Revisiting the Wrong-Key-Randomization Hypothesis

Tomer Ashur[1,2], Tim Beyne[1], and Vincent Rijmen[1,3]

[1] imec-COSIC, KU Leuven, Belgium
[2] Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, The Netherlands
[3] University of Bergen, Norway
firstname.lastname@esat.kuleuven.be

Abstract. Linear cryptanalysis is considered to be one of the strongest techniques in the cryptanalyst's arsenal. In most cases, Matsui's Algorithm 2 is used for the key recovery part of the attack. The success rate analysis of this algorithm is based on an assumption regarding the bias of a linear approximation for a wrong key, known as the wrong-key-randomization hypothesis. This hypothesis was refined by Bogdanov and Tischhauser to take into account the stochastic nature of the bias for a wrong key. We provide further refinements to the analysis of Matsui's algorithm 2 by considering sampling without replacement.
This paper derives the distribution of the observed bias for wrong keys when sampling is done without replacement and shows that less data is required in this scenario. It also develops formulas for the success probability and the required data complexity when this approach is taken. The formulas predict that the success probability may reach a peak, then decrease as more pairs are considered. We provide a new explanation for this behavior and derive the conditions for encountering it. We empirically verify our results and compare them to previous work.

Keywords: linear cryptanalysis · wrong-key-randomization hypothesis · success probability · data complexity

1 Introduction

Linear cryptanalysis is considered to be one of the most powerful cryptanalytic techniques. Due to its enormous importance, it is standard practice for algorithm designers to show that their new schemes are resistant to it. This is usually done by providing upper bounds for the cryptographic properties exploited by a linear attack, which is used to argue that the success probability of an adversary is negligible.

From the cryptanalyst's side, much effort was invested in order to improve the attack, either by better understanding it, or by developing possible extensions. Among these extensions we can name the zero correlation attack [7,8], extensions using more than a single approximation such as multiple linear cryptanalysis [3] and multidimensional linear cryptanalysis [16], partitioning cryptanalysis [15], and more. On the front of better understanding the attack we can find research on the linear hull effect [21], various papers suggesting statistical models for analyzing the attack [4,9,17,23], and attempts to quantify the success probability of the various algorithms underlying the attack and the amount of data required.

In his original paper, Matsui estimated that the data complexity should be approximately the squared inverse of the bias of the linear approximation employed in the attack. Selçuk improved this analysis based on the work of Junod and Vaudenay [17] and suggested a complete statistical model, yielding closed formulas for the required data complexity and the success probability. Being able to precisely estimate the success probability of an attack is of great importance to cipher designers as, without it, larger security margins need to be used, in contradiction to the growing trend of using lightweight cryptography.

Estimates of the success probability of linear attacks have traditionally used a simplifying assumption about the behavior of wrong keys. This assumption, commonly known as the *wrong-key-randomization hypothesis* [14], says that when the adversary tries to determine the right key among a set including some wrong keys, the key-dependent bias is much larger for the right key than for wrong ones. The wrong-key-randomization hypothesis was commonly understood to mean that the the bias for a wrong key is exactly zero. However, Bogdanov and Tischhauser noted in [9] that the bias for a wrong key is a random variable with a normal distribution rather than a fixed value. They proposed a corrected wrong-key-randomization hypothesis, taking this distribution into account, and developed a model for the distribution of the empirical bias. Using this model, they have extended Selçuk's formula [23] for the success probability of a linear attack.

An interesting consequence of the new formula is that it reaches a maximum for certain parameters. This result was described as counter-intuitive as it implies that increasing the data complexity may sometimes lead to a reduced success probability [9].

1.1 Our Contributions

In this paper we point out the importance of the sampling strategy employed for obtaining plaintext/ciphertext pairs in a linear attack. In previous work, sampling with replacement was often assumed. We argue that an equally common case is that plaintext/ciphertext pairs are not used more than once, and thus, that sampling without replacement should also be considered. For instance, some modes of operation such as Counter Mode are designed to avoid duplicate plaintext/ciphertext pairs.

Under this assumption, we derive a formula for the success probability and the data complexity of a linear attack for sampling without replacement. This

formula confirms the intuitive notion that, for sampling without replacement, the empirical bias converges faster to its real value, which means that the data complexity can be reduced. Our formula agrees with the recent work of Nyberg and Blondeau in [5] where the notion of "distinct-known-plaintext" is independently introduced for the case of hypothesis testing (see also Section 2.4).

For the purpose of deriving this formula, we redevelop a model for the distribution of the empirical bias for wrong keys. Our result shows that, for sampling without replacement, the correction introduced by Bogdanov and Tischhauser disappears.

We also confirm that the success probability can show non-monotonic behavior as was observed by Tischhauser and Bogdanov [9]. However, their explanation does not account for all of our observations. Hence, we explain the phenomenon anew and derive necessary conditions for its occurrence. The average-case condition, given in Theorem 1, represents a prerequisite for the applicability of Matsui's algorithm 2.

The paper is organized as follows: Section 2 briefly recalls a few basic notions from probability theory and discusses previous work on the wrong-key-randomization hypothesis. In Section 3 we discuss the behavior of the empirical bias. The influence of the sampling strategy is clarified, and the distribution of the sample bias is derived. Section 4 deals with the non-monotonicity of the success probability. The phenomenon is explained, and the conditions for its occurrence are developed. A discussion of the success probability and the data complexity is provided in Section 5. Finally, we verify our results experimentally in Section 6 and discuss their implications in Section 7. Section 8 concludes the paper.

2 Preliminaries and Related Work

A random variable is denoted by bold letters *e.g.*, $\mathbf{X}, \mathbf{Y}$. The expected value of a random variable $\mathbf{X}$ is denoted by $\mathsf{E}[\mathbf{X}]$, and its variance by $\mathsf{Var}[\mathbf{X}]$. The conditional random variable $\mathbf{X}$ given $\mathbf{Y}$ is denoted by $\mathbf{X} \mid \mathbf{Y}$. This notation carries over to conditional expectations and variances. By writing $\mathbf{X} \sim \mathcal{N}\left(\mu, \sigma^2\right)$, we mean that $\mathbf{X}$ follows a normal distribution with mean μ and variance σ^2. Similarly, $\mathbf{X} \sim \text{Hypergeometric}(N, M, R)$ means that $\mathbf{X}$ follows a hypergeometric distribution, *i.e.*, $\mathbf{X}$ is a random variable counting the number of occurrences of an item of type I in N draws from a population of size M known to include R such items, where the draws are performed without replacement. The standard normal cumulative distribution function will be denoted by Φ, *i.e.* $\Phi(z) = \Pr[\mathbf{Z} < z]$ for $\mathbf{Z} \sim \mathcal{N}(0, 1)$.

During our analysis, it will frequently be convenient to approximate the hypergeometric distribution. Several accurate χ^2 and normal approximations exist, see for example [20]. For our purposes, the following result given by Feller [12] shall suffice. The interested reader may find two proofs of the result by Pinsky in [22].

Lemma 1 (Feller [12]) *Let* $\mathbf{X} \sim$ *Hypergeometric*(N, M, pM)*. If* $N, M \to \infty$ *in such manner that* $N/M \to t \in (0, 1)$*, then* $\mathbf{X}$ *has asymptotic distribution*

$$\mathbf{X} \sim \mathcal{N}\left(pN, N(1-t)p(1-p)\right).$$

The factor $(1-t)$ in the variance of $\mathbf{X}$ in Lemma 1 may be interpreted as a correction factor that accounts for the difference between sampling with and without replacement.

2.1 Linear Cryptanalysis

We now describe linear cryptanalysis [19]. Let $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an n-bit permutation. A linear approximation for f is a pair of masks $(\alpha, \beta) \in \mathbb{F}_2^n \times \mathbb{F}_2^n$ such that $\alpha^t f(\mathbf{Y}) = \beta^t \mathbf{Y}$ holds with probability p for $\mathbf{Y}$ uniform over $\mathbb{F}_2^n$. The bias of the approximation is defined as $\epsilon = p - 1/2$. In the following, the bit-length of the round key of the cipher under attack will be denoted by m.

Without loss of generality, in a key-recovery attack using Matsui's Algorithm 2, the approximation (α, β) covers $R-1$ rounds of an R-round cipher used to encrypt N plaintexts. The resulting N ciphertexts are then 1-round decrypted using 2^m different key guesses, and the linear approximation is evaluated against the resulting pairs. For each key k_i, the adversary keeps a counter $\mathbf{T}_i$ counting the number of times a pair satisfies the linear approximation. Once enough data has been observed, the adversary calculates the empirical bias $\hat{\boldsymbol{\epsilon}}_i = \mathbf{T}_i/N - 1/2$. The biases are sorted in descending order according to their magnitude (*i.e.* absolute value), and their respective keys are tried in this order. If the bias corresponding to the right key is ranked among the highest 2^{m-a} biases, the attacker is said to obtain an advantage a over brute-force. Throughout our paper, the subscript zero refers to the right key. In particular, the counter and bias for the correct key are denoted respectively by $\mathbf{T}_0$ and ϵ_0.

The success probability P_S of the above procedure is defined as follows. Note that Definition 1 considers the success rate for a fixed right-key bias. As will be discussed in Section 3.2, in practice a distribution of right-key biases may have to be considered.

Definition 1 (Success probability) *The success probability* P_S *is the probability that Matsui's Algorithm 2 ranks the right key among the top* 2^{m-a} *candidate keys. Note that* P_S *is a function of the right-key bias* ϵ_0*, the number of known plaintexts* N *and the advantage* a*.*

2.2 The Wrong-Key-Randomization Hypothesis

The success rate analysis performed by Selçuk [23] uses order statistics to investigate the probability that the right key is among the 2^{m-a} top ranked keys. The main underlying assumption of this analysis is that the real bias for a wrong key is zero, and thus, that the sample bias for a wrong key would have a normal distribution centered around zero. This assumption may be summarized in the following hypothesis:

Hypothesis 1 (Simple wrong-key-randomization hypothesis) *The bias for a wrong key equals zero:*

$$\epsilon_w = 0.$$

If Hypothesis 1 is true, we have the following lemma.

Lemma 2 *Let $\hat{\epsilon}_w$ be the empirical bias obtained from a counter associated with a wrong key using N pairs of plaintexts and ciphertexts. Assuming Hypothesis 1 is true and sampling is performed with replacement, we have approximately*

$$\hat{\boldsymbol{\epsilon}}_w \sim \mathcal{N}\left(0, \frac{1}{4N}\right).$$

However, Bogdanov and Tischhauser noted in [9] that, in accordance with Daemen and Rijmen [11], the underlying bias of a random linear approximation is not necessarily zero but a random variable. This resulted in the following extension of Hypothesis 1:

Hypothesis 2 (Bogdanov and Tischhauser [9]) *The bias $\boldsymbol{\epsilon}_w$ for a wrong key is distributed as for a random permutation,* i.e.

$$\boldsymbol{\epsilon}_w \sim \mathcal{N}\left(0, 2^{-n-2}\right).$$

This hypothesis requires the usage of a compound model for the empirical bias, which takes into account the distribution of the wrong bias. This leads to the following statement about the distribution of the sample bias for wrong keys.

Lemma 3 (Bogdanov and Tischhauser [9, Lemma 1]) *Let $\hat{\boldsymbol{\epsilon}}_w$ be defined as before, then assuming the validity of Hypothesis 2, we have approximately*

$$\hat{\boldsymbol{\epsilon}}_w \sim \mathcal{N}\left(0, \frac{1}{4} \cdot \left(\frac{1}{N} + \frac{1}{2^n}\right)\right).$$

Selçuk gives the success probability of a linear attack as

$$P_S(N) = \Phi\left(2\sqrt{N}|\epsilon_0| - \Phi^{-1}(1 - 2^{-a-1})\right), \tag{1}$$

which holds under Hypothesis 1. However, as was noted by Bogdanov and Tischhauser, the bias for wrong keys has a normal distribution centered around zero, in accordance with Hypothesis 2. Using the distribution of Lemma 3, they extend Selçuk's formula as follows:

$$P_S(N) = \Phi\left(2\sqrt{N}|\epsilon_0| - \sqrt{1 + \frac{N}{2^n}}\Phi^{-1}(1 - 2^{-a-1})\right). \tag{2}$$

An experimental verification of the above formula is provided in [6].

2.3 Compound Model

It is important to distinguish between two different random variables: the sample bias for a specific wrong key, and the sample bias for a uniformly selected random wrong key. We shall refer to the former by $\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w$ and the latter will be written as $\hat{\boldsymbol{\epsilon}}_w$. The idea of a compound model is that a parameter in the distribution of a random variable is itself a random variable. The probability density of the compound variable $\hat{\boldsymbol{\epsilon}}_w$ is given by the probability density of $\hat{\boldsymbol{\epsilon}}_w \mid (\boldsymbol{\epsilon}_w = \varepsilon)$, weighted by the probability that $\boldsymbol{\epsilon}_w = \varepsilon$ for any ε that $\boldsymbol{\epsilon}_w$ can take. Formally, if $f_{\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w}$ is the probability density function of $\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w$, and $f_{\boldsymbol{\epsilon}_w}$ likewise for $\boldsymbol{\epsilon}_w$, then we may write

$$f_{\hat{\boldsymbol{\epsilon}}_w}(\epsilon_w) = \int_{\varepsilon} f_{\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w}(\epsilon_w, \varepsilon) f_{\boldsymbol{\epsilon}_w}(\varepsilon) d\varepsilon,$$

for the density of $\hat{\boldsymbol{\epsilon}}_w$. This is depicted in Figure 1.

The behavior of the random variable $\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w$ is completely determined by the sampling strategy. For example, for sampling without replacement, the counter $\mathbf{T}_w \mid \boldsymbol{\epsilon}_w$ follows a hypergeometric distribution centered around $N\left(\frac{1}{2} + \epsilon_w\right)$.

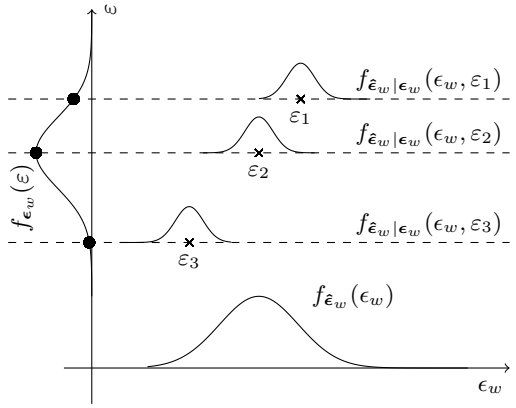

Fig. 1: The curve along the vertical axis represents the density function of $\boldsymbol{\epsilon}_w$ (the bias for a random wrong key). The probability density function of $\hat{\boldsymbol{\epsilon}}_w$ (the sample bias for a random wrong key) is shown at the bottom. $\hat{\boldsymbol{\epsilon}}_w$ has a compound distribution obtained by weighted integration over the smaller curves which represent the sample biases for specific keys.

Since the right key bias depends on the unknown value of the right key, an additional compound model must be introduced. In order to deal with this case, a right key hypothesis is required. This will be discussed in more detail in Section 3.2.

2.4 Comparison with Blondeau and Nyberg [5]

The idea of using sampling without replacement for linear cryptanalysis was recently introduced by Blondeau and Nyberg [5]. Their analysis was developed independently from ours and follows the line of work using hypothesis testing, whereas ours is based on key ranking. Although both approaches lead to similar success probabilities, much of the existing literature uses key ranking, and our results allow to easily evaluate the implications of such works.

Unlike [5], our results do not assume a particular right-key-hypothesis. This is possible by the use of a compound model: all results can be derived for a fixed right key bias and the distribution afterwards. This is discussed in more detail in Section 3.2.

Finally, we also note that the possibility of a non-monotonic success probability is not considered in [5]. We discuss this phenomenon in detail in Section 4 and provide a complete explanation.

3 Sample Bias

This section deals with the distribution of the sample bias for wrong keys. The details of this distribution are relevant for the construction and the analysis of statistical procedures that attempt to distinguish the right key from wrong keys.

As mentioned in the related work section, the distribution of $\hat{\boldsymbol{\epsilon}}_w$ must be described by a compound model whenever Hypothesis 2 is used. Generally speaking, the sample bias of wrong keys can be fully described given the distribution of the bias for the wrong keys and a *sampling strategy*. The former is completely determined by the choice of Hypothesis 1 or 2 and requires no further discussion. The latter will be discussed in Section 3.1.

Finally, the main result of this section is presented in Lemma 4, which approximates the distribution of the sample bias for a random wrong key in the case of sampling without replacement. Surprisingly, the resulting distribution turns out to be the same as the one given in Lemma 2.

3.1 Sampling Strategies

The way plaintext/ciphertext pairs are obtained in linear cryptanalysis corresponds to sampling from a population of size 2^n. For each sample, a trial is conducted: it is checked whether or not a fixed linear approximation holds. Recall that the sum of the outcomes (zero or one) of these trials is stored in a counter $\mathbf{T}_i$.

One could conceive many strategies for sampling, but here we will limit the discussion to two common cases:

1. Sampling with replacement, trials are independent and $N > 2^n$ is possible.[1]
2. Sampling without replacement. Trials are not independent and $N \leq 2^n$.

The use of the first sampling strategy leads to a binomial distribution for the counters. The existing analyses that we are aware of [4, 9, 23] implicitly start from this assumption. An exception is the notion of "distinct-known-plaintext" attacks in recent work by Blondeau and Nyberg [5], discussed in Section 2.4, which was developed independently from the present paper.

We now argue that the second strategy is preferable. This will lead to a hypergeometric distribution for the counters. Since for a given key, a specific plaintext always results in the same ciphertext, duplicates provide no new information for the estimation of the real bias. Moreover, increasing the data complexity beyond what is needed for the attack reduces its efficiency, and may render it worse than exhaustive search when the bias is small. Hence, whenever possible, an adversary would prefer sampling without replacement. The disadvantage of sampling without replacement is that the attacker must discard duplicate plaintext/ciphertext pairs, which may require a large amount of memory. Nevertheless, even for known-plaintext attacks, sampling without replacement can occur naturally. For example, some modes of operation such as Counter Mode avoid duplicate plaintext/ciphertext pairs by design.

3.2 Bias for the Right Key

In expressions (1) and (2) given in Section 2 for the success probability, the absolute bias for the right key is represented by a fixed value $|\epsilon_0|$. In practice, however, the right-key bias depends on the unknown value of the right key. The absolute right-key bias should then be modeled as a random variable $|\boldsymbol{\epsilon}_0|$. Hence, it is necessary to find an adequate model for the distribution of the bias for the right key. This is an independent problem, which we do not attempt to solve in this paper. Instead, we will assume that the probability density function $f_{\boldsymbol{\epsilon}_0}(\varepsilon)$ of the bias for the right key is known. Several proposals for such a *right-key hypothesis* can be found in the literature, see for instance [1, 5, 9, 10].

In the setting described above, the success probability becomes a random variable $\mathbf{P_S}(N)$, or more explicitly $\mathbf{P_S}(N, \boldsymbol{\epsilon}_0)$. Typically, one is interested in the average success probability $\mathsf{E}[\mathbf{P_S}(N, \boldsymbol{\epsilon}_0)]$. That is,

$$\mathsf{E}[\mathbf{P_S}(N, \boldsymbol{\epsilon}_0)] = \int_{-1/2}^{1/2} P_S(N, \varepsilon) f_{\boldsymbol{\epsilon}_0}(\varepsilon) d\varepsilon. \tag{3}$$

Despite this complication, we shall continue to use the notation $P_S(N)$ in the sense that $P_S(N) = \mathsf{E}[\mathbf{P_S}(N, \boldsymbol{\epsilon}_0)]$ when the absolute value of the right key bias

[1] Note that, by the Coupon Collector's problem, it is likely that every plaintext/ciphertext pair has been sampled at least once when $N > n2^n$ for sampling with replacement.

is fixed. Furthermore, we will continue to use the term $|\epsilon_0|$ to denote the actual right key bias selected from the appropriate distribution as a result of fixing the key.

Finally, we note that Hypothesis 2 makes no mention of the right key. The dependence of the biases for wrong keys on the right key is neglected.

3.3 Sampling without Replacement

Given that duplicate draws provide no additional information to the cryptanalyst, we would like to analyze the attack when sampling without replacement is used.

Assume then that N distinct plaintext/ciphertext pairs are sampled at random from the total population of 2^n pairs. The counter for a specific wrong key follows a hypergeometric distribution

$$\mathbf{T}_w \mid \mathbf{R} \sim \text{Hypergeometric}(N, 2^n, R),$$

where $R = 2^n(\epsilon_w + 1/2)$ equals the amount of plaintext/ciphertext pairs in the population for which the linear approximation holds. Given this starting point, the proof of the next lemma derives the distribution of the sample bias for a random wrong key.

Lemma 4 (Lemma 2, *stet.*) *Under Hypothesis 2, and for sampling without replacement, we have for the sample bias $\hat{\boldsymbol{\epsilon}}_w$ of a random wrong key*

$$\hat{\boldsymbol{\epsilon}}_w \sim \mathcal{N}\left(0, \frac{1}{4N}\right),$$

approximately.

Proof. By Lemma 1 we have asymptotically

$$\mathbf{T}_w \mid \boldsymbol{\epsilon}_w \sim \mathcal{N}\left(N\left(\frac{1}{2} + \epsilon_w\right), N\left(1 - \frac{N}{2^n}\right)\left(\frac{1}{4} - \epsilon_w^2\right)\right).$$

Note that this application of Lemma 1 provides an accurate approximation of the hypergeometric distribution since N and 2^n are large in all relevant cases. Since ϵ_w^2 is small, we have approximately

$$\mathbf{T}_w \mid \boldsymbol{\epsilon}_w \sim \mathcal{N}\left(N\left(\frac{1}{2} + \epsilon_w\right), \frac{N}{4}\left(1 - \frac{N}{2^n}\right)\right).$$

It follows that for the sample bias

$$\hat{\boldsymbol{\epsilon}}_w \mid \boldsymbol{\epsilon}_w \sim \mathcal{N}\left(\epsilon_w, \frac{1}{4N}\left(1 - \frac{N}{2^n}\right)\right).$$

A compound normal distribution with normally distributed mean is again normal. That is, if $\mathbf{X} \sim \mathcal{N}\left(\boldsymbol{\mu}, \sigma_1^2\right)$ with $\boldsymbol{\mu} \sim \mathcal{N}\left(\mu, \sigma_2^2\right)$, then $\mathbf{X} \sim \mathcal{N}\left(\mu, \sigma_1^2 + \sigma_2^2\right)$.[2] In this particular case we obtain

$$\begin{aligned} \mathsf{E}\left[\hat{\boldsymbol{\epsilon}}_w\right] &= 0 \\ \mathsf{Var}\left[\hat{\boldsymbol{\epsilon}}_w\right] &= \frac{1}{4N}\left(1 - \frac{N}{2^n}\right) + \frac{1}{2^{n+2}} = \frac{1}{4N}. \end{aligned}$$

□

The mean and variance of $\mathbf{T}_w$ can also be computed directly, without using any of the approximations used in Lemma 4. This derivation can be found in Appendix A and leads to approximately the same result.

Conclusion: The preceding lemma shows that $\hat{\boldsymbol{\epsilon}}_w$ for sampling without replacement has approximately the same distribution as given by Lemma 2. In other words, the correction provided in Lemma 3 should *not* be taken into account for sampling without replacement. Note, however, that the result is based on Hypothesis 2 rather than Hypothesis 1.

4 Non-Monotonicity of the Success Probability

So far, we derived the distribution of the empirical bias for a uniformly chosen wrong key when the attack is executed with distinct data pairs. We now turn to investigate the success probability of such an attack, and explain the non-monotonic behaviour first observed in [9].

For sampling with replacement, Selçuk estimates the success probability as given by (1). Bogdanov and Tischhauser have extended this result to (2). Due to the fact that the biases for the wrong keys are selected at random from $\mathcal{N}\left(0, 2^{-n-2}\right)$, the success probability should be considered as a random variable. Recall from Section 3.2 that $P_S(N)$ denotes the mean of the success probability $\mathbf{P_S}(N)$ when the absolute right key bias is fixed. This is the value computed in (1) and (2). We give two results related to two aspects of the random variable $\mathbf{P_S}(N)$:

— According to (2), $P_S(N)$ can be non-monotonic. This effect was also observed in [9]. We provide an explanation for this phenomenon, as well as necessary conditions for its occurrence.
— In Section 5, we derive a formula for the average success probability in the case of sampling without replacement.

4.1 Explanation of Non-Monotonicity

Bogdanov and Tischhauser have observed that, in some cases, the success probability exhibits a maximum [9]. They attribute this effect to the fact that, as N

[2] This follows from the observation that $\mathbf{X} - \boldsymbol{\mu} \sim \mathcal{N}\left(0, \sigma_1^2\right)$ is independent of $\boldsymbol{\mu}$.

increases, the number of duplicate samples increases which amplifies the "noise" due to the random distribution of the biases for the wrong keys. In the present section we propose a different explanation for the non-monotonicity and show that the phenomenon is not counter-intuitive. Note that a maximum can also be observed for sampling without replacement, hence without duplicates, which is difficult to reconcile with the explanation given by [9]. The discussion below and the conditions given in the next section are independent of the sampling strategy.

In the following, we will assume that the bias of the right key is fixed. This is the setting in which Bogdanov and Tischhauser have observed non-monotonicity [9]: a subset of keys was used for which the bias of the linear hull was exactly equal to ϵ_0. It can be seen from (3) that, depending on the right-key hypothesis, the *average* success probability might never show non-monotonicity. This is for instance the case when the right-key hypothesis proposed by [5] is used, provided that certain assumptions on the distribution of the right key bias are met. Nevertheless, if we restrict our attention to the success probability for a subset of keys for which the right key bias does not vary much, non-monotonicity remains possible.

When the bias ϵ_0 of the right key is close to zero, there is a non-negligible probability that some of the wrong keys have a higher absolute bias than $|\epsilon_0|$. This is depicted in Figure 2. In this case, the correct key should *not* be expected to be ranked higher than (some of the) wrong keys. As N increases, the accuracy of the ranking increases because the variances of all sample biases decrease. It follows that, if there are wrong keys with absolute bias higher than $|\epsilon_0|$, then for large N those will be ranked higher than the right key. If there are more such keys than the attacker advantage allows, *i.e.* more than 2^{m-a}, then, due to the sample variance, the right key may start high (among the top 2^{m-a} values) on the list of candidate keys as a false-positive but will slowly drop down to its "real" position (below 2^{m-a}) due to improved accuracy as a result of using more data. In this case, if $N \to \infty$ (or $N = 2^n$ without replacement) then also $\mathbf{P_S}(N) \to 0$, almost surely. In other words: given all possible information, the attack will always fail.

One can conclude that two different scenarios exist for the success probability of the linear attack, depending on the bias of the right key. This is depicted in Figure 3.

As indicated in Figure 3, there is a threshold such that any bias that exceeds it in magnitude corresponds to a success probability which is monotonic with probability higher than 50%. For example, a set of keys with bias ϵ_1 will have a monotonic average success probability whereas biases ϵ_2 and ϵ_3 would lead to non-monotonicity. A special case is ϵ_4, because it is very close to the threshold. In this case, the success probability will be monotonic for about half of the keys that have this bias. Theorem 1 says that the threshold equals $2^{-n/2-1}\Phi^{-1}(1-2^{-a-1})$.

From the discussion above, we may conclude that non-monotonic behavior indicates that the attack can not be conducted using Matsui's algorithm 2. In fact, a correct identification of the right key amounts to a false positive. This is

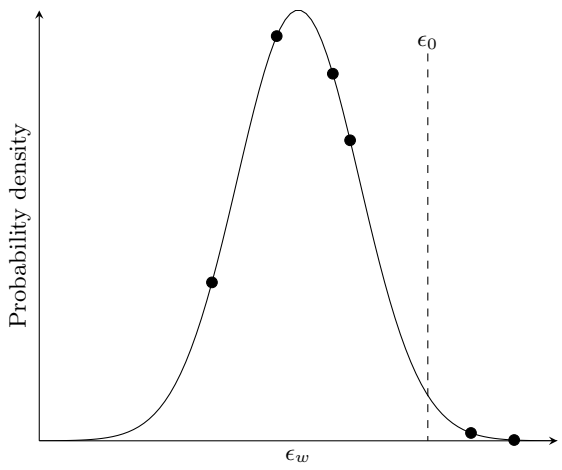

Fig. 2: The biases for a few keys are indicated by dots, the dashed line represents the bias for the right key. Two of the wrong keys have a larger bias than the right key. If the adversary requires such an advantage that the right key needs to be among the top two keys, the attack would fail once enough data is obtained to place the keys in their true order.

formalized in the next section, by giving a bound on the required bias for given values of 2^n and a. This bound hence also expresses a prerequisite for Matsui's algorithm 2.

4.2 Conditions for Non-Monotonicity

This subsection derives necessary conditions for non-monotonic behavior of the success probability. These conditions are necessarily probabilistic, since they are determined by the biases of individual wrong keys. Hence, it can be expected that for some values of a, n, m and ϵ_0, $\mathbf{P_S}(N)$ is non-monotonic only for *some*

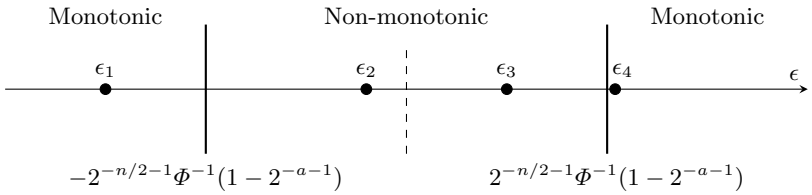

Fig. 3: Depending on the right key bias, the success probability can be monotonic (ϵ_1, ϵ_4) or non-monotonic (ϵ_2, ϵ_3).

keys. The main result of this section is Theorem 1, which gives a necessary condition for average non-monotonicity and hence for the applicability of Matsui's algorithm 2.

The following lemma gives a first result on the probability of monotonicity. It is independent of the sampling strategy.

Lemma 5 *The probability that* $\mathbf{P_S}(N)$ *is a monotonic function is given by*

$$\Pr\left[\mathbf{P_S}(N) \text{ is monotonic}\right] = \Phi\left(\frac{\left(2^{-a} - 2^{-m-1} - p\right)2^{m/2}}{\sqrt{p(1-p)}}\right),$$

where

$$p = 2\left(1 - \Phi\left(|\epsilon_0| 2^{n/2+1}\right)\right).$$

and ϵ_0 *is the bias for the right key as defined in Section 3.2.*

Proof. The probability that a random permutation has absolute bias larger than $|\epsilon_0|$ can be computed as

$$p = \Pr\left[|\boldsymbol{\epsilon}_w| \geq |\epsilon_0|\right] = 2(1 - \Pr\left[\boldsymbol{\epsilon}_w < |\epsilon_0|\right]) = 2\left(1 - \Phi\left(|\epsilon_0| 2^{n/2+1}\right)\right),$$

since $\boldsymbol{\epsilon}_w \sim \mathcal{N}\left(0, 2^{-n-2}\right)$. For a non-monotonic success probability, we require at least 2^{m-a} wrong keys with bias larger than $|\epsilon_0|$. Let $\mathbf{C}$ be the random variable describing the number of such keys, then $\mathbf{C}$ is binomially distributed. Furthermore, if the number of keys 2^m is sufficiently large, $\mathbf{C}$ can be approximated with a normal distribution:

$$\mathbf{C} \sim \mathcal{N}\left(p2^m, p(1-p)2^m\right) \text{ where } p = 2\left(1 - \Phi\left(|\epsilon_0| 2^{n/2+1}\right)\right).$$

The probability that $P_S(N)$ is monotonic for some $|\epsilon_0|, m$ and a can hence be computed as

$$\Pr\left[\mathbf{P_S}(N) \text{ is monotonic}\right] = \Pr\left[\mathbf{C} < 2^{m-a}\right].$$

Using the normal approximation of $\mathbf{C}$, we get:

$$\begin{aligned}\Pr\left[\mathbf{P_S}(N) \text{ is monotonic}\right] &= \Phi\left(\frac{2^{m-a} - 2^{-1} - p2^m}{\sqrt{p(1-p)2^m}}\right) \\ &= \Phi\left(\frac{(2^{-a} - 2^{-m-1} - p)2^{m/2}}{\sqrt{p(1-p)}}\right).\end{aligned}$$

□

In the previous lemma, a fixed absolute right-key bias $|\epsilon_0|$ was assumed. When $\boldsymbol{\epsilon}_0$ has a known probability distribution, we can compute the average probability of monotonicity from Lemma 5 by

$$\Pr\left[\mathbf{P_S}(N) \text{ is monotonic}\right] = \int_{-1/2}^{1/2} \Phi\left(\frac{\left(2^{-a} - 2^{-m-1} - p(\varepsilon)\right)2^{m/2}}{\sqrt{p(\varepsilon)(1-p(\varepsilon))}}\right) d\varepsilon,$$

with $p(\varepsilon)$ as before:

$$p(\varepsilon) = 2\left(1 - \Phi\left(|\varepsilon| 2^{n/2+1}\right)\right).$$

Observe that, as the bias $\boldsymbol{\epsilon}_0$ approaches 0 in probability, the success probability is almost surely non-monotonic. If $\mathsf{E}[\mathbf{C}] \geq 2^{m-a}$, then $\mathbf{P_S}(N)$ is non-monotonic on average, *i.e.* over all keys and all attacks. This condition can be used to derive the following theorem, which assumes a fixed absolute bias for the right key.

Theorem 1 (Prerequisite of Matsui's algorithm 2) *The success probability is monotonic on average if and only if*

$$|\epsilon_0| > 2^{-n/2-1}\Phi^{-1}(1 - 2^{-a-1}).$$

Proof. The condition $\mathsf{E}[\mathbf{C}] \geq 2^{m-a}$ corresponds to the inequality

$$2\left(1 - \Phi\left(|\epsilon_0| 2^{n/2+1}\right)\right) 2^m \geq 2^{m-a},$$

which can be rewritten as

$$\begin{aligned} & 1 - \Phi\left(|\epsilon_0| 2^{n/2+1}\right) \geq 2^{-a-1} \\ \iff & |\epsilon_0| \leq 2^{-n/2-1}\Phi^{-1}(1 - 2^{-a-1}). \end{aligned}$$

□

Theorem 1 expresses a necessary condition for a nonzero success probability in the average case as $N \to \infty$ (or $N = 2^n$ without replacement). Hence, when the condition does not hold, the advantage a can only be obtained as a false-positive during key recovery with Matsui's algorithm 2.

It can be verified that the condition of Theorem 1 ensures that the maximum of $P_S(N)$ as defined by equation (2) corresponds to a positive value of N.

Bogdanov and Tischhauser have observed non-monotonic behavior with $|\epsilon_0| = 2^{-10}, a = 12$ and $n = 20$ [9]. Theorem 1 gives:

$$p = 2\left(1 - \Phi(2^{-10} \cdot 2^{11})\right) \approx 0.0455.$$

and

$$|\epsilon_0| \leq 2^{-11}\Phi^{-1}(1 - 2^{-13}) \approx 2^{-9.125}.$$

Hence, with these parameters, the average attack setup will lead to non-monotonic behavior. By Lemma 5, the probability of monotonicity is $\Phi(-218.78) \approx 0$.

5 Average Success Probability and Data Complexity

In this section, we will derive formulas for the average success probability of an attack using sampling without replacement and the data complexity required for a successful attack.

To compute the average success probability, we will make the approximation that the non-identically distributed sample biases for wrong keys can be replaced by an equal amount of independent and identically distributed random variables with distribution given by Lemma 4. A similar approximation was also implicitly made in [9] and greatly simplifies the distribution of the order statistics.

The derivation of $P_S(N)$ is similar to that of Selçuk [23], with the important difference that the counter for the right key is distributed as

$$\mathbf{T}_0 \sim \text{Hypergeometric}\left(N, 2^n, \left(\frac{1}{2} + \epsilon_0\right) 2^n\right). \tag{4}$$

The corresponding distribution function of $\hat{\boldsymbol{\epsilon}}_0$ will be denoted by $F_{\hat{\boldsymbol{\epsilon}}_0}$ and can be written in terms of the distribution function $F_{\mathbf{T}_0}$ of $\mathbf{T}_0$:

$$F_{\hat{\boldsymbol{\epsilon}}_0}(\varepsilon) = F_{\mathbf{T}_0}\left(\frac{N}{2} + N\varepsilon\right).$$

Following Selçuk, without loss of generality, we only consider the case $\epsilon_0 \geq 0$. The discussion for $\epsilon_0 < 0$ is completely analogous. It will be assumed that the distribution of an order statistic of the sample biases $\hat{\boldsymbol{\epsilon}}_w$ for wrong keys is approximately degenerate relative to that of the right key — Selçuk makes the same approximation in his discussion. The mean of the $(2^m - 2^{m-a})$th order statistic $\boldsymbol{\zeta}$ is approximately given by $\mathsf{E}[\boldsymbol{\zeta}] = \Phi^{-1}(1-2^{-a-1})/(2\sqrt{N})$ [23]. Noting that $\Pr[\hat{\boldsymbol{\epsilon}}_{\mathbf{0}} < 0] \approx 0$, we have for the average success probability

$$\begin{aligned}
P_S(N) &= \Pr\left[\hat{\boldsymbol{\epsilon}}_0 - \boldsymbol{\zeta} > 0\right] \\
&\approx \Pr\left[\hat{\boldsymbol{\epsilon}}_0 > \mathsf{E}[\boldsymbol{\zeta}]\right] \\
&= 1 - F_{\hat{\boldsymbol{\epsilon}}_0}\left(\frac{\Phi^{-1}(1-2^{-a-1})}{2\sqrt{N}}\right) \\
&= 1 - F_{\mathbf{T}_0}\underbrace{\left(\frac{N}{2} + \frac{\sqrt{N}}{2}\Phi^{-1}(1-2^{-a-1})\right)}_{k(N)}.
\end{aligned}$$

An accurate approximation of $F_{\mathbf{T}_0}$ can be obtained by using a normal approximation with respect to N. Indeed, by applying Lemma 1 to $\mathbf{T}_0$, one obtains the approximation (assuming $\epsilon_0^2 \approx 0$)

$$\mathbf{T}_0 \sim \mathcal{N}\left(N\left(\frac{1}{2} + |\epsilon_0|\right), \left(1 - \frac{N}{2^n}\right)\frac{N}{4}\right),$$

which is accurate if N and 2^n are sufficiently large. It can be verified that the above expression also holds for $\epsilon_0 < 0$. In terms of the standard normal distri-

bution, we have

$$
\begin{aligned}
F_{\mathbf{T}_0}(k(N)) &\approx \Phi\left(\frac{k(N) - N\left(\frac{1}{2} + |\epsilon_0|\right)}{\sqrt{\frac{N}{4}\left(1 - \frac{N}{2^n}\right)}}\right) \\
&= \Phi\left(\frac{\sqrt{N}\Phi^{-1}(1 - 2^{-a-1})/2 - N|\epsilon_0|}{\sqrt{\frac{N}{4}\left(1 - \frac{N}{2^n}\right)}}\right) \\
&= \Phi\left(\frac{\Phi^{-1}(1 - 2^{-a-1}) - 2\sqrt{N}|\epsilon_0|}{\sqrt{1 - \frac{N}{2^n}}}\right).
\end{aligned}
$$

By symmetry, we then obtain the simple result of the theorem below.

Theorem 2 *Assume Hypothesis 2 holds. Let $P_S(N)$ denote the average success probability of a linear attack on an n-bit block cipher given N distinct known plaintext/ciphertext pairs. If the bias of the right key is ϵ_0 and the desired advantage is a, then we have*

$$
P_S(N) \approx \Phi\left(\frac{2\sqrt{N}|\epsilon_0| - \Phi^{-1}(1 - 2^{-a-1})}{\sqrt{1 - \frac{N}{2^n}}}\right),
$$

for sampling without replacement.

Note that in the above expression for the success probability, and in the preceding discussion, we have assumed that the bias ϵ_0 is fixed. In practice, this is not the case and instead the right-key hypothesis should be taken into account. Recall from Section 3.2 that the average success probability can be obtained as

$$
\mathsf{E}\left[\mathbf{P_S}(N, \boldsymbol{\epsilon}_0)\right] = \int_{-1/2}^{1/2} P_S(N, \varepsilon) f_{\boldsymbol{\epsilon}_0}(\varepsilon) d\varepsilon, \tag{3}
$$

where $f_{\boldsymbol{\epsilon}_0}(\varepsilon)$ is the probability density function of $\boldsymbol{\epsilon}_0$.

Theorem 2 directly leads to an expression for the data complexity, which is given below. Note that when $P_S(N)$ is non-monotonic, the value of the success probability P_S will in general correspond to two data complexities N. For simplicity, and because monotonicity is a prerequisite for practical usage of Matsui's algorithm 2, we only deal with the monotonic case.

Theorem 3 *Under the same conditions as Theorem 2, and when the condition given by Theorem 1 is satisfied, the number of plaintext/ciphertext pairs required to obtain an average success probability P_S is*

$$
N = \left(\frac{2|\epsilon_0|\alpha \pm \sqrt{(2\epsilon_0\alpha)^2 - (\alpha^2 - \beta^2)(2^{-n}\beta^2 + 4\epsilon_0^2)}}{4\epsilon_0^2 + 2^{-n}\beta^2}\right)^2,
$$

where $\alpha = \Phi^{-1}(1 - 2^{-a-1})$ *and* $\beta = \Phi^{-1}(P_S)$*. The plus sign applies whenever* $P_S \geq 1/2$*, otherwise the minus sign applies. For* $|\epsilon_0| \gg 2^{-n/2-1}\Phi^{-1}(P_S)$*, this simplifies to*

$$N = \left(\frac{\Phi^{-1}(1 - 2^{-a-1}) + \Phi^{-1}(P_S)}{2\epsilon_0}\right)^2.$$

Proof. The result is obtained by solving the equation

$$\Phi^{-1}(P_S)\sqrt{1 - \frac{N}{2^n}} = 2\sqrt{N}|\epsilon_0| - \Phi^{-1}(1 - 2^{-a-1}).$$

A trite calculation (*cf.* Appendix B) yields

$$\left(2^{-n}\beta^2 + 4\epsilon_0^2\right) N - 4|\epsilon_0|\alpha\sqrt{N} + \alpha^2 - \beta^2 = 0,$$

which can be solved to obtain the result. □

Note that the approximation for large $|\epsilon_0|$ gives the same data complexity as Selçuk [23]. This is due to the fact that large biases require fewer plaintext/ciphertext pairs, and for very small N the difference between sampling with and without replacement is negligible. Since Selçuk's result was obtained under Hypothesis 1 rather than Hypothesis 2, this also shows that Hypothesis 1 is a reasonable approximation when $|\epsilon_0| \gg 2^{-n/2-1}\Phi^{-1}(P_S)$.

In general, the data complexity for sampling without replacement is lower. This is a consequence of the fact that no duplicates are used. Bogdanov and Tischhauser provide an algorithm to compute the data complexity for a given success probability [9].[3] Here, the following equivalent closed-form formula for the monotonic case will be used instead:

$$N = \left(\frac{2|\epsilon_0|\beta + \sqrt{(2\epsilon_0\beta)^2 - (\alpha^2 - \beta^2)(2^{-n}\alpha^2 - 4\epsilon_0^2)}}{4\epsilon_0^2 - 2^{-n}\alpha^2}\right)^2,$$

where $\alpha = \Phi^{-1}(1 - 2^{-a-1})$ and $\beta = \Phi^{-1}(P_S)$.

Figure 4 shows the data complexity for a large bias and for a small bias close to the bound of Theorem 1 (with $n = 32$). For $|\epsilon_0| = 2^{-14}$ the difference between the data complexities is relatively small. For instance, at a success probability of 95%, the data complexity is about 14% higher for sampling with replacement. The difference with sampling without replacement is much more significant for small values of the bias. In this case, for a success probability of 95%, the data complexity is 69% higher for sampling with replacement. Note that, due to duplicates, the data complexity for sampling with replacement can exceed the size of the codebook, but not that of the key space (*i.e.*, the time complexity of handling the data cannot exceed that of brute force). For completeness and comparison with [9], we also compute the maximum of $P_S(N)$.

[3] In the non-monotonic case, their algorithm returns the lowest data complexity corresponding to the given success probability.

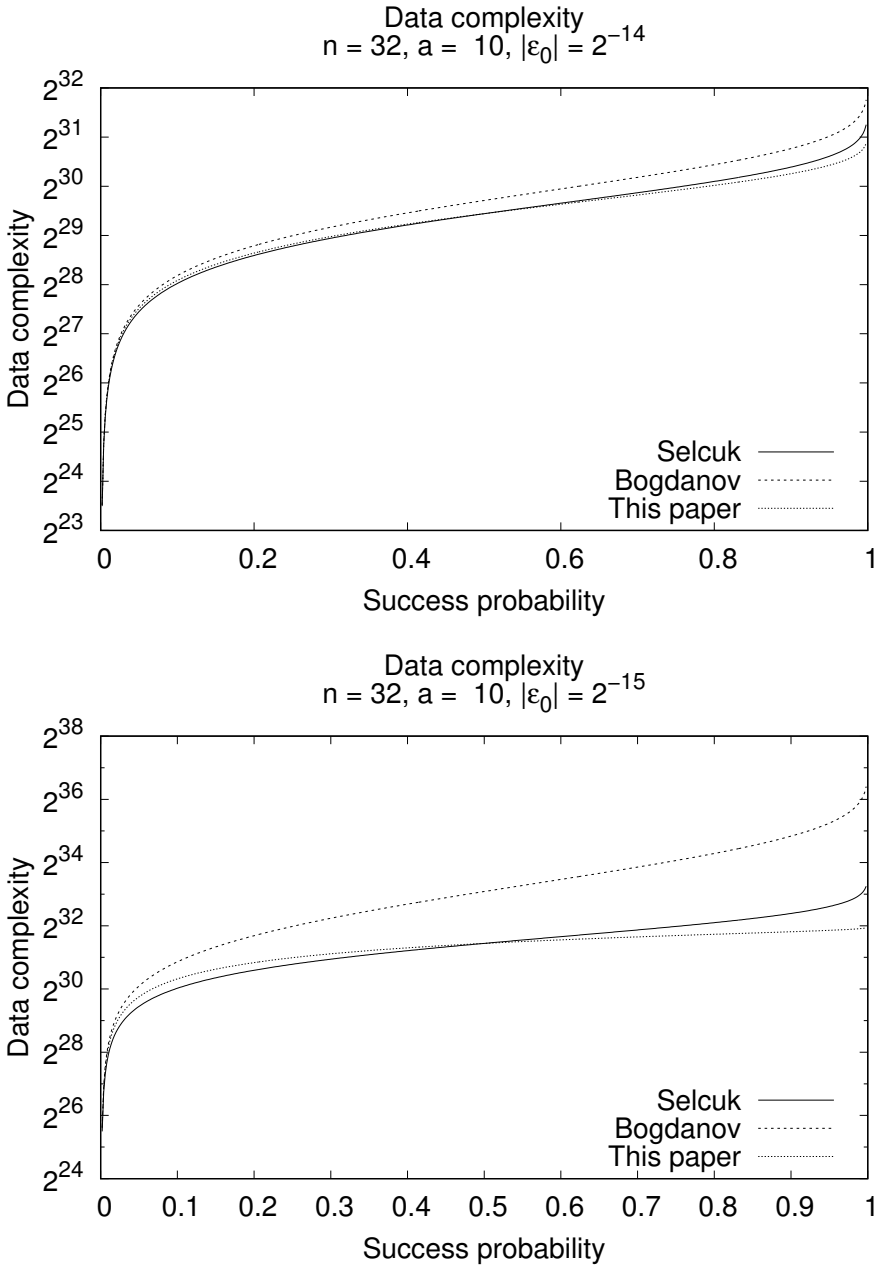

Fig. 4: The theoretical data complexity for a given success probability. The top figure corresponds to a relatively large bias compared to the bias in the bottom figure.

Corollary 1 *Under the conditions of Theorem 2 and Theorem 1, the success probability attains a maximum at*

$$\arg\max_N P_S(N) = \left(\frac{2^{n+1}|\epsilon_0|}{\Phi^{-1}(1-2^{-a-1})}\right)^2.$$

Proof. Maximizing $P_S(N)$ amounts to solving

$$\frac{d}{dN}\left(\frac{2\sqrt{N}|\epsilon_0| - \Phi^{-1}(1-2^{-a-1})}{\sqrt{1-\frac{N}{2^n}}}\right) = 0.$$

A trite calculation (*cf.* Appendix C) shows that the solution is

$$N = \left(\frac{2^{n+1}|\epsilon_0|}{\Phi^{-1}(1-2^{-a-1})}\right)^2.$$

Finally, note that the condition $N \leq 2^n$ corresponds exactly to the condition given by Theorem 1. □

6 Experimental Verification

Bogdanov *et al.* have conducted a series of large-scale experiments to verify their model for the success probability in the monotonic case [6]. Hence, Hypothesis 2 has already been verified implicitly. We use this to simplify the verification of our theoretical results by using simulations that are based on Hypothesis 2.

Since our model for the success probability relies heavily on the validity of the distribution of the sample bias $\hat{\boldsymbol{\epsilon}}_w$ for wrong keys as given by Lemma 4, we verify the accuracy of this claim in Section 6.1

In Section 6.2, we provide simulation data for the success probability and compare the results with Theorem 2.

Finally, our explanation for the non-monotonicity of the success probability given in Section 5 suggests that the probabilistic nature of the phenomenon becomes relevant when the bias is close to the bound of Theorem 1. We verify this for parameters such that the success probability is monotonic with probability 1/2.

6.1 Sample Bias

In this section we verify the proposition that, for sampling without replacement, the distribution of the sample bias is given by Lemma 4. Our experiments do not attempt to evaluate the validity of Hypothesis 2. We therefore propose the following simulation procedure:

1. Sample 2^{13} biases from $\mathcal{N}\left(0, 2^{-n-2}\right)$ and, for each such bias ϵ_i, compute the corresponding number of plaintext/ciphertext pairs for which the approximation holds, *i.e.* $R = 2^n(1/2 + \epsilon_i)$.
2. For each ϵ_i, sample N values from the corresponding population without replacement. Keep a counter T_i for the number of successful approximations. For this purpose, the simulation performs a series of dependent Bernoulli trials, *i.e.* after N samples, increasing the counter has probability $(R-T_i)/(2^n-N)$.

With respect to Lemma 4, two aspects must be verified: Firstly, the normality of $\hat{\boldsymbol{\epsilon}}_w$ and secondly the variance, which we claim is $1/(4N)$ such that the additional term 2^{-n-2} from Lemma 2 only applies to sampling with replacement. The normal probability plots in Figure 5 allow for a quick graphical check of both aspects.

From Figure 5, we see that the empirical quantiles of the observed sample biases are a linear function of the quantiles of the standard normal distribution. This indicates that the distribution of the sample bias $\hat{\boldsymbol{\epsilon}}_w$ for wrong keys is approximately normal. A Kolmogorov-Smirnov test supports this conclusion, yielding P-values of 91.68% ($n = 20$, $N = 2^{19}$) and 92.14% ($n = 32$, $N = \lfloor 2^{31.5} \rceil$). For both plots, the slope of a least-squares fit is close to $1/(2\sqrt{N})$, the standard deviation of $\hat{\boldsymbol{\epsilon}}_w$. In Figure 5, we have drawn another straight line with slope $\frac{1}{2}\sqrt{1/N + 1/2^n}$ to clarify the difference between Lemma 2 and Lemma 4.

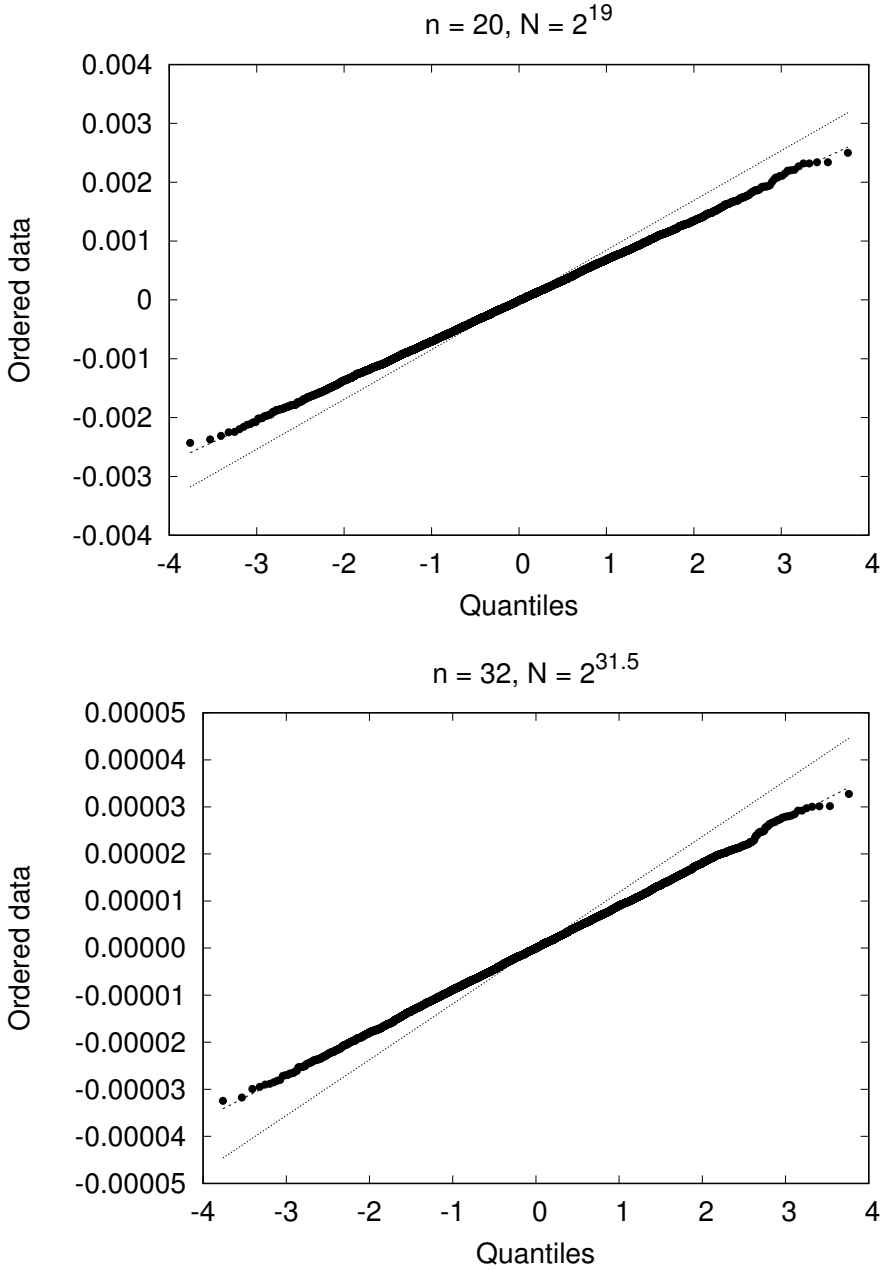

Fig. 5: Standard normal probability plots of the sample bias for sampling without replacement. Each plot contains 2^{13} data points. The straight line matching the data corresponds to the distribution for the sample bias as given by Lemma 4. The other straight line represents the result of Lemma 3.

6.2 Success Probability

In this subsection, we verify the formula for the success probability given by Theorem 2. We simulate the attack and compare our formula with the result of the simulation. This allows a simple verification of Theorem 2 independent of the accuracy of the assumptions on the sample bias. An attack is simulated as follows:

1. Generate a list of $2^m - 1$ zero-initialized counters, each corresponding to a key with associated bias sampled from $\mathcal{N}\left(0, 2^{-n-2}\right)$ in accordance with Hypothesis 2. An additional counter is kept for the right key, which has bias ϵ_0.
2. For several values of N, repeat the following steps 2^8 times to estimate the proportion of successes:
   (a) The value of each counter with corresponding bias ε is sampled directly from
   $$\mathcal{N}\left(N\left(\varepsilon + \frac{1}{2}\right), \frac{N^2}{4}\left(1 - \frac{N}{2^n}\right)\right),$$
   (b) Check whether the absolute bias corresponding to the right key is among the first 2^{m-a} entries of the list of absolute biases.

To capture the stochastic nature of the success probability, we repeat the above procedure 400 times.

The second step of the procedure can be justified by the results of the previous section, where it was shown that the distribution of the sample bias provided by Lemma 4 is sufficiently accurate. That is, the distribution from which the counters are sampled was the starting point for the proof of Lemma 4. The correspondence between the simulation and our theoretical results is shown in Figure 6.

6.3 Non-Monotonicity

To test the stochastic nature of the non-monotonicity of the success probability, we choose parameters $n, |\epsilon_0|$ and a such that Lemma 5 predicts monotonicity with some fixed probability. For $n = 32$, $m = 14$ and $a = 10$, Lemma 5 shows that $\mathbf{P_S}(N)$ is monotonic with probability $1/2$ for

$$|\epsilon_0| = 2^{-n/2-1}\Phi^{-1}\left(1 - 2^{-a-1} + 2^{-m-2}\right) \approx 2^{-15.275}.$$

Note that since m is relatively small,[4] the above value for $|\epsilon_0|$ is not exactly equal to the bound of Theorem 1 contrary to what one might expect from symmetry.

To detect whether an observation of the success probability is monotonic, it suffices to check that it reaches one for $N = 2^n$, In a total of 400 simulations, we have observed monotonic behavior in 204 cases (51%). Figure 7 shows the average curves $\mathsf{E}\left[\mathbf{P_S}(N) \mid \mathbf{P_S}(2^n) = 0\right]$ and $\mathsf{E}\left[\mathbf{P_S}(N) \mid \mathbf{P_S}(2^n) = 1\right]$, corresponding respectively to the monotonic and non-monotonic case.

[4] More precisely, the continuity correction in Lemma 5 is non-negligible.

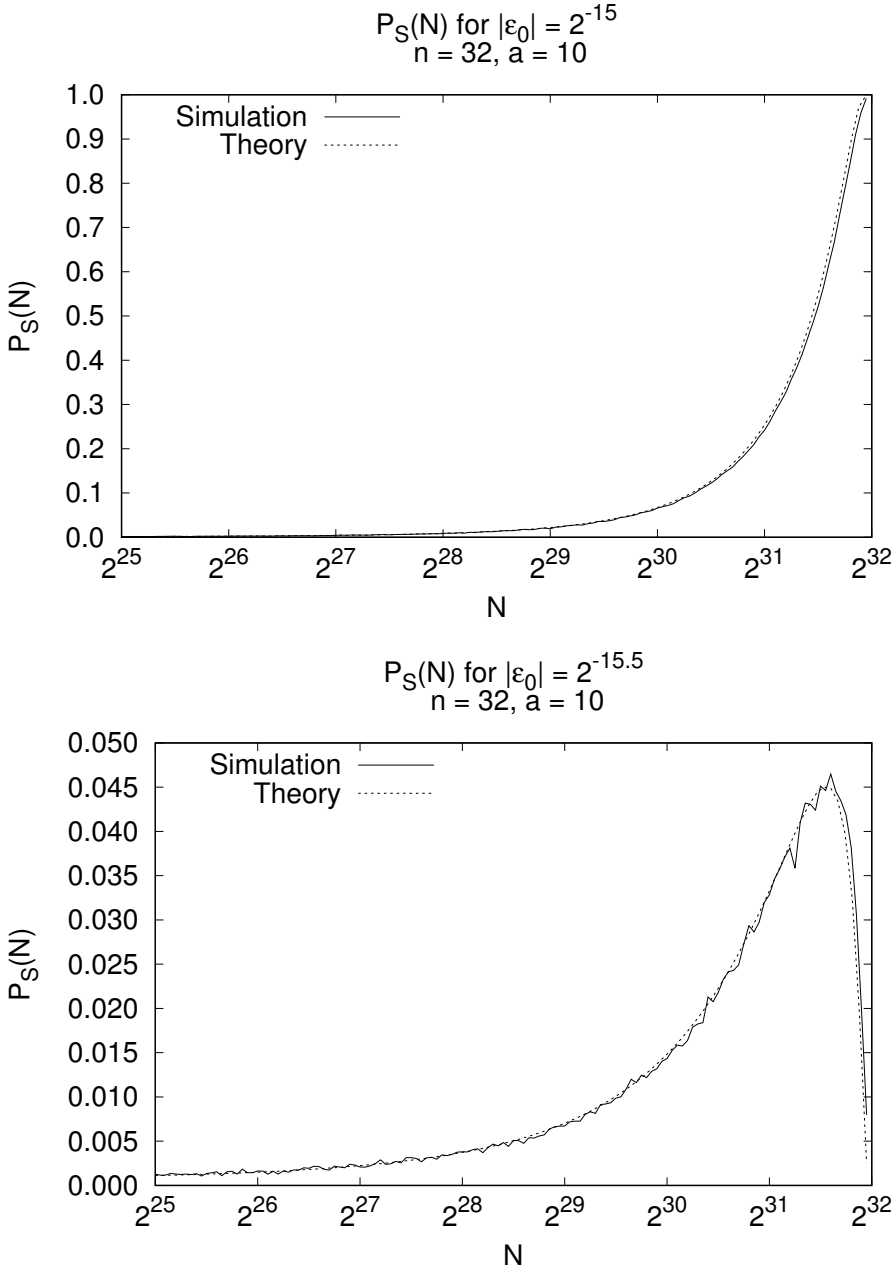

Fig. 6: Comparison of the theoretical success probability with a simulation of the success probability. The top figure on corresponds to the monotonic case, *i.e.* an attack using a bias larger than the bound of Theorem 1. The other figure shows the non-monotonic behavior.

From Corollary 1, we see that if there is a maximum success probability, it should occur for N very close to 2^n. Hence, for relatively small values of N, the average success probability should behave the same in the monotonic and non-monotonic case:

$$\mathsf{E}\left[\mathbf{P_S}(N) \mid \mathbf{P_S}(2^n) = 0\right] \approx \mathsf{E}\left[\mathbf{P_S}(N) \mid \mathbf{P_S}(2^n) = 1\right],$$

for small N. This is reflected by Figure 7.

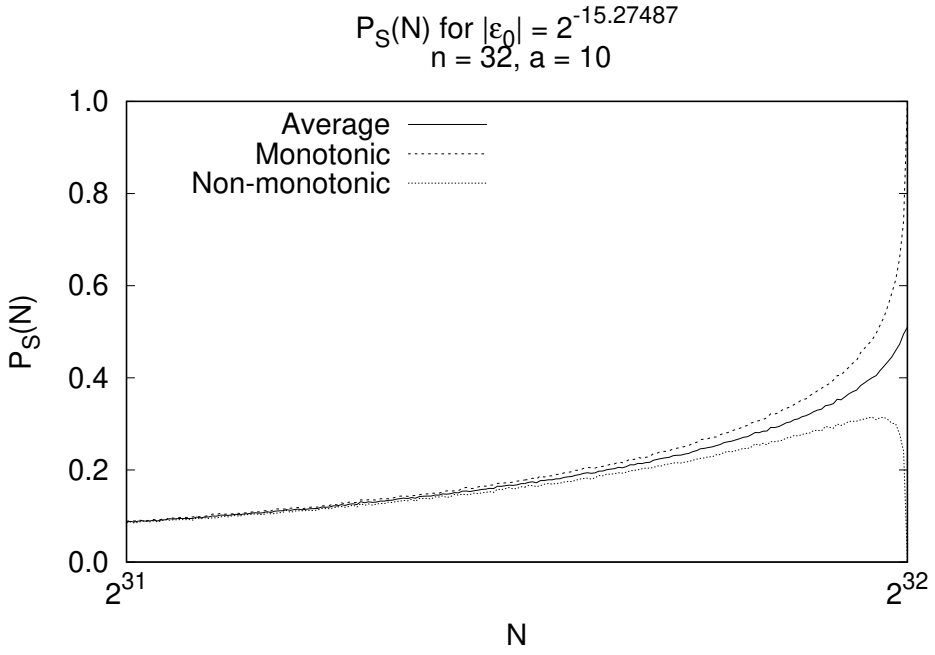

Fig. 7: Averages of the experimental success probability. The middle curve is the average over all experiments. The upper curve was computed by averaging over the experiments with monotonic behavior. Finally, the lower curve corresponds to the average over all non-monotonic experiments. Note that the scale for the horizontal axis is logarithmic and does not start from zero.

7 Discussion

To illustrate the practical consequences of some of the results above, this section contains two case studies related to the block cipher families Simon and Speck [2].

As a first example, consider a key-recovery attack based on the 15-round linear approximation of Speck-96 due to Fu *et al.* [13] with bias $|\varepsilon_0| = 2^{-46}$. The number of guessed key bits m depends on the details of the attack, but is not important for the statistical analysis as long as $a \leq m$. Table 1 lists the amount of data required by Matsui's Algorithm 2 for several values of a and with success rate $P_S = 1/2$.

Table 1: Required amount of data for linear attacks on Speck-96 with $P_S = 1/2$ for various values of a. The approximation from Fu *et al.* [13] with bias $|\varepsilon_0| = 2^{-46}$ is considered. We see that that for $a \geq 20$ the data complexity (and hence the time complexity) is beyond 2^{95} if replacement is allowed, but remains below this number if sampling is done without replacement.

a	$\log_2 N$	
	With replacement (Bogdanov *et al.* [9])	Without replacement (Theorem 3)
16	94.72	94.23
20	95.26	94.59
24	95.76	94.88

Note that, when sampling with replacement and $a \in \{20, 24\}$, the computational cost of handling the data exceeds the cost of brute-force search for Speck-96 with a 96-bit key. This situation does not occur for sampling without replacement. Hence, sampling without replacement enables attacks with a larger advantage a. Alternatively, as shown in Figure 8, a significantly higher success probability can be obtained for given values of a and N.

In the Speck-96 example above, non-monotonicity occurs (for a fixed bias) when the advantage exceeds 50.42. This provides an upper bound on the key-recovery advantage a that one can obtain using Matsui's Algorithm 2. For the approximation of Fu *et al.* [13], this bound is sufficiently large such that it is not a hindrance to realistic key-recovery attacks.

In other cases, the breakdown of Matsui's Algorithm 2 can be a serious obstacle. For example, Liu *et al.* [18] provide upper bounds on the correlation of linear trails in Simon and Simeck. For 49 rounds of Simon-96, Liu *et al.* give the upper bound $|\varepsilon_0| \leq 2^{-49}$. They conclude that Simon-96 with a 96-bit key has a margin of only three rounds with respect to linear distinguishers – which might be too small if Matsui's Algorithm 2 is accounted for. Considering that average-case non-monotonicity already occurs for $a \geq 1.66$, this conclusion seems to be too pessimistic even if the upper bound is assumed to be tight. Since only small

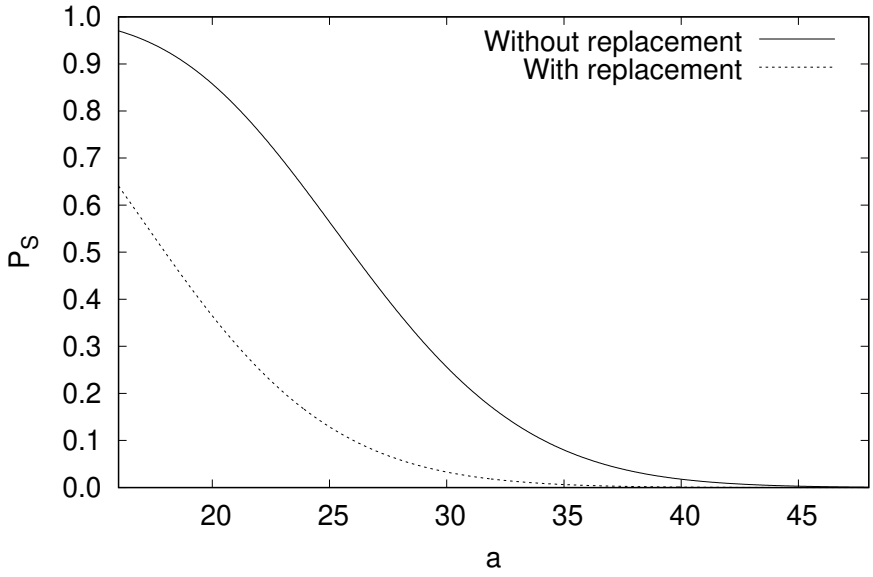

Fig. 8: The success probability for a linear attack on Speck-96 as a function of the requested advantage based on a linear approximation with bias $|\varepsilon_0| = 2^{-46}$ and $N = 2^{95}$.

advantages can be achieved, it is doubtful that one could improve over brute force even if data and memory costs are not taken into account. For instance, with $a = 1.5$, one can reduce the search space to $2^{94.5}$ candidate keys provided that $2^{95.78}$ distinct known plaintexts or $2^{98.63}$ non-distinct known plaintexts are available. Hence, the total computational cost at the very least exceeds $2^{94.5} + 2^{95.78} > 2^{96}$ if sampling is done without replacement and $2^{94.5} + 2^{98.63} > 2^{96}$ if sampling is done with replacement.

8 Conclusion

In this paper we revisited the behavior of the empirical bias for wrong keys. We have pointed out that previous works implicitly assume that plaintext/ciphertext pairs are sampled with replacement, which results in larger data complexities than necessary. We have redeveloped the theory under the assumption that the adversary can discard duplicate plaintext/ciphertext pairs, and have presented formulas for the success probability and the data complexity. The previously observed non-monotonic behavior of the success probability, which was characterized as counter-intuitive, was explained and the conditions for its occurrence were derived.

All the results in this paper have been verified through simulations. We conclude that when an adversary using Matsui's Algorithm 2 attempts to increase their advantage beyond certain bounds, what is witnessed to be an increase in the success probability is in fact a false positive. As the accuracy of the estimation increases, the attack is doomed to fail, which is evidenced by a decreasing success probability as N increases beyond a certain point.

Our simulations also show that, independent of the success probability, the attack converges faster to its final result when sampling without replacement is preferred over sampling with replacement. This results in a reduced data complexity. Since the overall time complexity of an attack includes the time it takes to generate the required data (*i.e.*, the data complexity), this reduction may decrease the overall complexity of attacks believed to be more expensive than exhaustive search.

We believe that alternatives for or extensions to Matsui's algorithm 2 could extend the reach of linear cryptanalysis to absolute biases below the bound given by Theorem 1, provided that they take into account the origins of this prerequisite.

Acknowledgments. Tomer Ashur is an FWO post-doctoral fellow under Grant Number 12ZH420N. Tim Beyne is supported by a PhD Fellowship from the Research Foundation - Flanders (FWO).

A Alternative Derivation of Lemma 4

In this section, the exact value of the mean and variance of $\mathbf{T}_w$ are computed. This leads to an alternative derivation of Lemma 4. The first two central moments of $\mathbf{R} = 2^n(\boldsymbol{\epsilon}_w + 1/2)$ are given by

$$\begin{aligned}\mathsf{E}[\mathbf{R}] &= 2^{n-1} \\ \mathsf{Var}[\mathbf{R}] &= 2^{2n}\mathsf{Var}[\hat{\epsilon}_w + 1/2] = 2^{n-2}.\end{aligned}$$

Hence, the expected value of $\mathbf{T}_w$ is given by

$$\mathsf{E}[\mathbf{T}_w] = \mathsf{E}[\mathsf{E}[\mathbf{T}_w \mid \mathbf{R}]] = \mathsf{E}\left[N\mathbf{R}\,2^{-n}\right] = N/2.$$

For the variance of T_w we have

$$\begin{aligned}\mathsf{Var}[\mathbf{T}_w] &= \mathsf{E}[\mathsf{Var}[\mathbf{T}_w \mid \mathbf{R}]] + \mathsf{Var}[\mathsf{E}[\mathbf{T}_w \mid \mathbf{R}]] \\ &= \mathsf{E}\left[N\frac{\mathbf{R}}{2^n}\frac{2^n-\mathbf{R}}{2^n}\frac{2^n-N}{2^n-1}\right] + \mathsf{Var}\left[N\frac{\mathbf{R}}{2^n}\right] \\ &= \frac{N(2^n-N)}{2^{2n}(2^n-1)}\mathsf{E}[\mathbf{R}(2^n-\mathbf{R})] + \mathsf{Var}\left[N\frac{\mathbf{R}}{2^n}\right] \\ &= \frac{N(2^n-N)}{2^{2n}(2^n-1)}\left(2^n\mathsf{E}[\mathbf{R}] - \mathsf{E}\left[\mathbf{R}^2\right]\right) + \frac{N^2}{2^{2n}}\mathsf{Var}[R] \\ &= \frac{N(2^n-N)}{2^{2n}(2^n-1)}\left(2^n\mathsf{E}[\mathbf{R}] - \mathsf{Var}[\mathbf{R}] - \mathsf{E}[\mathbf{R}]^2\right) + \frac{N^2}{2^{2n}}\mathsf{Var}[\mathbf{R}] \\ &= \frac{N(2^n-N)}{2^{2n}(2^n-1)}\left(2^{2n-2} - 2^{n-2}\right) + \frac{N^2}{2^{n+2}}.\end{aligned}$$

If n is sufficiently large, it is reasonable to assume that $2^{2n-2} - 2^{n-2} \approx 2^{2n-2}$. This gives

$$\mathsf{Var}[\mathbf{T}_w] \approx \frac{N(2^n-N)}{2^{n+2}-1/4} + \frac{N^2}{2^{n+2}} \approx \frac{N}{4}.$$

Assuming that the distribution of $\mathbf{T}_w$ can be approximated using a normal distribution, we also obtain Lemma 4.

B Data Complexity

This section provides the calculations in the proof of Theorem 3. The objective is to solve the equation

$$\Phi^{-1}(P_S)\sqrt{1-\frac{N}{2^n}} = 2\sqrt{N}|\epsilon_0| - \alpha.$$

Letting $\alpha = \Phi^{-1}(1-2^{-a-1})$ and $\beta = \Phi^{-1}(P_S)$, and squaring yields

$$\beta^2\left(1-2^{-n}N\right) = 4N|\epsilon_0|^2 - 4\sqrt{N}|\epsilon_0|\alpha + \alpha^2.$$

Grouping terms appropriately, we obtain

$$(4|\epsilon_0|^2 + 2^{-n}\beta^2)N - 4\sqrt{N}|\epsilon_0|\alpha + \alpha^2 - \beta^2 = 0.$$

This equation is quadratic in $\sqrt{N}$ and has the solutions

$$\sqrt{N} = \frac{2|\epsilon_0|\alpha \pm \sqrt{(2\epsilon_0\alpha)^2 - (\alpha^2 - \beta^2)(2^{-n}\beta^2 + 4|\epsilon_0|^2)}}{4|\epsilon_0|^2 + 2^{-n}\beta^2}.$$

C Maximum of $P_S(N)$

In the proof of Corollary 1, it is mentioned that the maximum is obtained by solving

$$\frac{d}{dN}\left(\frac{2\sqrt{N}|\epsilon_0| - \Phi^{-1}(1 - 2^{-a-1})}{\sqrt{1 - \frac{N}{2^n}}}\right) = 0.$$

Note that

$$\frac{d}{dN}\left(\frac{1}{\sqrt{1 - \frac{N}{2^n}}}\right) = \frac{1}{2^{n+1}\sqrt{\left(1 - \frac{N}{2^n}\right)^3}},$$

such that we obtain the equivalent equation

$$\frac{|\epsilon_0|}{\sqrt{N\left(1 - \frac{N}{2^n}\right)}} = \frac{\Phi^{-1}(1 - 2^{-a-1}) - 2\sqrt{N}|\epsilon_0|}{2^{n+1}\sqrt{\left(1 - \frac{N}{2^n}\right)^3}}.$$

This is readily simplified to

$$|\epsilon_0|\left(1 - \frac{N}{2^n}\right) = \sqrt{N}2^{-n-1}\Phi^{-1}(1 - 2^{-a-1}) - 2^{-n}N|\epsilon_0|,$$

and further

$$|\epsilon_0| = \sqrt{N}2^{-n-1}\Phi^{-1}(1 - 2^{-a-1}).$$

Finally, we obtain the result:

$$N = \left(\frac{|\epsilon_0|2^{n+1}}{\Phi^{-1}(1 - 2^{-a-1})}\right)^2.$$

References

1. Abdelraheem, M.A., Ågren, M., Beelen, P., Leander, G.: On the Distribution of Linear Biases: Three Instructive Examples, pp. 50–67. Springer Berlin Heidelberg, Berlin, Heidelberg (2012), http://dx.doi.org/10.1007/978-3-642-32009-5_4
2. Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., Wingers, L.: The SIMON and SPECK families of lightweight block ciphers. Cryptology ePrint Archive, Report 2013/404 (2013), https://eprint.iacr.org/2013/404
3. Biryukov, A., Cannière, C.D., Quisquater, M.: On Multiple Linear Approximations. In: Franklin, M.K. (ed.) Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3152, pp. 1–22. Springer (2004), http://dx.doi.org/10.1007/978-3-540-28628-8_1
4. Blondeau, C., Gérard, B., Tillich, J.: Accurate estimates of the data complexity and success probability for various cryptanalyses. Des. Codes Cryptography 59(1-3), 3–34 (2011), http://dx.doi.org/10.1007/s10623-010-9452-2
5. Blondeau, C., Nyberg, K.: Joint data and key distribution of simple, multiple, and multidimensional linear cryptanalysis test statistic and its impact to data complexity. Designs, Codes and Cryptography 82(1), 319–349 (2017), http://dx.doi.org/10.1007/s10623-016-0268-6
6. Bogdanov, A., Kavun, E.B., Tischhauser, E., Yalçin, T.: Large-scale high-resolution computational validation of novel complexity models in linear cryptanalysis. J. Computational Applied Mathematics 259, 592–598 (2014), http://dx.doi.org/10.1016/j.cam.2013.10.020
7. Bogdanov, A., Rijmen, V.: Zero-Correlation Linear Cryptanalysis of Block Ciphers. IACR Cryptology ePrint Archive 2011, 123 (2011), http://eprint.iacr.org/2011/123
8. Bogdanov, A., Rijmen, V.: Linear hulls with correlation zero and linear cryptanalysis of block ciphers. Des. Codes Cryptography 70(3), 369–383 (2014), http://dx.doi.org/10.1007/s10623-012-9697-z
9. Bogdanov, A., Tischhauser, E.: On the Wrong Key Randomisation and Key Equivalence Hypotheses in Matsui's Algorithm 2. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 19–38. Springer (2013), http://dx.doi.org/10.1007/978-3-662-43933-3_2
10. Bogdanov, A., Tischhauser, E., Vejre, P.S.: Multivariate Linear Cryptanalysis: The Past and Future of PRESENT. IACR Cryptology ePrint Archive 2016, 667 (2016), http://eprint.iacr.org/2016/667
11. Daemen, J., Rijmen, V.: Probability distributions of correlation and differentials in block ciphers. J. Mathematical Cryptology 1(3), 221–242 (2007), http://dx.doi.org/10.1515/JMC.2007.011
12. Feller, W.: An Introduction to Probability Theory and Its Applications, vol. 1. John Wiley & Sons (1967), exercice 10
13. Fu, K., Wang, M., Guo, Y., Sun, S., Hu, L.: Milp-based automatic search algorithms for differential and linear trails for speck. In: International Conference on Fast Software Encryption. pp. 268–288. Springer (2016)
14. Harpes, C., Kramer, G.G., Massey, J.L.: A Generalization of Linear Cryptanalysis and the Applicability of Matsui's Piling-Up Lemma. In: Advances in Cryptology - EUROCRYPT '95, International Conference on the Theory and Application of Cryptographic Techniques, Saint-Malo, France, May 21-25, 1995, Proceeding. Lecture Notes in Computer Science, vol. 921, pp. 24–38. Springer (1995)

15. Harpes, C., Massey, J.L.: Partitioning Cryptanalysis. In: Biham, E. (ed.) Fast Software Encryption, 4th International Workshop, FSE '97, Haifa, Israel, January 20-22, 1997, Proceedings. Lecture Notes in Computer Science, vol. 1267, pp. 13–27. Springer (1997), http://dx.doi.org/10.1007/BFb0052331
16. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional Linear Cryptanalysis of Reduced Round Serpent. In: Mu, Y., Susilo, W., Seberry, J. (eds.) Information Security and Privacy, 13th Australasian Conference, ACISP 2008, Wollongong, Australia, July 7-9, 2008, Proceedings. Lecture Notes in Computer Science, vol. 5107, pp. 203–215. Springer (2008), http://dx.doi.org/10.1007/978-3-540-70500-0_15
17. Junod, P., Vaudenay, S.: Optimal Key Ranking Procedures in a Statistical Cryptanalysis. In: Johansson, T. (ed.) Fast Software Encryption, 10th International Workshop, FSE 2003, Lund, Sweden, February 24-26, 2003, Revised Papers. Lecture Notes in Computer Science, vol. 2887, pp. 235–246. Springer (2003), http://dx.doi.org/10.1007/978-3-540-39887-5_18
18. Liu, Z., Li, Y., Wang, M.: The security of SIMON-like ciphers against linear cryptanalysis. Cryptology ePrint Archive, Report 2017/576 (2017), https://eprint.iacr.org/2017/576
19. Matsui, M.: Linear Cryptanalysis Method for DES Cipher. In: Helleseth, T. (ed.) Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings. Lecture Notes in Computer Science, vol. 765, pp. 386–397. Springer (1993), http://dx.doi.org/10.1007/3-540-48285-7_33
20. Molenaar, W.: Approximations to the Poisson, Binomial and Hypergeometric Distribution Functions. Ph.D. thesis, Mathematisch Centrum Amsterdam (1970)
21. Nyberg, K.: Linear Approximation of Block Ciphers. In: Santis, A.D. (ed.) Advances in Cryptology - EUROCRYPT '94, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, May 9-12, 1994, Proceedings. Lecture Notes in Computer Science, vol. 950, pp. 439–444. Springer (1994), http://dx.doi.org/10.1007/BFb0053460
22. Pinsky, M.A.: The normal approximation to the hypergeometric distribution. Unpublished manuscript, https://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/pinsky-hypergeometric.pdf
23. Selçuk, A.A.: On Probability of Success in Linear and Differential Cryptanalysis. J. Cryptology 21(1), 131–147 (2008), http://dx.doi.org/10.1007/s00145-007-9013-7