPT. Volume 6632 of LNCS., Springer (2011) 303–322
4. Collard, B., Standaert, F.X.: A Statistical Saturation Attack against the Block Cipher PRESENT. In Fischlin, M., ed.: CT-RSA. Volume 5473 of LNCS., Springer (2009) 195–210
5. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional Extension of Matsui's Algorithm 2. In: FSE. Volume 5665 of LNCS., Springer (2009) 209–227
6. Bogdanov, A., Leander, G., Nyberg, K., Wang, M.: Integral and Multidimensional Linear Distinguishers with Correlation Zero. In Wang, X., Sako, K., eds.: ASIACRYPT. Volume 7658 of LNCS., Springer (2012) 244–261
7. Knudsen, L.R.: Truncated and Higher Order Differentials. In Preneel, B., ed.: FSE. Volume 1008 of LNCS., Springer (1994) 196–211
8. Chabaud, F., Vaudenay, S.: Links Between Differential and Linear Cryptanalysis. In Santis, A.D., ed.: EUROCRYPT. Volume 950 of LNCS., Springer (1994) 356–365

9. Blondeau, C., Nyberg, K.: New Links between Differential and Linear Cryptanalysis. In Johansson, T., Nguyen, P.Q., eds.: EUROCRYPT. Volume 7881 of LNCS., Springer (2013) 388–404
10. Cho, J.Y.: Linear Cryptanalysis of Reduced-Round PRESENT. In Pieprzyk, J., ed.: CT-RSA. Volume 5985 of LNCS., Springer (2010) 302–317
11. Biham, E., Shamir, A.: Differential Cryptanalysis of DES-like Cryptosystems. In Menezes, A., Vanstone, S.A., eds.: CRYPTO. Volume 537 of LNCS., Springer (1990) 2–21
12. Matsui, M.: Linear Cryptanalysis Method for DES Cipher. In Helleseth, T., ed.: EUROCRYPT. Volume 765 of LNCS., Springer (1993) 386–397
13. Carlet, C.: Boolean Functions for Cryptography and Error Correcting. In Crama, Y., Hammer, P.L., eds.: Boolean Models and Methods in Mathematics, Computer Science, and Engineering. Cambridge University Press, Oxford (2010) 257–397
14. Bogdanov, A., Wang, M.: Zero Correlation Linear Cryptanalysis with Reduced Data Complexity. In Canteaut, A., ed.: FSE. Volume 7549 of LNCS., Springer (2012) 29–48
15. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials. In Stern, J., ed.: EUROCRYPT. Volume 1592 of LNCS., Springer (1999) 12–23
16. Friedman, W.F.: The index of coincidence and its applications in cryptology. Riverbank Laboratories. Department of Ciphers. Publ., 22, Geneva, Ill (1922)
17. Selçuk, A.A.: On Probability of Success in Linear and Differential Cryptanalysis. J. Cryptology **21** (2008) 131–147
18. Blondeau, C., Gérard, B., Tillich, J.P.: Accurate estimates of the data complexity and success probability for various cryptanalyses. Des. Codes Cryptography **59** (2011) 3–34
19. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In Paillier, P., Verbauwhede, I., eds.: CHES. Volume 4727 of LNCS., Springer (2007) 450–466
20. Kerckhof, S., Collard, B., Standaert, F.X.: FPGA Implementation of a Statistical Saturation Attack against PRESENT. In Nitaj, A., Pointcheval, D., eds.: AFRICACRYPT. Volume 6737 of LNCS., Springer (2011) 100–116
21. Daemen, J., Knudsen, L.R., Rijmen, V.: The Block Cipher Square. In Biham, E., ed.: FSE. Volume 1267 of LNCS., Springer (1997) 149–165
22. Luo, Y., Lai, X., Wu, Z., Gong, G.: A unified method for finding impossible differentials of block cipher structures. Information Sciences (In Press)
23. Chen, J., Wang, M., Preneel, B.: Impossible Differential Cryptanalysis of the Lightweight Block Ciphers TEA, XTEA and HIGHT. In Mitrokotsa, A., Vaudenay, S., eds.: AFRICACRYPT. Volume 7374 of LNCS., Springer (2012) 117–137
24. Wu, S., Wang, M.: Automatic Search of Truncated Impossible Differentials for Word-Oriented Block Ciphers. In Galbraith, S.D., Nandi, M., eds.: INDOCRYPT. Volume 7668 of LNCS., Springer (2012) 283–302
25. Aumasson, J.P., Fischer, S., Khazaei, S., Meier, W., Rechberger, C.: New Features of Latin Dances: Analysis of Salsa, ChaCha, and Rumba. In Nyberg, K., ed.: FSE. Volume 5086 of LNCS., Springer (2008) 470–488
26. Blondeau, C., Leander, G., Nyberg, K.: Differential-Linear Cryptanalysis Revisited. In Cid, C., Rechberger, C., eds.: Fast Software Encryption, FSE 2014, Springer-Verlag (To appear)