Cryptanalysis of the Alternative Mod-2/Mod-3 Weak PRF

Augustin Bariant[1], Christina Boura[2,6], Baptiste Germon[2,3], Rachelle Heim Boissier[4], Charles Meyer-Hilfiger[3], Tyge Tiessen[5]

[1] Agence Nationale de la Sécurité des Systèmes d'Information, Paris, France
augustin.bariant@ssi.gouv.fr
[2] Université Paris Cité, CNRS, IRIF, Paris, France
christina.boura@irif.fr
[3] Univ Rennes, Inria, CNRS, IRISA, Rennes, France
{baptiste.germon, charles.meyer-hilfiger}@inria.fr
[4] Université Libre de Bruxelles, Bruxelles, Belgium
rachelle.heim@ulb.be
[5] Technical University of Denmark, Kongens Lyngby, Denmark
tyti@dtu.dk
[6] Institut universitaire de France (IUF), Paris, France

Abstract. The alternative mod-2/mod-3 function is one of the most widely used weak PRF constructions in modern cryptographic protocols. Despite its practical importance, its security has received relatively limited attention, with the main cryptanalytic results consisting of two distinguishing attacks due respectively to Cheon et al. and Johansson et al. In this work, we revisit the cryptanalysis of this primitive by analyzing the output distribution of the weak PRF under fixed Hamming weights for both the secret key and the inputs. This refined analysis allows us to isolate and amplify statistical biases that were averaged out in previous works. Using this approach, we derive a new distinguishing attack with asymptotic data and time complexity $\mathcal{O}(2^{0.099n})$. We implemented the attack for the original parameter set $n = 384$, thereby obtaining the first practical attack against this instance of the construction. We then introduce a generic technique, called the splitting strategy, which consists in partially fixing or guessing part of the secret key in order to amplify the biases while introducing an additional computational cost that can be efficiently handled using Fast Fourier Transform-like techniques. This leads to the currently best known attack against the construction, with asymptotic data, time, and memory complexities $\widetilde{\mathcal{O}}(2^{0.09n})$. This last technique also provides a useful time-memory trade-off for estimating the security of real-world constructions when the available data is bounded: we show that the weak PRF offers less than 128-bit security for $n = 510$ when the data is limited to 2^{45}. Finally, we revisit the attack of Johansson et al. and provide a corrected and refined analysis of the underlying bias, showing that the statistical behavior of the attack differs significantly once the Hamming weight of the secret key is taken into account. This new analysis explains phenomena previously observed experimentally but left unexplained. Thanks to this approach we are

able to identify a large class of keys for which the attack performs much better asymptotically than anticipated by Johansson et al.

Keywords: weak PRFs · alternative mod-2/mod-3 function · statistical cryptanalysis

1 Introduction

A *weak pseudorandom function (wPRF)* is a variant of a classical pseudorandom function with a weaker notion of security, where indistinguishability from a random function is only required when the adversary is restricted to querying the function on uniformly random inputs. In particular, unlike in the standard PRF setting, the adversary cannot (adaptively) choose the inputs. Weak PRFs have important applications in modern cryptography. In particular, many constructions are designed to be shallow, meaning that they can be computed by low-depth circuits, which makes them especially attractive in contexts where evaluation cost is dominated by circuit complexity. This is notably the case in secure multi-party computation, fully homomorphic encryption and zero-knowledge proof systems, where the use of shallow primitives improves the efficiency of protocols [4,7,13]. We refer to a recent Systematization of Knowledge (SoK) article [3] for an overview of shallow weak PRFs and their applications in modern cryptographic protocols.

At the same time, these constructions differ significantly from traditional symmetric primitives: they typically consist of a single, non-iterated computation and often output only one or a few bits. Their very simple structure and low algebraic complexity therefore raise important questions regarding the extent to which they can realistically behave like random functions in practice, even within the weaker security properties expected from weak PRFs.

Among the various candidates for shallow wPRFs, a particularly popular design paradigm is the *alternating moduli construction* introduced by Boneh, Ishai, Passelègue, Sahai, and Wu in 2018 [2]. The central idea of this design is to combine simple linear functions over different moduli in order to obtain a function that is both extremely efficient to evaluate while avoiding simple low-degree representations over a single finite field. A widely used instance is the *alternative mod-2/mod-3* construction, defined for $k, x \in \mathbb{F}_2^n$ by

$$F(k,x) := (\langle k,x\rangle \bmod 2 + \langle k,x\rangle \bmod 3) \bmod 2,$$

where $\langle k,x\rangle = \sum_{i=0}^{n-1} k_i x_i$ is the integer scalar product of k and x.

The security of the alternative moduli constructions is supported in the design document by theoretical arguments relying on results by Razborov [16] and Smolensky [17]. These results show that this type of function cannot be efficiently expressed nor approximated by low-degree polynomials over a finite field. They thus suggest that these functions are likely to resist algebraic and correlation

attacks. Furthermore, the construction presents structural similarities with the Learning Parity with Noise (LPN) problem, motivating the study of BKW-type techniques in this setting. The BKW attack [1] consists in combining existing samples to produce new ones without knowledge of the secret key, exploiting the underlying linear structure in order to simplify the recovery problem. This allows solving the LPN problem with subexponential time and data complexities of $2^{\mathcal{O}(n/\log n)}$. The designers argue, however, that while this technique applies to the mod-2/mod-3 construction, it requires a very large number of samples and therefore does not constitute a practical threat.

However, the initial analysis provided by the designers was not intended to be exhaustive and offers only limited insight into the concrete security of the construction or the choice of parameters (typically, the size of n). In particular, since many applications naturally restrict the number of queries, evaluating the practical relevance of attacks against such constructions requires considering not only their time complexity but also their data complexity. The designers themselves explicitly called for further third-party cryptanalysis.

Security claims. In the original proposal, the designers of the alternative mod-2/mod-3 construction target a security level of $\lambda = 128$ bits. The initial parameter choice for the construction was $n = 384$. Following the cryptanalytic results of [6,11], later works increased the suggested parameter sizes. More precisely, the approach of [4] for constructing pseudorandom correlation functions (PCFs) targeting oblivious transfer suggests using a key and input length of $n = 770$ to achieve a security level of $\lambda = 128$ bits while taking previous attacks into account. Similarly, the framework of [7] recommends using $n = 768$ instead of 770, as it is a multiple of 32 and therefore more convenient for software implementations.

In applications where weak PRFs are used inside Multi-Party Computation (MPC) protocols, for instance to construct Oblivious Pseudorandom Functions (OPRFs), distributed Pseudorandom Functions (distributed PRFs), or Pseudorandom Correlation Functions (PCFs), the relevant notion of data complexity corresponds to the maximal number of protocol executions performed under the same secret key, which naturally corresponds to the maximal number of evaluations of the primitive that an attacker can obtain for a fixed key. While precise bounds naturally depend on the considered application and protocol, recent works based on other weak PRF candidates already consider concrete limits on the number of allowed evaluations. For instance, the QuietOT protocol [7] considers a bound of approximately 2^{33} Oblivious Transfers (OTs), corresponding to roughly 2^{33} evaluations of the underlying weak PRF. Similarly, the construction of [4] considers bounds around 2^{45} OTs. Although these works rely on a different weak PRF candidate based on Goldreich's PRG, they suggest that similar ranges of query bounds are meaningful in practice for weak-PRF-based constructions.

Previous works. To date, two works have improved upon the designers' analysis. In 2021, Cheon et al. [6] introduced a distinguishing attack with asymptotic complexity $\mathcal{O}(2^{0.21n})$. Their attack exploits a bias observable on single samples, relying on a statistical dependence between the output of the function and a

single input bit when conditioning on the corresponding key bit. Since the exploitable bias decreases exponentially with the Hamming weight of the key, the attack is more efficient against low-weight keys. Although its time complexity is below the claimed 128-bit security level for $n = 384$, its data complexity remains of the same order and is therefore incompatible with the practical query limitations typically imposed in MPC protocols.

In 2023, Johansson et al. [11] presented the currently best known cryptanalysis of this construction, leading to a distinguisher and a key-recovery attack with asymptotic complexity $\mathcal{O}(2^{0.166n})$. Their attack studies pairs of inputs at fixed Hamming distance and exploits a statistical bias appearing under additional structural conditions. A key observation is that the biased events considered in these previous works fix the Hamming weight either of the key (Cheon et al.) or of parts of the inputs (Johansson et al.) but never of both the inputs and the key at the same time.

Note that both previous attacks have time and data complexities that are exponential in the dimension, whereas, as mentioned earlier, the BKW attack has subexponential complexity. Nevertheless, for sufficiently small dimensions, these attacks outperform BKW because the constants in their exponents are small. They are therefore used to select concrete parameters.

Finally, another line of work based on an algebraic approach was proposed in [10]. For the original parameter set $\lambda = 128$ and $n = 384$, the resulting key-recovery attack has estimated time and query complexities of about $2^{149.2}$ and $2^{108.31}$ respectively. It is therefore not competitive with the previous attacks.

Our contributions. In this paper, we revisit the statistical properties of the mod-2/mod-3 weak PRF by analyzing the output distribution while fixing the Hamming weights of *both* the inputs and the key. The main insight behind our analysis is that the sign of the bias itself strongly depends on these Hamming weights: for some weights, the considered event is biased towards 0, while for others it is biased towards 1. As a consequence, previous analyses aggregate positively and negatively biased samples together, which significantly reduced the overall exploitable bias. More precisely, our contributions are as follows:

- We first introduce a new distinguishing attack based on single samples by conditioning simultaneously on the Hamming weights of the inputs and the key. This allows us to isolate significantly stronger statistical biases than in previous analyses and leads to an attack with asymptotic data and time complexity of $\mathcal{O}(2^{0.099n})$. We furthermore derive a precise concrete complexity analysis of the attack by carefully studying the exact bias values and by selecting the number of samples so that the success probability of the distinguisher is close to 1, rather than relying only on asymptotic estimates. This leads to the *first practical attack against the original parameter set* $n = 384$, with both time and data complexity equal to $2^{43.34}$, and which we successfully validated experimentally.
- We then introduce a generic improvement technique that we call the *splitting strategy*, which can be applied to the previous attack. This approach

can be viewed as a generalization of the Fast Fourier Transform (FFT)-based technique of [12], originally introduced in the context of the LPN problem to efficiently test many hypotheses simultaneously. In our setting, the idea is to partially guess or condition on part of the contribution of the secret key, which amplifies the exploitable biases while introducing an additional computational cost. We show that this cost can be efficiently handled using FFT-like techniques, allowing us to aggregate a large number of statistical contributions at the same time. By carefully balancing the gain in bias and the additional computational cost, we obtain attacks with improved trade-offs between time and data complexity. This significantly widens the range of parameters for which attacks remain feasible under realistic data constraints. For example, when $n = 510$ and the available data is bounded by 2^{45} samples, our attack achieves a time complexity of $2^{124.14}$, whereas previous attacks required at least $2^{55.7}$ samples to reach comparable time complexities below the claimed security level. On the other hand, when the amount of available data is unrestricted, this provides a slight asymptotic improvement over the attack described previously: the resulting attack has time and data complexities of $\tilde{\mathcal{O}}(2^{0.09n})$.

- Finally, we revisit the pair-based attack of Johansson et al. [11]. We show that, once the Hamming weight of the secret key is taken into account, the statistical behavior of the attack in some parameter regimes differs significantly from that predicted by the original analysis. In particular, our analysis explains several phenomena that had been observed experimentally but remained unexplained, including bias amplification for keys with high Hamming weight. Using this refined analysis, we derive exact formulas and asymptotic estimates for the underlying bias and propose several variants of the attack. We also identify a large class of keys for which the attack achieves even lower complexity.

Table 1 presents the asymptotic complexities of our different attacks. To facilitate a fair comparison with state-of-the-art attacks, these results omit polynomial and logarithmic factors. Detailed complexity analyses are provided in the corresponding sections. We also provide, when available, the concrete complexities for the parameter sets $n = 384$ and $n = 768$.

The rest of the article is organized as follows. Section 2 introduces the weak PRF under study, the mathematical tools used throughout the paper, and reviews in detail the previous cryptanalytic results. Section 3 presents our first distinguishing attack. Section 4 introduces the splitting strategy and its application to our attack. Finally, Section 5 revisits the attack of Johansson et al. and presents improved analysis and results. Our code is available at

`https://github.com/bgermon/Cryptanalysis-Mod-2-Mod-3`.

Table 1. Comparison of the best known attacks against the mod-2/mod-3 weak PRF. Time complexities are given in number of operations on n-bit words. All attacks are distinguishing attacks unless stated otherwise. The attack of Section 4 is parametrized to minimize the time complexity, regardless of the data complexity. The attack marked with * is a weak-key attack, while the attack marked with † is a key-recovery attack. For the attacks marked with ‡, we assume that the attacker knows the Hamming weight of the key. In Section 3.2, we also provide the corresponding complexities without this assumption. For the attack of Johansson et al. [11], the concrete complexities are those reported by the authors for their optimized choice of parameters minimizing the asymptotic time complexity, while ignoring polynomial factors. The attacks of [10] were specifically tailored to the parameter sets $(n, \lambda) = (384, 128)$ and $(768, 256)$ respectively, and therefore do not admit a direct asymptotic interpretation.

Attack	Asymptotic		$n = 384$		$n = 768$	
	Time	**Data**	**Time**	**Data**	**Time**	**Data**
[6]‡	$\mathcal{O}(2^{0.21n})$	$\mathcal{O}(2^{0.21n})$	$2^{80.6}$	$2^{80.6}$	$2^{161.3}$	$2^{161.3}$
[11]	$\mathcal{O}(2^{0.166n})$	$\mathcal{O}(2^{0.1n})$	$2^{74.27}$	$2^{44.5}$	$2^{138.8}$	$2^{83.62}$
[10]†	—	—	$2^{149.2}$	$2^{108.31}$	$2^{207.9}$	$2^{206.31}$
Section 3.2‡	$\mathcal{O}(2^{0.099n})$	$\mathcal{O}(2^{0.099n})$	$2^{43.34}$	$2^{43.34}$	$2^{81.76}$	$2^{81.76}$
Section 4	$\tilde{\mathcal{O}}(2^{0.09n})$	$\tilde{\mathcal{O}}(2^{0.09n})$	$2^{40.87}$	$2^{40.67}$	$2^{75.93}$	$2^{75.85}$
Section 5.4*	$\mathcal{O}(2^{0.147n})$	$\mathcal{O}(2^{0.08n})$	—	—	—	—

2 Preliminaries

We begin this section by introducing the weak PRF under study, together with the notation and probabilistic tools used throughout the paper. We then review the previous cryptanalytic results on this construction.

2.1 Weak pseudo-random functions

A weak pseudo-random function (wPRF) is a function that cannot be distinguished from random when its inputs are drawn at random. A more rigorous security definition inspired by [3] is the following.

Definition 1 (wPRF security). *Let n, q, t, λ be positive integers, and let $F : \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2$ be a function. The family of functions $(F_k := F(k, \cdot))_{k \in \mathbb{F}_2^n}$ is said to achieve (q, t, λ)-weak PRF security if, for every adversary $\mathcal{A}$ running in time at most t and making at most q queries on independent and uniformly random inputs $\boldsymbol{x} = (x^{(1)}, \ldots, x^{(q)}) \in \mathbb{F}_2^n$, the advantage $\mathbf{Adv}(\mathcal{A})$ of the adversary $\mathcal{A}$ in distinguishing F from random satisfies*

$$\mathbf{Adv}(\mathcal{A}) := \left| \mathbb{P}\left(\mathcal{A}(\boldsymbol{x}; y_1, \ldots, y_q) = 1\right) - \mathbb{P}\left(\mathcal{A}(\boldsymbol{x}; r_1, \ldots, r_q) = 1\right) \right| \leq \frac{t}{2^\lambda},$$

where, for every $i \in \{1, \ldots, q\}$, we have $y_i = F(k, x^{(i)})$, with k drawn uniformly at random from $\mathbb{F}_2^n$, and where the r_i are sampled independently and uniformly at random from $\mathbb{F}_2$.

The parameter λ can be viewed as a security parameter: any distinguishing attack with advantage p must have time complexity at least $p2^{\lambda}$.

In this paper, we study the weak-PRF security of the mod-2/mod-3 alternative moduli function F defined in the sequel.

2.2 The mod-2/mod-3 function

The alternative mod-2/mod-3 construction is a function $F : \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2$ defined by

$$\begin{aligned} F(k, x) &\coloneqq (\langle k, x \rangle \bmod 2 + \langle k, x \rangle \bmod 3) \bmod 2 \\ &= H\big(\langle k, x \rangle\big) \end{aligned}$$

where $H : \mathbb{Z} \to \mathbb{F}_2$ is defined by

$$H(w) = \begin{cases} 0 & \text{if } w \bmod 6 \in \{0, 1, 2\}, \\ 1 & \text{otherwise.} \end{cases}$$

Note that, following the terminology introduced in [8], this construction is also often referred to as *LPN-wPRF*, since it can be interpreted as a variant of the Learning Parity with Noise (LPN) problem in which the noise depends on both the secret key and the input.

2.3 Previous works

In this section, we describe in more detail the attacks of Cheon et al. [6] and Johansson et al. [11].

The attack of Cheon et al. [6] Cheon et al. proposed a simple statistical distinguisher against the alternative mod-2/mod-3 function. Their attack exploits a bias in the conditional distribution of individual input bits when conditioning on the output of the function. More precisely, let $k \in \mathbb{F}_2^n$ be a secret key of Hamming weight βn, where $\beta \in (0, 1)$ and such that βn is an integer. Cheon et al. prove that there exists an index $j \in \{0, \ldots, n-1\}$ such that

$$\mathbb{P}\left(x_j = 0 \mid k_j = 1 \text{ and } F_k(x) = 0\right) = \frac{1}{2} + \varepsilon,$$

where the bias satisfies $|\varepsilon| \approx 2^{-0.21\beta n}$. For a random key, the expected value of β is $1/2$, which leads to a bias of order $2^{-0.105n}$. The distinguishing attack then consists in collecting samples $(x^{(i)}, y_i)$ and restricting to those for which $y_i = 0$. For each coordinate $j \in \{0, \ldots, n-1\}$, the empirical probability

$\Pr[x_j = 0 \mid F_k(x) = 0]$ is then estimated. If there exists a coordinate j for which this empirical probability significantly deviates from $1/2$, the attack concludes that the samples were computed using F, that is, $y_i = F(k, x^{(i)})$. Otherwise, it concludes that the y_i were sampled uniformly at random. Since detecting a bias of magnitude ε requires $\mathcal{O}(\varepsilon^{-2})$ samples, this leads to a distinguishing attack with data and time complexity $\mathcal{O}(2^{0.21n})$.

The attack of Johansson et al. [11] Johansson, Meier and Nguyen [11] introduced in 2023 a distinguishing and key-recovery attack against the same wPRF candidate. Their approach exploits statistical properties on pairs of inputs rather than on single samples, significantly improving both the data and time complexities compared to Cheon et al.'s attack.

Attack strategy. Let $x, x' \in \mathbb{F}_2^n$ be two uniformly random inputs and let $\alpha n = \mathrm{wt}(x \oplus x')$ denote the Hamming weight of their difference. Writing

$$I = \{i \in \{0, .., n-1\} : (x_i, x'_i) = (1, 0)\},\ J = \{i \in \{0, .., n-1\} : (x_i, x'_i) = (0, 1)\},$$

we have $|I| + |J| = \alpha n$.

The attack then exploits pairs (x, x') such that αn is even and $|I| = |J| = \alpha n/2$. For those pairs, the event $\{F(k, x) = 0\}$ is highly dependent on the event $\{F(k, x') = 0\}$. More precisely, they claim that

$$\mathbb{P}\left(F(k, x) = 0 \mid x_0 = x'_0, F(k, x') = 0\right) = \frac{1}{2} + \gamma,$$

where the bias γ satisfies $|\gamma| \approx 2^{-0.41\frac{\alpha n}{2}+1.17}$ and is thus non-negligible. The condition $x_0 = x'_0$ was introduced by Johansson et al. mainly for key recovery purposes in order to distinguish between the cases $k_0 = 0$ and $k_0 = 1$. However, as detailed in Section 5.5, our experiments suggest that converting the distinguisher into a key-recovery attack using this approach requires significantly more data than distinguishing alone.

According to their result, the bias decreases exponentially as the Hamming weight αn increases. The intuition behind the attack is that, given two inputs x, x' satisfying the conditions described above, the corresponding scalar products can be decomposed as follows:

$$\begin{aligned} \langle x, k\rangle &= \langle x, k\rangle_C + \langle x, k\rangle_I \\ &= s + \langle x, k\rangle_I \end{aligned} \qquad \begin{aligned} \langle x', k\rangle &= \langle x', k\rangle_C + \langle x', k\rangle_J \\ &= s + \langle x', k\rangle_J \end{aligned}$$

where C denotes the set of indices corresponding to the common bits of x and x'. Since both I and J have cardinality $\alpha n/2$, when αn is small the contributions $\langle x, k\rangle_I$ and $\langle x', k\rangle_J$ are also small. As a consequence, the dominant contribution in both scalar products comes from the common part $\langle x, k\rangle_C = \langle x', k\rangle_C$, which implies that $\langle x, k\rangle$ and $\langle x', k\rangle$ are strongly correlated. This explains why the values $F(k, x)$ and $F(k, x')$ are themselves strongly related when αn is small. The main difficulty is therefore to efficiently find pairs of inputs at Hamming

distance αn for relatively small values of αn. Johansson et al. reduce this problem to a nearest-neighbor search problem and solve it using the algorithm of May and Ozerov [14]. A detailed analysis of this algorithm and its complexity can be found in [5]. Combining the cost of producing these pairs with the number of pairs required to detect the bias leads to a trade-off between both terms. Optimizing over αn, they obtain an overall time complexity of $\mathcal{O}(2^{0.166n})$.

2.4 Probability background, notations

In this subsection, we introduce the probabilistic and statistical tools used throughout the paper, starting with some useful notations.

- $\mathcal{B}(n,p)$ denotes the *binomial distribution* with parameters $n \in \mathbb{N}_{>0}$ and $p \in (0,1)$. It models the number of successes obtained in n independent *Bernoulli* trials with success probability p. In the special case of a single trial, we simply write $\mathcal{B}(p)$.
- $\mathcal{N}(\mu,\sigma^2)$ denotes the *normal (Gaussian) distribution* with mean $\mu \in \mathbb{R}$ and variance $\sigma^2 > 0$.
- $\mathcal{H}(n,m,s)$ denotes the *Hypergeometric distribution* with parameters $n \in \mathbb{N}_{>0}$ (population size), $m \in \{0,\ldots,n\}$ (number of marked elements), and $s \in \{0,\ldots,n\}$ (number of draws). It models the number of marked elements obtained when drawing s elements uniformly at random without replacement from a population of size n.

For any random variable X, we denote its expected value by $\mathbb{E}(X)$, its variance by $\mathbb{V}(X)$ and its standard deviation by $\mathbb{S}(X)$.

In the sequel, we will frequently compute statistics on the data returned by the oracle. Such statistics behave differently depending on whether the oracle outputs uniformly random bits or computes them using the primitive. We will therefore model each statistic as a random variable and use the notations $\mathbb{E}_\$(X)$ (resp. $\mathbb{V}_\$(X)$, $\mathbb{S}_\$(X)$) in the random case, and $\mathbb{E}_F(X)$ (resp. $\mathbb{V}_F(X)$, $\mathbb{S}_F(X)$) when the oracle computes its outputs using the primitive. When we consider a statistic computed from observed data and, in particular, when the underlying distribution is unknown, we simply write $\mathbb{E}(X)$ (resp. $\mathbb{V}(X)$, $\mathbb{S}(X)$).

Hoeffding's inequality. [9] We will frequently use the following concentration bound throughout the paper. Let $X_1,\ldots,X_N$ be independent random variables such that $a_i \le X_i \le b_i$ for all $i \in \{1,\ldots,N\}$, and let $X \coloneqq \sum_{i=1}^{N} X_i$. Then, for any $T \ge 0$, Hoeffding's inequality states that

$$\mathbb{P}\left(X - \mathbb{E}(X) \ge T\right) \le \exp\left(-\frac{2T^2}{\sum_{i=1}^{N}(b_i-a_i)^2}\right),$$
$$\mathbb{P}\left(|X - \mathbb{E}(X)| \ge T\right) \le 2\exp\left(-\frac{2T^2}{\sum_{i=1}^{N}(b_i-a_i)^2}\right).$$

Likelihood. Since our attacks are statistical, we will often study the *likelihood* of an observed variable with respect to a given distribution. The likelihood, as suggested by its name, allows to capture how likely it is that an observed data y has been generated following a given distribution. We give a definition of the likelihood in the particular case of independent Bernoulli distributions, as this is the only case we will consider in our attacks. More precisely, let $N \in \mathbb{N}_{>0}$, and let $Y_1, \ldots, Y_N$ be independent random variables such that, for all $1 \leq i \leq N$,

$$Y_i \sim \mathcal{B}_i \coloneqq \mathcal{B}(1/2 + \varepsilon_i)\,,$$

for some $\varepsilon_i \in [-1/2, 1/2]$. We define $\mathcal{D} \coloneqq \otimes_{i=1}^{N} \mathcal{B}_i$ so that $(Y_1, \ldots, Y_N) \sim \mathcal{D}$.

Definition 2 (Likelihood). *Let* $\mathbf{y} = (y_1, \ldots, y_N) \in \mathbb{F}_2^N$. *The likelihood of* $\mathbf{y}$ *with respect to* $\mathcal{D}$ *is defined as*

$$L_{\mathcal{D}}(\mathbf{y}) \coloneqq \mathbb{P}\left(\bigcap_{i=1}^{N} Y_i = y_i\right).$$

A standard approach is to consider the log-likelihood, that is, the logarithm of the likelihood. In the following, we will use the quantity below, which we show to approximate the log-likelihood up to an additive and multiplicative constant.

Definition 3. *The* score *of* $\mathbf{y} = (y_1, \ldots, y_N) \in \mathbb{F}_2^N$ *with respect to the distribution* $\mathcal{D}$ *is defined by*

$$R_{\mathcal{D}}(\mathbf{y}) \coloneqq \sum_{i=1}^{N} (-1)^{y_i} \varepsilon_i.$$

This score can be computed efficiently on the fly, which will play a central role in the *splitting strategy* introduced in Section 4.

Remark 1. As mentioned above, the score is approximately equal, up to an additive and multiplicative constant, to the log-likelihood. Since the Y_i's are independent, we have

$$\begin{aligned} L_{\mathcal{D}}(\mathbf{y}) &= \prod_{i=1}^{N} \mathbb{P}\left(Y_i = y_i\right) = \prod_{i=1}^{N} \left(y_i \left(\frac{1}{2} + \varepsilon_i\right) + (1 - y_i)\left(\frac{1}{2} - \varepsilon_i\right)\right) \\ &= \frac{1}{2^N} \prod_{i=1}^{N} \left(1 - 2(-1)^{y_i} \varepsilon_i\right). \end{aligned}$$

Using the classical approximation $\ln(1 + x) \sim x$ as $x \to 0$, we obtain $\ln\left(1 - 2(-1)^{y_i}\varepsilon_i\right) \sim -2(-1)^{y_i}\varepsilon_i$ when $\varepsilon_i \to 0$. Therefore,

$$\log_2\big(L_{\mathcal{D}}(\mathbf{y})\big) \approx -N - \frac{2}{\log(2)} \sum_{i=1}^{N} (-1)^{y_i} \varepsilon_i.$$

Remark 2. The score defined above can also be viewed as approximately proportional to the log-likelihood *ratio*, where the second distribution is the uniform distribution over $\mathbb{F}_2^N$. Intuitively, considering the likelihood ratio removes the additive constant term $-N$. In our context, this is particularly relevant since our attacks aim at distinguishing a given distribution from the uniform distribution $\mathcal{D}_\$$. Although the Neyman–Pearson lemma [15] does not apply directly to our score function, its guarantee of the optimality of likelihood-ratio tests provides strong theoretical motivation for our analysis.

3 A first distinguishing attack

We now introduce our first distinguishing attack against the alternative mod-2/mod-3 weak PRF. In the following, we let $0 \leq \alpha, \beta \leq 1$ be rational numbers such that αn and βn are integers. The next proposition captures the fact that the mod-2/mod-3 primitive is biased when restricting the inputs and the key to specific weights.

Lemma 1. *Let $k \in \mathbb{F}_2^n$ be such that $\mathrm{wt}(k) = \beta n$. Then:*

$$\mathbb{P}_x \left(F(k,x) = 1 \mid \mathrm{wt}(x) = \alpha n \right) = \frac{1}{2} + \epsilon(\alpha, \beta, n)$$

where

$$\epsilon(\alpha,\beta,n) = \binom{n}{\alpha n}^{-1} \sum_{\substack{i=0 \\ i \bmod 6 \in \{3,4,5\}}}^{\beta n} \binom{\beta n}{i} \binom{n-\beta n}{\alpha n - i} - \frac{1}{2}\,. \tag{1}$$

Proof. Fix a key k of Hamming weight βn, and let x be uniformly distributed among the vectors of weight αn. Since $F(k,x) = H(\langle x, k \rangle)$, we have $F(k,x) = 1$ if and only if $\langle x, k \rangle \bmod 6 \in \{3,4,5\}$.

For a fixed i, the number of x such that $\mathrm{wt}(x) = \alpha n$ and $\mathrm{wt}(k \odot x) = i$ is

$$\binom{\beta n}{i} \binom{n-\beta n}{\alpha n - i}.$$

Indeed, one has to choose i positions of x on the support of k, and $\alpha n - i$ positions outside the support of k.

Therefore,

$$\mathbb{P}_x \left(F(k,x) = 1 \mid \mathrm{wt}(x) = \alpha n \right) = \binom{n}{\alpha n}^{-1} \sum_{\substack{i=0 \\ i \bmod 6 \in \{3,4,5\}}}^{\beta n} \binom{\beta n}{i} \binom{n-\beta n}{\alpha n - i}.$$

Writing this probability as $1/2 + \epsilon(\alpha, \beta, n)$ gives the claimed expression. □

Remark 3. The expression of $\epsilon(\alpha, \beta, n)$ shows that the bias is symmetric in α and β. The bias is also largest when one or both of the two quantities α and

β significantly deviate from 1/2. Indeed, we are essentially considering a conditioned hypergeometric distribution. For values of α or β close to 0 or 1, the Hypergeometric distribution is concentrated in a smaller interval, leading to a more unbalanced distribution across residue classes modulo 6.

The expression in Eq. (1) gives the exact value of the bias and can be computed efficiently even for very large values of n. It is therefore the expression we use in practice to estimate the complexity of the attack we present later in this section. Exact bias values for selected values of n, α, and β are given in Table 1 of Appendix A.

For the purpose of asymptotic analysis, however, it is useful to derive an explicit analytic approximation of this bias that captures its main behavior as n grows. The following lemma provides such an approximation. Its proof uses the Fourier expansion of an indicator function to express the bias ϵ as a sum of values of the characteristic function of $\mathcal{H}(n, \beta n, \alpha n)$. To complete the proof, we assume that this distribution is asymptotically Gaussian. This assumption is stated explicitly in the proof in Appendix B, and Lemma 2 holds only under this assumption.

Lemma 2 (Asymptotic estimate of the bias). *Let $0 < \alpha, \beta < 1$ be fixed rational numbers such that αn and βn are integers. Let $\mu \coloneqq \alpha\beta$ and $\sigma^2 \coloneqq \alpha\beta(1-\alpha)(1-\beta)$. Under Assumption 1, the bias $\epsilon(\alpha, \beta, n)$ satisfies*

$$\epsilon(\alpha,\beta,n) \underset{n\to+\infty}{\sim} \frac{2}{3}\exp\left(-\frac{\pi^2\sigma^2}{18}n\right)\cos\left(\frac{\pi}{3}\mu n+\frac{2\pi}{3}\right)+\mathcal{O}\left(\exp\left(\frac{-\pi^2\sigma^2}{2}n\right)\right).$$

In the following, we do not consider the last term as it is negligible compared to the first one. Lemma 2 above shows more clearly that the function is increasingly biased as the Hamming weight of the input or the key deviates significantly from 1/2.

To evaluate the accuracy of the asymptotic estimate from Lemma 2, we experimentally compared it with the exact bias formula given in Lemma 1. More precisely, we computed both the exact bias and its asymptotic approximation for values of α and β in $\{0.3, 0.4, 0.5, 0.6, 0.7\}$, and for all values of n in the interval $[1000, 10000]$ with step size 100. Figure 1 only displays the representative case $\alpha = \beta = 1/2$, but the same behavior was observed for all tested parameter choices. As shown in Figure 1, the asymptotic approximation closely matches the exact bias.

We now show how to use Lemma 1 to mount a distinguishing attack against the alternative mod-2/mod-3 weak PRF.

Remark 4. When $\beta = 1/2$, we obtain $\epsilon(\alpha, 1/2, n) = \mathcal{O}\left(2^{-\log_2(e)\cdot b(\alpha)\cdot n}\right)$ where $b(\alpha) = \pi^2\alpha(1-\alpha)/72$. Note that $\log_2(e)\cdot b(\alpha) \approx 0.198\alpha(1-\alpha)$. By further letting $\alpha = 1/2$, the bias becomes $\epsilon\left(1/2, 1/2, n\right) = \mathcal{O}\left(2^{-0.0495n}\right)$.

Comparison with Cheon et al. Asymptotically, Cheon et al. [6] obtain a bias decreasing in $2^{-0.105n}$, while our analysis shows a slower decrease in $2^{-0.0495n}$

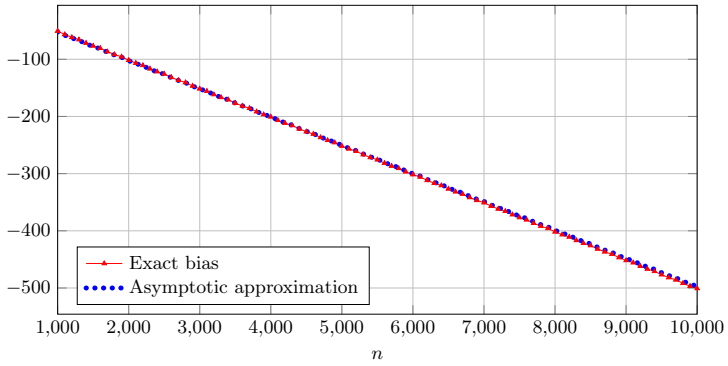


Fig. 1. Comparison between the exact bias $\epsilon(1/2, 1/2, n)$ given by Lemma 1 and its asymptotic approximation from Lemma 2, for $n \in \{1000, 1100, 1200, \ldots, 10000\}$. The figure plots $\log_2 |\epsilon(1/2, 1/2, n)|$ and shows that the asymptotic estimate accurately captures the exponential decrease rate of the exact bias over the whole tested range.

for $\alpha = \beta = 1/2$. In other words, the logarithm of the bias is roughly twice as large in our attack, which translates into a quadratic improvement in the complexity of the corresponding distinguishing attack. Figure 2 illustrates the main difference between our analysis and that of Cheon et al. More precisely, the figure displays the exact bias $\epsilon(\alpha, \beta, n)$ as a function of the Hamming weight αn of the input, for fixed parameters n and β. One can observe that the sign of the bias strongly depends on the value of α: for some input weights, the function is biased towards 1, resulting in a positive bias, whereas for other weights it is biased towards 0, resulting in a negative bias. This oscillating behavior is a consequence of the modulo-6 structure of the function H and is accurately captured by the cosine term appearing in Lemma 2. In our approach, we condition on the Hamming weight of the input and therefore isolate these individual biases. In contrast, the attack of Cheon et al. [6] does not condition on $\mathrm{wt}(x)$ and instead computes a global bias over the full input space. As a consequence, their analysis likely averages together positively and negatively biased contributions coming from different Hamming weights. These contributions partially cancel each other, which significantly reduces the magnitude of the overall exploitable bias. This explains why the bias obtained in our analysis is substantially larger than the one exploited by Cheon et al.

3.1 A simple distinguishing attack based on Lemma 1

Let us fix n and α, and denote β the Hamming weight of the key. Since α, β, and n are fixed, the bias $\epsilon(\alpha, \beta, n)$ is also fixed, and we simply denote it by ε. A standard distinguishing attack can be mounted using $\tilde{\mathcal{O}}(\varepsilon^{-2})$ samples.

Algorithm. A pseudo-code for our attack is given in Algorithm 1. The adversary makes N oracle calls and stores in a list L the samples $(x^{(i)}, y_i)$

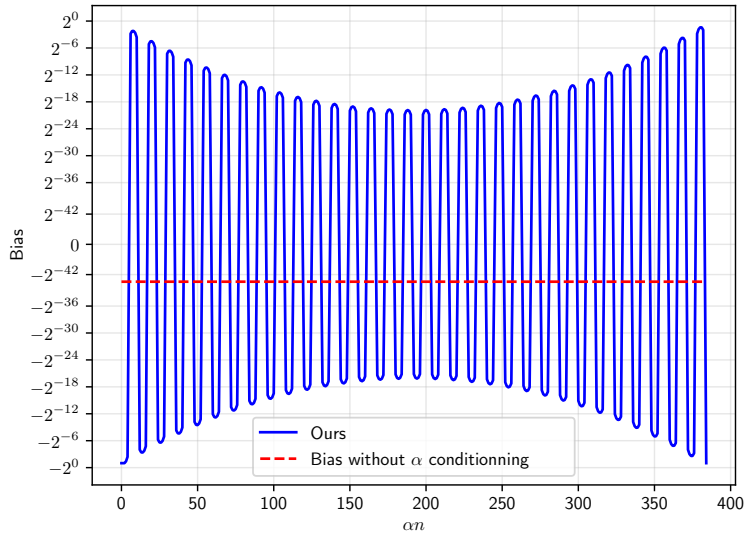


Fig. 2. Comparison of our bias computed with the exact formula (1) to the bias that we would obtain without conditioning on αn, for $n = 384$ and $\beta = 1/2$.

such that $\mathrm{wt}(x^{(i)}) = \alpha n$. It then evaluates how likely the observed outputs $\mathbf{y} \coloneqq (y_i)_{(x^{(i)},y_i)\in L}$ are under the hypothesis that they are independent samples from $\mathcal{B}(1/2+\varepsilon)$. As defined in Definition 3,[7] this likelihood can be approximated using the score

$$R(\mathbf{y}) \coloneqq \sum_{(x,y)\in L} (-1)^y \varepsilon\,.$$

In the random and weak-PRF cases, respectively, we have

$$\mathbb{E}_{\$}\left(R\right) = 0 \qquad \text{and} \qquad \mathbb{E}_{F}\left(R\right) = -2|L|\varepsilon^2.$$

The algorithm therefore decides that the samples are random if $|R(\mathbf{y})| < T$, where the threshold is chosen halfway between these two expectations:

$$T \coloneqq |\mathbb{E}_F(R) - \mathbb{E}_{\$}(R)|/2 = |L|\varepsilon^2.$$

[7] Here, ε is constant, so the statistic could be simplified. We keep this notation for consistency with the rest of the paper.

Algorithm 1 Basic attack

Parameters: N, α, β.
Assumption: In the case where the oracle is a weak PRF then the key is a vector of $\mathbb{F}_2^n$ of Hamming weight βn.
Initialize an empty list L
for all $i \in [\![1, N]\!]$ **do**
    Query the oracle, get a pair (x, y).
    if $\mathrm{wt}(x) = \alpha n$ **then**
        Add (x, y) to the list L
$R \leftarrow \sum_{(x,y)\in L} (-1)^y \varepsilon$
$T \leftarrow |L|\, \varepsilon^2$
if $|R| < T$ **then**
    return "Random".
return "weak PRF".

Remark 5. For simplicity, we describe the attack using an explicitly stored list of samples. However, the statistic $R(\mathbf{y})$ can be computed on-the-fly as the samples are obtained. Therefore, the attack can be implemented using negligible memory. This remark also applies to the attack we present in Section 3.2.

Analysis. We give a simplified heuristic analysis by assuming that the size of the list L is equal to its expected value. Namely, we assume that

$$|L| = Np_\alpha, \qquad p_\alpha \coloneqq 2^{-n} \cdot \binom{n}{\alpha n},$$

where p_α is the probability that a uniformly random $x \in \mathbb{F}_2^n$ has Hamming weight αn. Recall that the threshold is $T = |L|\varepsilon^2 = |\mathbb{E}_F(R)|/2$. By using the Hoeffding bound (see Section 2.4),[8] we obtain

$$\begin{aligned}\mathbb{P}_F\left(|R| < T\right) &= \mathbb{P}_F\left(|R| - |\mathbb{E}_F(R)| < -T\right) \leq \mathbb{P}_F\left(||R| - |\mathbb{E}_F(R)|| > T\right) \\ &\leq \mathbb{P}_F\left(|R - \mathbb{E}_F(R)| \geq T\right) \leq 2\exp\left(-T/2\right) .\end{aligned}$$

and

$$\mathbb{P}_\$\left(|R| < T\right) = 1 - \mathbb{P}_\$\left(|R| \geq T\right) \geq 1 - 2\exp\left(-T/2\right) .$$

Taking $T = 7.38$, equivalently $|L|\varepsilon^2 = 7.38$, we obtain that the advantage of our algorithm is given by

$$\left|\mathbb{P}_\$\left(|R| < T\right) - \mathbb{P}_F\left(|R| < T\right)\right| \geq 1 - 4\exp\left(-7.38/2\right) \approx 0.9.$$

This shows that, if N satisfies

$$N = \frac{7.38}{\epsilon(\alpha, \beta, n)^2} \cdot \frac{2^n}{\binom{n}{\alpha n}}.$$

[8] We also use the fact that $|\varepsilon| < 1/2$ implies $2T^2/N < T/2$.

then Algorithm 1 distinguishes with advantage greater than 0.9 in

$$\mathbf{Data} = N, \quad \mathbf{Time} = \mathcal{O}(N), \quad \mathbf{Memory} = \mathcal{O}(1).$$

where the time complexity is given in number of operations on n-bit words.

Remark 6. Whilst we provide the time complexity using the Landau notation for rigor, we expect the practical complexity to be very close to N operations on n-bit words (there is no significant hidden constant). This is also what our experiments suggest (see the end of the section).

Remark 7. Note that there is no extra complexity factor for guessing the value of β. For any n and α, we observed that the exponential factor in the bound of Lemma 2 reaches its minimum non-zero value for $\beta = 1/2$. For $\beta \neq 1/2$, there is a probability greater than $1/2$ that the cosinus term is non-zero,[9] in which case the attack still distinguishes correctly. To mitigate the cases where the cosinus term is zero, the attack can simply be repeated with other values of α. Only a (small) constant number of iterations is enough.

Asymptotic analysis. Asymptotically, our distinguishing attack thus has the following data complexity and time complexity (given in number of operations on n-bit words)

$$N = \mathcal{O}\left(\frac{1}{p_\alpha \cdot \varepsilon^2}\right) = \mathcal{O}\left(\frac{1}{\varepsilon^2} \cdot \frac{2^n}{\binom{n}{\alpha n}}\right).$$

We propose a simple estimation using the Stirling formula and Remark 4. As discussed before, we assume β to be equal to $1/2$. We obtain

$$N = \mathcal{O}\left(\frac{2^{n(0.396\alpha(1-\alpha)+1)}}{\binom{n}{\alpha n}}\right) = \tilde{\mathcal{O}}\left(2^{n(0.396\alpha(1-\alpha)+1-h(\alpha))}\right),$$

where h denotes the binary entropy function, i.e. $h(\alpha) = -\alpha \log_2(\alpha) - (1-\alpha)\log_2(1-\alpha)$. The minimum of the complexity exponent is obtained for $\alpha^\star = 0.5$, and our asymptotic time and data complexities are thus in $\tilde{\mathcal{O}}(2^{2 \cdot 0.0495n}) = \tilde{\mathcal{O}}(2^{0.099n})$. If we do not ignore logarithmic factors, for $\alpha = 1/2$, the time and data complexities are in $\mathcal{O}(2^{0.099n}\sqrt{n})$.

3.2 An aggregated distinguishing attack

We now show how to exploit the bias without discarding any input by exploiting the dependence of the bias on the Hamming weight of each sample using the likelihood ratio. The goal is to gain a polynomial factor $\Theta(\sqrt{n})$ in data complexity, lost in the previous attack due to the filtering step, obtaining an estimated asymptotic complexity of $\mathcal{O}(2^{0.099n})$ in data. We fix β and n to some value, and denote by ε_α the bias $\epsilon(\alpha, \beta, n)$. In this section, we assume that the Hamming

[9] The sign of the bias does not matter as we take the absolute value of R.

weight of the key is known.[10] Remark 8 discusses the case where the Hamming weight of the key is unknown.

Algorithm. The new attack is described in Algorithm 2. Similarly to the previous attack, it consists in making N oracle calls. Each sample $(x^{(i)}, y_i)$ is stored in a list $L_{\mathrm{wt}(x)}$. It then computes the likelihood of observing $\mathbf{y} = (y_1, \ldots, y_N)$ under the hypothesis that each y_i is drawn according to $\mathcal{B}(1/2 + \varepsilon_{\mathrm{wt}(x_i)/n})$. Then, it computes the statistic

$$R(\mathbf{y}) \coloneqq \sum_{w=0}^{n} \varepsilon_{w/n} \sum_{(x,y)\in L_w} (-1)^y \,.$$

As detailed in Section 2.4, this statistic allows to estimate the log-likelihood of y. $R(y)$ is the sum of N independent random variables $X_i \coloneqq (-1)^{y_i}\varepsilon_{\alpha_i}$. In the random case and weak-PRF cases respectively, we have $\mathbb{E}_{\$}[X_i] = 0$ and $\mathbb{E}_F[X_i] = -2\varepsilon_{\alpha_i}^2$, leading to

$$\mathbb{E}_{\$}(R) = 0 \qquad \text{and} \qquad \mathbb{E}_F(R) = -2\sum_w |L_w|\varepsilon_{w/n}^2 \,.$$

As before, the threshold T is chosen to be $T \coloneqq \left(\mathbb{E}_F(R) - \mathbb{E}_{\$}(R)\right)/2$.

Algorithm 2 Aggregated attack

Parameters: N, n, α, β.
Assumption: In the case the given oracle is a weak PRF, the key k is drawn uniformly at random among vectors of $\mathbb{F}_2^n$ of Hamming weight βn.
Initialize $n+1$ lists $L_0, \ldots, L_n$
for all $i \in [\![1, N]\!]$ **do**
    Query the oracle, get a pair (x, y).
    Add (x, y) to the list $L_{\mathrm{wt}(x)}$.
$R \leftarrow \sum_{w=0}^{n} \varepsilon_{w/n} \sum_{(x,y)\in L_w} (-1)^y$.
$T \leftarrow -\sum_{w=0}^{n} |L_w|\ \varepsilon_{w/n}^2$
if $R > T$ **then**
    return "Random".
return "weak PRF".

Analysis. Again, to simplify the analysis, we use the heuristic that each list L_w has its expected cardinality $|L_w| = p_{w/n}N$. Using Hoeffding's inequality, we obtain

$$\mathbb{P}_F\left(R \geq T\right) = \mathbb{P}_F\left(R - \mathbb{E}_F(R) \geq -T\right) \leq \exp\left(T/2\right),$$

[10] The complexity claims in [6] rely on the same assumption. In [11], the authors omit all factors bounded by a polynomial in n.

and

$$\mathbb{P}_{\$}\left(R \geq T\right) \geq \mathbb{P}_{\$}\left(|R| < |T|\right) = 1 - \mathbb{P}_{\$}\left(|R| \geq |T|\right) \geq 1 - 2\exp\left(T/2\right).$$

Thus, if

$$N \geq \frac{6.8}{\sum_{w=0}^{n} p_{w/n}\varepsilon_{w/n}^2},$$

then taking $T = -6.8$, and Algorithm 2 distinguishes with advantage at least

$$1 - 3\exp\left(-6.8/2\right) \approx 0.9.$$

Assuming the Hamming weight β of the key is known, this leads to an attack with complexities

$$\textbf{Data} = N, \quad \textbf{Time} = \mathcal{O}\left(N\right), \quad \textbf{Memory} = \mathcal{O}\left(1\right),$$

where the time complexity is measured in number of operations on n-bit words.

Remark 8. This new attack allows a polynomial gain in complexity compared to Section 3.1 when the Hamming weight of the key is known, allowing to have an asymptotic complexity equal to $\mathcal{O}(2^{0.099n})$ instead of $\mathcal{O}(2^{0.099n}\sqrt{n})$. However, when the Hamming weight of the key β is unkown, the attacker needs to guess β and re-run the attack for each guess. On average, $\mathcal{O}(\sqrt{n})$ guesses of β are sufficient to identify the correct value. Moreover, the amount of required data must be increased by a factor of $\mathcal{O}(\log(n))$ in order to maintain the same false-positive rate.[11] When the Hamming weight of the key is unknown, this gives:

$$\textbf{Data} = \mathcal{O}(N\log(n)), \quad \textbf{Time} = \mathcal{O}\left(N\sqrt{n}\log(n)\right), \quad \textbf{Memory} = \mathcal{O}\left(1\right),$$

which has a decreased data complexity compared to the attack of Section 3.1, but a slightly increased time complexity.

Experimental verification We implemented the aggregated distinguisher with known β in `C++` and evaluated it experimentally.

Each experiment consists of 10 independent trials in the weak-PRF setting and 10 independent trials in the random setting. For $n = 250$, we obtain $\log_2 d \approx -27.17$, where

$$d = \sum_{w=0}^{n} p_{w/n}\epsilon(w/n, \beta, n)^2.$$

This gives $N \approx \lceil 6.8/d \rceil \approx 2^{29.94}$. The complete experiment, including the pre-computation of the exact bias values using Lemma 1, runs in one minute on a standard laptop using 8 threads. In all 10 weak-PRF trials and all 10 random trials, the distinguisher correctly classified the instance. For the original parameter

[11] Indeed, in the random case, all $\mathcal{O}(\sqrt{n})$ random runs of the algorithm must return "Random" with high probability.

set $n = 384$, we obtain $\log_2 d \approx -40.58$ leading to $N \approx 2^{43.34}$. In this setting, the experiment becomes significantly more computationally demanding and requires approximately six days on a server equipped with 32 processors. Nevertheless, all 10 weak-PRF trials and all 10 random trials were again classified correctly. Note, however, that a concrete attack scenario would only require attacking a single instance once. In that case, since the attack uses negligible memory, it would remain practical and relatively fast even on a standard laptop. Finally, for the larger parameter set $n = 768$, we obtain $N \approx 2^{81.76}$, which is too large to be tested experimentally in practice.

4 Improving the previous attack with a splitting strategy

In this section, we show how the previous distinguishing attack can be improved using a generic technique that we call the *splitting strategy*. The main idea is to partially reduce the uncertainty coming from the unknown contribution of the secret key to the weak PRF evaluation by fixing or guessing part of this contribution. This amplifies the statistical biases exploited by the distinguisher and therefore reduces the number of samples required for the attack, at the cost of an additional computation over a subset of key candidates. Our approach is inspired by the techniques introduced in [12], where Fast Fourier Transform (FFT) methods are used to accelerate Learning Parity with Noise (LPN) solvers. Denoting by N_n the query complexity of the distinguisher of Section 3.1 against the weak PRF over $\mathbb{F}_2^n$, we will essentially show that, for any integer $0 \leq s \leq n-1$, the splitting strategy gives a new distinguisher with time and memory complexities $\mathcal{O}(s2^s + N_{n-s})$ and data (or query) complexity $\mathcal{O}(N_{n-s})$. Our gain in time complexity comes from the fact that we are able to compute those in batch in time much better than $2^s|N_{n-s}|$.

4.1 General idea

The main idea of the *splitting strategy* is to split the secret key into two parts and condition on the contribution of one of them. Consider an integer $0 \leq s \leq n-1$ and the following two complementary subsets of coordinates

$$I := \{0, \ldots, s-1\}, \qquad J := \{s, \ldots, n-1\}.$$

The main observation behind our approach is that, when the value of k_I is known, the quantity $\langle k_I, x_I \rangle$ can be computed by the attacker, and the output distribution of the function becomes significantly more biased as s grows. Then, given many samples (x, y) our strategy is to determine whether there exists a vector $z \in \mathbb{F}_2^s$ such that the likelihood that y was generated according to the distribution of

$$H\left(\langle z, x_I \rangle + \langle k_J, x_J \rangle\right)$$

is sufficiently high. If this is the case, we decide that we are in the weak-PRF setting, since this likelihood is expected to be large when $z = k_I$.

In a similar manner to the approach of Section 3, we assume the Hamming weight of the secret key on the part J to be known and equal to $\beta(n-s)$.[12]

High-level strategy. Our attack starts by making oracle calls and creating a filtered set L of samples such that for all $(x, y) \in L$ we have that

$$\mathrm{wt}\,(x_J) = \alpha(n-s)\,.$$

As in Section 3, we exploit the fact that the function F is biased for a given Hamming weight of x_J and k_J, and for a given value of $\langle x_I, k_I \rangle$.

Lemma 3. *For any $m \in \mathbb{Z}_6$ and any fixed k_J such that $\mathrm{wt}(k_J) = \beta(n-s)$ we have*

$$\mathbb{P}_{x_J}\left(H\left(m + \langle k_J, x_J \rangle\right) = 1 \mid \mathrm{wt}(x_J) = \alpha(n-s)\right) = \frac{1}{2} + \eta_{(\alpha,\beta,n-s)}\,(m)$$

where

$$\eta_{(\alpha,\beta,n-s)}\,(m) := -\frac{1}{2} + \sum_{j=0}^{n-s} H(m+j) \frac{\binom{\alpha(n-s)}{j}\binom{(1-\alpha)(n-s)}{\beta(n-s)-j}}{\binom{n-s}{\beta(n-s)}}\,.$$

The proof of this lemma is very similar to that of Lemma 1, and is therefore omitted. Denoting by $(x^{(i)}, y_i)$ the i-th element of L, our algorithm evaluates the likelihood that $\mathbf{y} := (y_1, ..., y_{|L|})$ was drawn according to the distribution

$$\mathcal{D} := \mathcal{B}\left(\frac{1}{2} + \eta_{(\alpha,\beta,n-s)}\left(\langle z, x_I^{(i)} \rangle\right)\right)$$

using the score function from Definition 3:

$$R^{(z)} := R_{\mathcal{D}}\,(\mathbf{y}) = \sum_{(x,y)\in L} (-1)^y \eta_{(\alpha,\beta,n-s)}\left(\langle z, x_I \rangle\right). \tag{2}$$

If there exists a vector z such that the score $R^{(z)}$ is smaller than a suitably chosen threshold T, we decide that we are in the weak-PRF setting, since in that case we expect $R^{(k_I)}$ to be negative. More precisely, we choose

$$T := -|L| \max_{m\in\mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m)^2/64.$$

This choice will be justified in the analysis.

Computing the score function. The only step left to describe is how we actually find out whether there exists z with a score meeting the threshold. To do so, we simultaneously compute $R^{(z)}$ for all $z \in \mathbb{F}_2^s$.

[12] In practice, the attacker can guess this quantity with very limited overhead only.

Proposition 1. *There exists a procedure* ComputeScoreFunction *that computes* $R^{(z)}$ *for all* $z \in \mathbb{F}_2^s$ *in time and memory* $\mathcal{O}\left(s2^s\right)$.

The proof and the description of the algorithm are given in Appendix F. Furthermore, throughout this article, the memory complexities are given in the model where we assume that random memory access can be done in constant time. More, note that in our algorithms, the list L will be (implicitly) constructed on the fly. This is the reason why the cardinality of L never appears in the memory complexity of our algorithms.

We are finally ready to give our algorithm.

Algorithm. Our algorithm takes as input the parameters N, n, s, α, β, and assumes that, in the weak-PRF case, the secret key $k \in \mathbb{F}_2^n$ satisfies $\mathrm{wt}\left(k_{\{s,\dots,n-1\}}\right) = \beta(n-s)$.

Algorithm 3 Splitting strategy

Parameters: N, n, s, α, β.
Assumption: In the weak-PRF case, the key k satisfies $\mathrm{wt}\left(k_J\right) = \beta(n-s)$.
$I \leftarrow \{0, \dots, s-1\}$ and $J \leftarrow \{s, \dots, n-1\}$
Initialize an empty L.
for all $i \in [\![1, N]\!]$ **do**
    Query the oracle and receive a pair (x, y).
    if $\mathrm{wt}\left(x_J\right) = \alpha(n-s)$ **then**
        Add (x, y) to the list L.
$R \leftarrow \mathrm{ComputeScoreFunction}\left(L, n, s, \alpha, \beta\right)$. ▷ $R^{(z)}$ is the score evaluated in z
$T \leftarrow -|L| \max_{m \in \mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m)^2/64$,
if $\min_{z \in \mathbb{F}_2^s} R^{(z)} > T$ **then**
    return "Random"
else
    return "weak PRF"

Theorem 1. *There exists a constant* $C > 0$ *such that, for any parameters* N, n, s, α, β *satisfying* $s \geq 17$ *and*

$$N \geq C \frac{s}{\max\limits_{m \in \mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m)^2} \frac{2^{n-s}}{\binom{n-s}{\alpha(n-s)}},$$

There exists an algorithm (Algorithm 3 with some additional stopping conditions) that distinguishes with constant advantage in

$$\boldsymbol{Data} = N, \ \boldsymbol{Time} = \mathcal{O}\left(s2^s + N\right), \ \boldsymbol{Memory} = \mathcal{O}\left(s2^s\right).$$

The complexity part of the theorem is straightforward, and the correctness part, assuming the parameter constraints, is proved in Appendix E.

Remark 9. The precise choice of the threshold is mainly motivated by the simplicity of the proof of Theorem 1. Choosing a slightly different threshold would likely allow to reduce the constant C. In particular, our choice is clearly not optimal when $s = 0$, since the optimal threshold in that case is already given in Section 3.2 and differs from the one used here.

4.2 Trade-off and asymptotic complexity

As such, the splitting strategy introduces a natural trade-off. Increasing s amplifies the exploitable bias and therefore reduces the number of samples required for the attack. However, it also increases the size of the search space $\mathbb{F}_2^s$, making the actual computation of the score more costly. We now derive asymptotic complexities for the splitting strategy. Using the inequality

$$\max_{m \in \mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m) \geq \epsilon(\alpha, \beta, n-s),$$

together with Theorem 1, we obtain that it is sufficient to take

$$N = \mathcal{O}\left(s \cdot \frac{2^{n-s}}{\binom{n-s}{\alpha(n-s)}} \cdot \frac{1}{\epsilon(\alpha,\beta,n-s)^2} \right)$$

oracle queries. Assuming, as before, that $\beta = 1/2$, the asymptotic results from Section 3.1 show that the optimal choice is obtained for $\alpha \approx 1/2$. This gives,

$$N = \mathcal{O}\left(\sqrt{n-s}\ s 2^{0.099(n-s)} \right).$$

The total time complexity of the attack is therefore $\mathcal{O}\left(s2^s + 2^{0.099(n-s)} \right)$. Balancing the two terms gives $s \approx 0.09n$. With this choice of s, the attack achieves time (in operations on n-bit words), memory, and query complexities $\mathcal{O}\left(\sqrt{n-s}\ s 2^{0.09n} \right)$.

4.3 Aggregated splitting strategy

We now briefly describe an aggregated variant of the splitting strategy, obtained by relaxing the constraint that the vectors x_J must have fixed Hamming weight $\alpha(n-s)$. We refer the reader to Section 3.2 for a more detailed discussion of the aggregated approach. However, unlike in Section 3.2, we do not consider all possible weights in $\{0, \ldots, n-s\}$, but only those in

$$D := \{\alpha(n-s) - \sqrt{n-s}, \ldots, \alpha(n-s) + \sqrt{n-s}\},$$

where α is a parameter of the algorithm. This restriction is introduced in order to reduce polynomial factors in the complexity. Indeed, if L denotes the list of oracle samples, then the score function can be written as

$$R^{(z)} := \sum_{w=0}^{n} \sum_{(x,y) \in L_w} (-1)^y \eta_{(w/(n-s),\beta,n-s)}\left(\langle z, x_I \rangle \right),$$

where $L_w = \{(x, y) \in L : \mathrm{wt}(x_J) = w\}$. Each inner sum can be computed efficiently using the ComputeScoreFunction procedure. To reduce the number of calls to this procedure, however, it is preferable to restrict the computation to the weights that contribute most significantly to the sum, namely the weights contained in D.

We now describe the aggregated splitting strategy by highlighting the two main differences with the attack of Theorem 1.

- When querying the oracle, we store a sample (x, y) in the list $L_{\mathrm{wt}(x_J)}$ if $\mathrm{wt}(x_J) \in D$, and discard it otherwise.
- For each weight $w \in D$, we compute the score function associated to the list L_w using the fast ComputeScoreFunction procedure. The resulting score functions are then summed to obtain the final score, and the threshold is adjusted accordingly.

Concrete complexity estimate. We can show, using similar techniques as before as that there exist a constant C such that under the condition

$$N \geq Cs/\left(\sum_{i=\alpha(n-s)-\sqrt{n-s}}^{\alpha(n-s)+\sqrt{n-s}} \max_{m \in \mathbb{Z}_6} \eta_{(i/(n-s),\beta,n-s)}(m)^2 \frac{2^{n-s}}{\binom{n-s}{i}}\right)$$

then there exists an algorithm distinguishing with constant advantage in

$$\textbf{Data} = N,\ \textbf{Time} = \mathcal{O}\left(\sqrt{n-s}\ s2^s + N\right),\ \textbf{Memory} = \mathcal{O}\left(s2^s\right).$$

To compare fairly with our previous attacks, we heuristically estimate that the advantage is greater than 0.9 when $C = 2 \times 6.8$. The intuition behind this choice is that, with s close to 0, the data complexity of this attack should match that of the aggregated attack from Section 3.2. We obtained this value of C by numerically verifying that it accurately reproduces this behavior. The concrete complexities reported in the introduction were obtained with $\alpha = \beta = 0.5$.

5 Improving the analysis of the attack of Johansson et al.

In this section, we revisit the attack of Johansson et al. [11]. Our main result, verified experimentally, is to show that, in certain regimes, the bias behaves differently to what was originally predicted in their work. From this corrected analysis we are able to investigate a larger set of parameters. More importantly, we are able to expose that all keys having a Hamming weight $\beta n \equiv 5 \pmod 6$ yield a bias much lower than expected. When reevaluating the complexity of the attack only on keys satisfying $\beta n \not\equiv 5 \pmod 6$ we obtain a weak-key attack of time complexity $\mathcal{O}(2^{0.147n})$, which is also based on a nearest neighbor search (NNS) approach. Finally, we argue that extending this approach to a key-recovery attack is unlikely to succeed, or at least would require significantly more samples than predicted by the heuristic analysis. This limitation applies both to the original attack of Johansson et al. and to the analysis here.

5.1 Original attack strategy

As explained in Section 2.3, Johansson et al. considered pairs of plaintexts (x, x') satisfying the three following conditions:

- $x_0 = x'_0$
- $\mathrm{wt}(x \oplus x') = \alpha n$
- $|I| = |J| = \alpha n/2$ where $I = \{i \in [n] : (x_i, x'_i) = (1, 0)\}$ and $J = \{i \in [n] : (x_i, x'_i) = (0, 1)\}$.

From now on, we denote by $\mathcal{X}_\alpha$ the set of pairs satisfying these conditions. Johansson et al. then studied the probability

$$\mathbb{P}\left(F(k, x) = 0 \mid F(k, x') = 0,\ (x, x') \in \mathcal{X}_\alpha\right).$$

Our analysis is mainly driven by the, previously overlooked, fact that once the key is sampled, its Hamming weight is fixed as well. This motivates us to re-analyze the probability considered by Johansson et al. conditioned on a fixed key weight.

5.2 A new analysis of the bias

We start by defining precisely the bias we are interested in.

Definition 4. *Let $k \in \mathbb{F}_2^n$ such that $\mathrm{wt}(k) = \beta n$. We let $\gamma(\alpha, \beta, n, k_0)$ be the bias defined by:*

$$\mathbb{P}\left(F(k, x) = 0 \mid F(k, x') = 0, (x, x') \in \mathcal{X}_\alpha\right) = \frac{1}{2} + \gamma(\alpha, \beta, n, k_0).$$

Exact formulas. In order to find an exact formula for this probability we first write the conditional probability as:

$$\frac{\mathbb{P}\left(F(k, x) = F(k, x') = 0 | (x, x') \in \mathcal{X}_\alpha\right)}{\mathbb{P}\left(F(k, x') = 0 | (x, x') \in \mathcal{X}_\alpha\right)} = \frac{\mathbb{E}_{(x,x')\in\mathcal{X}_\alpha}(\mathbb{1}_{\{F(k,x)=0\}}\mathbb{1}_{\{F(k,x')=0\}})}{\mathbb{E}_{(x,x')\in\mathcal{X}_\alpha}(\mathbb{1}_{\{F(k,x')=0\}})}$$

Using this, we can compute a closed formula for both the numerator and denominator, and thus a closed formula for the bias. The closed formula for the numerator and denominator are obtained using a common approach described in Appendix C. Since these formulas are tedious to write, we only give a closed formula for the denominator here and refer the reader to Appendix C for more details.

Lemma 4. *Let $w \coloneqq \alpha n$ and $h \coloneqq \beta n$. The expectation $\mathbb{E}_{(x,x')\in\mathcal{X}_\alpha}(\mathbb{1}_{\{F(k,x')=0\}})$ is equal to:*

$$\sum_{\substack{x_0\in\mathbb{F}_2\\ 0\le u\le w\\ 0\le v\le w/2}} \frac{1}{2}\frac{\binom{w}{u}\binom{n-1-w}{h-k_0-u}}{\binom{n-1}{h-k_0}}\frac{\binom{u}{v}\binom{w-u}{w/2-v}}{\binom{w}{w/2}}\sum_{s=0}^{h-k_0-u}\binom{h-k_0-u}{s}\frac{2^{u+k_0}}{2^h}\mathbb{1}_{\{H(k_0x_0+s+u-v)=0\}}$$

Comparison to Johansson et al. Johansson et al. notice in their experiments that when the Hamming weight of the key is significantly larger than $n/2$, the observed bias becomes larger than predicted by their model. This is displayed in Table I of [11]. This phenomenon had not been theoretically explained by the authors, but is fully captured by our formula. For example, when setting $n = 384, \beta = 310/384$ and $\alpha \in \{12, 16, 24, 30, 36\}$, we obtain similar biases to the ones reported in [11]. Furthermore, our formulas have been experimentally verified for various configurations of n, α, β.

Asymptotic formulas. We next derive asymptotic formulas for the numerator and denominator probabilities, combining them to approximate $\gamma(\alpha, \beta, n, k_0)$. As in Lemma 2, we express the probabilities as sums of evaluations of characteristic functions via Fourier analysis. Again, to derive the asymptotic behavior we assume gaussian behavior and independence of the underlying laws. Those assumptions have been tested experimentally but remain unproven. As they can only be made clear within the context of the proofs, we redirect the reader to Appendix C for both the proofs and the assumptions.

Proposition 2. *Under Assumption 2, the numerator probability satisfies the following asymptotic approximation:*

$$\mathbb{P}\left(F(k,x) = F(k,x') = 0 \mid (x,x') \in \mathcal{X}_\alpha\right) \underset{n\to+\infty}{\sim} \frac{1}{4} + B_1 + B_2 + B_3 + \mathcal{O}(P) \quad (3)$$

where

$$\begin{aligned}
B_1 &= \frac{2}{3}\cos\left(\frac{\pi}{6}\beta n - \frac{\pi}{3}\right)\cos\left(\frac{\pi}{6}k_0\right)\exp\left(-\frac{\pi^2}{72}\beta n(1-\alpha\beta)\right),\\
B_2 &= \frac{2}{9}\cos\left(\frac{\pi}{3}\beta n - \frac{2\pi}{3}\right)\cos\left(\frac{\pi}{3}k_0\right)\exp\left(-\frac{\pi^2}{18}\beta n(1-\alpha)\right),\\
B_3 &= \frac{2}{9}\exp\left(-\frac{\pi^2}{18}\alpha\beta n(1-\beta)\right),\\
P &= B_1^9 + B_2^9 + B_2 B_3^4 + B_2^4 B_3.
\end{aligned}$$

Proposition 3. *Under Assumption 3, the denominator probability satisfies the following asymptotic approximation:*

$$\mathbb{P}\left(F(k,x') = 0 \mid (x,x') \in \mathcal{X}_\alpha\right) \underset{n\to+\infty}{\sim} \frac{1}{2} + C_1 + \mathcal{O}(C_1^9) \quad (4)$$

where

$$C_1 = \frac{2}{3}\cos\left(k_0\frac{\pi}{6}\right)\cos\left(\frac{\pi}{6}\beta n - \frac{\pi}{3}\right)\exp\left(\frac{-\pi^2}{72}\beta n(1-\alpha\beta)\right).$$

Overall, the two asymptotic expressions provide accurate approximations, with only a small number of cases where the actual bias is smaller than predicted. When the two formulas are combined to approximate $\gamma(\alpha, \beta, n, k_0)$, the

approximation error naturally becomes more noticeable. Nevertheless, for parameters such as $n = 384$ and $\beta = 0.5$, the asymptotic estimate differs on average (over all even values of αn) by only a factor of about $2^{-0.53}$ from the exact bias, with a maximal deviation of approximately $2^{1.51}$ attained at $\alpha n = 330$.

5.3 Comparison with the bias of Johansson et al.

In this section, we give a comparison of our new formula and the one from [11]. All the results in this section assume $k_0 = 0$ for clarity purposes, the role of the value of k_0 will be discussed in Section 5.5. In order to illustrate the influence of fixing the Hamming weight of the key, we display in Figure 3 the $\log_2$ of the absolute bias over all even values of αn for both our formula with $n = 384, \beta = 0.5$ and Theorem 2 from [11]. We observed that, once the Hamming weight of the

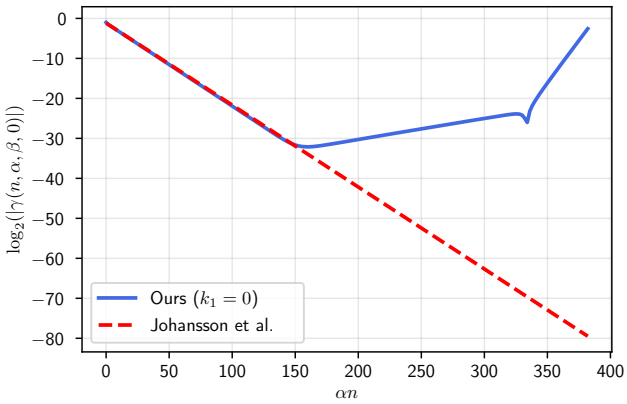


Fig. 3. Base-2 logarithm of the absolute value of the bias for $n = 384$. The red curve corresponds to the asymptotic bias predicted by Johansson et al. [11], while the blue curve corresponds to the bias given by our formula for $\beta = 0.5$.

key is fixed, the prediction of Johansson et al. remains accurate only up to a critical value α^*. Beyond this threshold, their asymptotic formula no longer matches the actual behaviour of the bias. More precisely, instead of continuing to decrease with α, the bias starts increasing again after α^*. Our results therefore show that the average-case analysis does not accurately capture the fixed-key behaviour, whereas the latter is precisely the setting relevant to an attacker. As a consequence, values $\alpha > 0.5$, which were not investigated by Johansson et al. because their heuristic analysis predicted significantly smaller biases, may in fact lead to competitive or even improved attacks. We discuss these new attack scenarios in Section 5.4. Before that, we provide an explanation for this unexpected phenomenon.

Bias Oscillations. To understand why for a fixed key, and in particular, for a fixed β, the behavior differs from the average case, we need to look at the evolution of the bias as a function of β for a fixed value of α. We illustrate this phenomenon in Figure 4 for $\alpha = 0.5$ and $n = 384$.

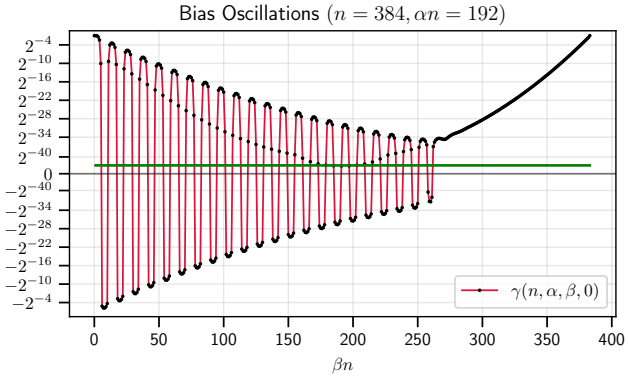


Fig. 4. Evolution of $\gamma(n, 0.5, \beta, 0)$ as a function of β. We also display the expected value of the bias over β in green curve.

The bias actually oscillates between positive and negative values until some threshold β^*, after which it stops oscillating and remains positive. As a consequence, averaging over all values of βn creates destructive interferences between positive and negative contributions, which can dramatically reduce the magnitude of the average bias, even though for most values of β the bias itself is significantly larger. From our experiments, we observed that the value of β^* increases with α. Therefore, when α is small, the oscillations occur far from the most probable values of β and thus have little influence on the average behaviour. However, as α increases, the oscillating region progressively moves towards the dominant values of β, eventually becoming the main contribution to the average bias. This precisely matches the behaviour observed in Figure 3: the analysis of Johansson et al. remains accurate up to a critical value α^*, but fails beyond this point because the average bias no longer reflects the fixed-key behaviour.

Remark 10. As can be observed in Figure 4, the bias is significantly smaller whenever $\beta n \equiv 5 \pmod 6$ than for neighboring values of β. This phenomenon is also accurately captured by the asymptotic formulas. Indeed, when $\beta n \equiv 5 \pmod 6$, the leading term B_1 vanishes in the numerator and the leading term C_1 also disappears in the denominator. As a consequence, the resulting value of γ becomes several orders of magnitude smaller than in the case $\beta n \not\equiv 5 \pmod 6$.

Main takeaway. The main conclusion of this section is that the analysis of Johansson et al. [11] discarded all values $\alpha > 0.5$, since their heuristic analysis predicted biases significantly smaller than those obtained for $\alpha = 0.5$. However,

our analysis shows that this intuition is incorrect: there is no reason to exclude values above 0.5, as the corresponding biases can in fact remain competitive.

Actual average behaviour As shown above, the average behaviour predicted by Johansson et al. does not capture the fixed-weight behaviour. However, even after averaging our formula over β, we find that it still does not coincide with the prediction of [11]. This divergence is illustrated in Figure 5. Again, we can clearly

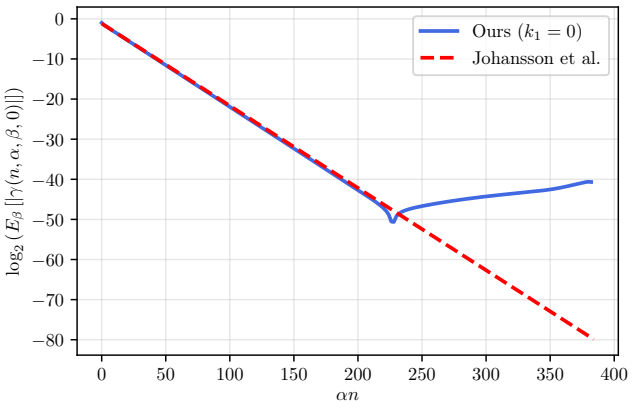


Fig. 5. Base-2 logarithm of the absolute value of the bias for $n = 384$. The red curve corresponds to the asymptotic bias predicted by Johansson et al. [11], while the blue curve corresponds to the bias given by our formula after averaging over βn.

observe that the behaviour predicted by our analysis differs significantly from that of Johansson et al. Since the proof in [11] is not provided in full detail, we were unfortunately unable to identify precisely where their analysis goes wrong. For completeness, we experimentally evaluated the bias for small parameters, namely $n = 40$ and $\alpha n = 36$, by averaging over 10^4 randomly chosen keys. The experimentally observed bias was approximately $2^{-4.96}$. Our formula predicts a bias of $2^{-4.58}$, whereas the formula from [11] predicts a significantly smaller value of $2^{-8.55}$.

5.4 Improved attack

As previously discussed, with our analysis, we have the opportunity to investigate new sets of parameters that were discarded by Johansson et al. Therefore, we obtain a new attack that relies on our asymptotic formulas together with the nearest-neighbor search (NNS) algorithm. Due to the complexity of the formulas, we are not able to derive a closed-form expression for the asymptotic complexity and instead estimate it experimentally up to $n = 2048$. This is shown in Figure 9 of Appendix D. In our experiments, we consider the minimum bias attained

over 99% of the key space, which consequently reduces the number of values of βn that must be considered in the computations, and we look for the value of α minimizing the time complexity. As discussed in Section 5.3, every value of $\beta n \equiv 5 \pmod 6$ leads to a significantly smaller bias than the neighboring cases. Therefore, we also include the complexity obtained when discarding these values from the analysis. From these experiments, we can conjecture that the attack achieves a time complexity of $\mathcal{O}(2^{0.162n})$ and a data complexity of $\mathcal{O}(2^{0.097n})$ for all keys. Yet, whether the observed discrepancy in asymptotic exponents reflects a genuine improvement of the attack or is an artifact of our heuristics remains unclear and is left as an open problem. More interestingly, when discarding the keys such that $\beta n \equiv 5 \pmod 6$, we obtain a time complexity of $\mathcal{O}(2^{0.147n})$ and a data complexity of $\mathcal{O}(2^{0.08n})$. It is also worth noting that this approach uses the same value of α for all values of β. Designing an attack in which the choice of α is tailored to each individual value of β could potentially further improve the complexity. However, we do not expect such an approach to outperform the attack proposed in Section 3.

5.5 Limitations for key recovery

Throughout our experiments, we only observed very small differences between the biases corresponding to the cases $k_0 = 0$ and $k_0 = 1$, whilst the key-recovery attack of Johansson et al. requires distinguishing between these two cases. Our experiments suggest that the corresponding signal is significantly weaker than the one exploited in the distinguishing setting and would require significantly more data. Consequently, although the distinguishing attack remains successful, we believe that extending this approach to a key-recovery attack is unlikely to be possible within the same order of magnitude of complexity.

6 Conclusion

In this work, we revisited the cryptanalysis of the alternative mod-2/mod-3 weak PRF by studying its statistical behavior under fixed Hamming weights for both the secret key and the inputs. This allowed us to isolate significantly stronger biases than in previous works, leading to several improved distinguishing attacks. In particular, we obtained the first practical attack against the original parameter set $n = 384$, with time and data complexity around $2^{43.16}$, and successfully implemented it in practice. This demonstrates that the original parameter recommendations were overly optimistic and further highlights the need for more cryptanalytic study of shallow weak PRF constructions and related primitives.

We also introduced a generic splitting strategy that further improves the attack trade-offs and gives the currently best known attacks against the construction, especially in scenarios where the available data is limited. In particular, for $n = 510$, our attack achieves a time complexity below the claimed security level while using only 2^{45} samples, although the attack remains impractical. For

$n = 768$, which is the currently suggested parameter set for practical applications, several of our attacks also achieve a time complexity below 2^{128}, but require more queries than what is typically available in practical MPC settings.

A promising direction for future work would be to study statistical correlations involving tuples of more than two inputs, under suitable restrictions on the Hamming weights of the inputs or of their pairwise differences. This could potentially lead to new attacks even under strict data limitations.

Use of Generative AI. Several of our proofs, in particular on asymptotic estimations of the biases, were partially guided using generative AI tools. We also used generative AI to implement some of our tests. All the results obtained by AI were thoroughly proofread, justified and re-written by the authors. Similarly, all the code generated by AI has been completely reviewed and checked by the authors. The authors take responsibility for all their results.

Acknowledgments. The authors thank Geoffroy Couteau for fruitful discussions about this project. This work was partially supported by the France 2030 program under grant agreement No. ANR22-PECY-0010 Cryptanalyse.

References

1. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. In: 32nd ACM STOC. pp. 435–440. ACM Press (May 2000). https://doi.org/10.1145/335305.335355
2. Boneh, D., Ishai, Y., Passelègue, A., Sahai, A., Wu, D.J.: Exploring crypto dark matter: New simple PRF candidates and their applications. In: Beimel, A., Dziembowski, S. (eds.) TCC 2018, Part II. LNCS, vol. 11240, pp. 699–729. Springer, Cham (Nov 2018). https://doi.org/10.1007/978-3-030-03810-6_25
3. Boura, C., Couteau, G., Perrin, L., Rotella, Y.: SoK: On shallow weak PRFs A common symmetric building block for MPC protocols. IACR Trans. Symm. Cryptol. **2025**(3), 289–336 (2025). https://doi.org/10.46586/tosc.v2025.i3.289-336
4. Bui, D., Couteau, G., Meyer, P., Passelègue, A., Riahinia, M.: Fast public-key silent OT and more from constrained Naor-Reingold. In: Joye, M., Leander, G. (eds.) EUROCRYPT 2024, Part VI. LNCS, vol. 14656, pp. 88–118. Springer, Cham (May 2024). https://doi.org/10.1007/978-3-031-58751-1_4
5. Carrier, K.: Recherche de Presque-Collisions pour le Décodage et la Reconnaissance de Codes Correcteurs. (Near-collisions finding problem for decoding and recognition of error correcting codes). Ph.D. thesis, Sorbonne University, France (2020), https://tel.archives-ouvertes.fr/tel-03370678
6. Cheon, J.H., Cho, W., Kim, J.H., Kim, J.: Adventures in crypto dark matter: Attacks and fixes for weak pseudorandom functions. In: Garay, J. (ed.) PKC 2021, Part II. LNCS, vol. 12711, pp. 739–760. Springer, Cham (May 2021). https://doi.org/10.1007/978-3-030-75248-4_26
7. Couteau, G., Devadas, L., Devadas, S., Koch, A., Servan-Schreiber, S.: QuietOT: Lightweight oblivious transfer with a public-key setup. In: Chung, K.M., Sasaki, Y. (eds.) ASIACRYPT 2024, Part II. LNCS, vol. 15485, pp. 197–231. Springer, Singapore (Dec 2024). https://doi.org/10.1007/978-981-96-0888-1_7

8. Dinur, I., Goldfeder, S., Halevi, T., Ishai, Y., Kelkar, M., Sharma, V., Zaverucha, G.: MPC-friendly symmetric cryptography from alternating moduli: Candidates, protocols, and applications. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part IV. LNCS, vol. 12828, pp. 517–547. Springer, Cham, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84259-8_18
9. Hoeffding, W.: Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58**(301), 13–30 (1963). https://doi.org/10.1080/01621459.1963.10500830, https://www.tandfonline.com/doi/abs/10.1080/01621459.1963.10500830
10. Hu, K., Leander, G., Raddum, H., Sandrib, A., Udovenko, A.: Cryptanalysis of two alternating moduli weak prfs. IACR Trans. Symmetric Cryptol. **2026**(1), 95–118 (2026). https://doi.org/10.46586/TOSC.V2026.I1.95-118, https://doi.org/10.46586/tosc.v2026.i1.95-118
11. Johansson, T., Meier, W., Nguyen, V.: Differential cryptanalysis of mod-2/mod-3 constructions of binary weak prfs. In: ISIT 2023. pp. 477–482. IEEE (2023). https://doi.org/10.1109/ISIT54713.2023.10206853
12. Levieil, É., Fouque, P.A.: An improved LPN algorithm. In: De Prisco, R., Yung, M. (eds.) SCN 06. LNCS, vol. 4116, pp. 348–359. Springer, Berlin, Heidelberg (Sep 2006). https://doi.org/10.1007/11832072_24
13. Liu, H., Wang, X., Yang, K., Yu, Y.: Authenticated BitGC for actively secure rate-one 2PC. In: Kalai, Y.T., Kamara, S.F. (eds.) CRYPTO 2025, Part IV. LNCS, vol. 16003, pp. 652–687. Springer, Cham (Aug 2025). https://doi.org/10.1007/978-3-032-01884-7_21
14. May, A., Ozerov, I.: On computing nearest neighbors with applications to decoding of binary linear codes. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 203–228. Springer, Berlin, Heidelberg (Apr 2015). https://doi.org/10.1007/978-3-662-46800-5_9
15. Neyman, J., Pearson, E.S.: On the Problem of the Most Efficient Tests of Statistical Hypotheses, pp. 73–108. Springer New York, New York, NY (1992). https://doi.org/10.1007/978-1-4612-0919-5_6, https://doi.org/10.1007/978-1-4612-0919-5_6
16. Razborov, A.A.: Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. Mathematical Notes of the Academy of Sciences of the USSR **41**(4), 333–338 (1987)
17. Smolensky, R.: Algebraic methods in the theory of lower bounds for boolean circuit complexity. In: Aho, A. (ed.) Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC). pp. 77–82. ACM Press (May 1987)

A Bias

Table 2 reports exact values of the bias $\epsilon(\alpha, \beta, n)$, as defined in Equation (1) (Lemma 1), for $n \in \{384, 768\}$ and selected values of α, and β.

B Proof of Lemma 2

Proof. Let $\mathcal{B} = \{x \in \mathbb{F}_2^n \text{ s.t. } \mathrm{wt}(x) = \alpha n \text{ and } F(k, x) = 1\}$. We thus want to compute the cardinality of $\mathcal{B}$. Let X be a random variable that follows a

Table 2. Exact bias values $\epsilon(\alpha, \beta, n)$ for $n \in \{384, 768\}$ and selected values of α, and β, computed from the formula of Lemma 1.

$n = 384$					
$\alpha \backslash \beta$	0.1	0.3	0.5	0.7	0.9
0.1	$+2^{-3.28}$	$+2^{-7.24}$	$-2^{-7.61}$	$+2^{-7.52}$	$+2^{-3.24}$
0.3	$+2^{-7.24}$	$+2^{-14.24}$	$+2^{-17.04}$	$+2^{-16.17}$	$-2^{-7.01}$
0.5	$-2^{-7.61}$	$+2^{-17.04}$	$-2^{-20.84}$	$-2^{-17.91}$	$+2^{-7.98}$
0.7	$+2^{-7.52}$	$+2^{-16.17}$	$-2^{-17.91}$	$-2^{-14.54}$	$-2^{-6.96}$
0.9	$+2^{-3.24}$	$-2^{-7.08}$	$+2^{-7.98}$	$-2^{-6.96}$	$+2^{-3.22}$
$n = 768$					
$\alpha \backslash \beta$	0.1	0.3	0.5	0.7	0.9
0.1	$-2^{-5.37}$	$+2^{-12.22}$	$-2^{-15.62}$	$-2^{-13.15}$	$+2^{-6.10}$
0.3	$+2^{-12.23}$	$-2^{-28.55}$	$-2^{-33.04}$	$-2^{-29.48}$	$+2^{-12.37}$
0.5	$-2^{15.62}$	$-2^{-33.04}$	$-2^{-40.05}$	$+2^{-33.32}$	$+2^{-14.99}$
0.7	$-2^{-13.15}$	$-2^{-29.47}$	$+2^{-33.32}$	$-2^{-33.14}$	$-2^{-14.38}$
0.9	$+2^{-6.11}$	$+2^{-12.37}$	$+2^{-14.99}$	$-2^{-14.380}$	$+2^{-6.54}$

hypergeometric distribution $\mathcal{H}(n, \beta n, \alpha n)$. Let $k \in \mathbb{F}_2^n$ be such that $\mathrm{wt}(k) = \beta n$ and let $j \in \{0, \ldots, n-1\}$. First, we have:

$$\mathcal{B} = \bigcup_{i=3}^{5} \bigcup_{j \equiv i[6]} \left\{ x \in \mathbb{F}_2^n \text{ s.t. } \mathrm{wt}(x) = \alpha n \text{ and } \mathrm{wt}\big(\langle k, x \rangle\big) = j \right\} \tag{5}$$

since $F(k, x) = H(\mathrm{wt}(\langle k, x \rangle))$, $F(k, x) = 1$ if and only if $\mathrm{wt}(\langle k, x \rangle) \mod 6 \in \{3, 4, 5\}$. In Equation (5) the unions are disjoint therefore we can express the cardinality of $\mathcal{B}$ as:

$$\begin{aligned} |\mathcal{B}| &= \sum_{i=3}^{5} \sum_{j \equiv i[6]} \left| \left\{ x \in \mathbb{F}_2^n \text{ s.t. } \mathrm{wt}(x) = \alpha n \text{ and } \mathrm{wt}\big(\langle k, x \rangle\big) = j \right\} \right| \\ &= \sum_{i=3}^{5} \sum_{j=0}^{\alpha n} \left| \left\{ x \in \mathbb{F}_2^n \text{ s.t. } \mathrm{wt}(x) = \alpha n \text{ and } \mathrm{wt}\big(\langle k, x \rangle\big) = j \right\} \right| \delta_{i \equiv j[6]} \end{aligned} \tag{6}$$

Now, it is well-known that:

$$\left|\left\{x \in \mathbb{F}_2^n \text{ s.t. } \mathrm{wt}(x) = \alpha n \text{ and } \mathrm{wt}\big(\langle k, x\rangle\big) = j\right\}\right| = \binom{\beta n}{j}\binom{n-\beta n}{\alpha n - j} = \binom{n}{\alpha n}\mathbb{P}\left(X = j\right) .$$

By substituting this expression in Equation (6), we obtain:

$$|\mathcal{B}| = \sum_{i=3}^{5}\sum_{j=0}^{\alpha n}\binom{n}{\alpha n}\mathbb{P}\left(X = j\right)\ \delta_{\equiv i[6]}(j) .$$

The function $\delta_{\equiv i[6]}$ can be expressed as $\delta_{\equiv i[6]}(j) = \frac{1}{6}\sum_{m=0}^{5}\omega^{m(i-j)}$ where $\omega \coloneqq e^{2i\pi/6}$ is a primitive 6^{th} root of unity. The following equalities hold:

$$\begin{aligned} 6|\mathcal{B}| &= \binom{n}{\alpha n}\sum_{i=3}^{5}\sum_{j=0}^{\alpha n}\mathbb{P}\left(X = j\right)\ \sum_{m=0}^{5}\omega^{m(i-j)} \\ &= \binom{n}{\alpha n}\sum_{i=3}^{5}\sum_{m=0}^{5}\omega^{mi}\sum_{j=0}^{\alpha n}\mathbb{P}\left(X = j\right)\ \omega^{-mj} \\ &= \binom{n}{\alpha n}\sum_{i=3}^{5}\sum_{m=0}^{5}\omega^{mi}\mathbb{E}(\omega^{-mX}) = \binom{n}{\alpha n}\sum_{m=0}^{5}S_m\mathbb{E}(\omega^{-mX}) , \end{aligned}$$

where $S_m \coloneqq \sum_{i=3}^{5}\omega^{im}$ for all $m \in [\![0, 5]\!]$. For a random variable Y, recall that its characteristic function ϕ_Y maps $t \in \mathbb{R}$ to $\mathbb{E}(e^{itY})$. In particular, since $\omega = e^{2i\pi/6} = e^{i\pi/3}$,

$$6|\mathcal{B}| = \binom{n}{\alpha n}\sum_{m=0}^{5}S_m\phi_X\left(\frac{-\pi m}{3}\right) .$$

It can be shown that $S_0 = 3$, $S_1 = 2e^{\frac{2i\pi}{3}}$, $S_2 = S_4 = 0$, $S_3 = -1$ and $S_5 = \bar{S_1}$. Denoting by $\Re(z)$ the real part of a complex number $z \in \mathbb{C}$, it thus holds that:

$$\begin{aligned} 6|\mathcal{B}| &= \binom{n}{\alpha n}\sum_{m=0}^{5}S_m\phi_X\left(-\frac{\pi m}{3}\right) \\ &= \binom{n}{\alpha n}\left[3 - \Re\big(\phi_X\left(-\pi\right)\big) + 2\cdot\Re\left(S_1\cdot\phi_X\left(-\frac{\pi}{3}\right)\right)\right] . \end{aligned}$$

Assumption 1 *We assume that X can be approximated by the gaussian law $\mathcal{N}(\mu n, \sigma^2 n)$ with $\mu = \alpha\beta$ and $\sigma^2 = \alpha\beta(1-\alpha)(1-\beta)$. This has been tested experimentally as it can be seen in Figure 6.*

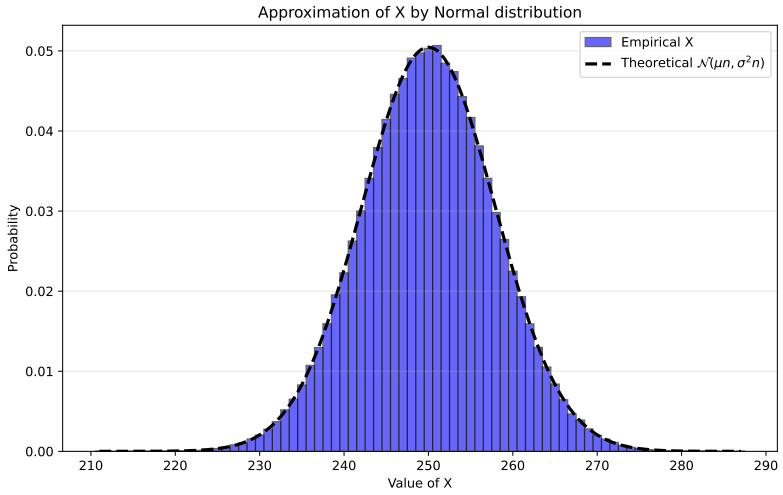


Fig. 6. Distribution of X for $n = 10000, \alpha = \beta = 0.5$ with $500,000$ samples.

By Assumption 1, we can approximate at infinity the characteristic function of X with that of its gaussian approximation:

$$\phi_X(t) \underset{n\to+\infty}{\sim} \exp\left(it\mu n - \frac{t^2\sigma^2 n}{2}\right)$$

We obtain the following approximation for $6|\mathcal{B}|$:

$$\begin{aligned} 3 - \Re\left(\phi_X(\pi)\right) + 2\cdot\Re\left(S_1\cdot\phi_X(\frac{\pi}{3})\right) \underset{n\to+\infty}{\sim} 3 - \Re\left(\exp\left(-i\pi\mu n - \frac{\pi^2\sigma^2 n}{2}\right)\right) \\ + 4\exp\left(\frac{-\pi^2\sigma^2}{18}n\right)\cos\left(\frac{\pi}{3}\mu n + \frac{2\pi}{3}\right) \end{aligned}$$

Finally, dividing this by $\binom{n}{\alpha n}$ yields the claimed formula for $\epsilon(\alpha,\beta,n)$:

$$\epsilon(\alpha,\beta,n) \underset{n\to+\infty}{\sim} \frac{2}{3}\exp\left(\frac{-\pi^2\sigma^2}{18}n\right)\cos\left(\frac{\pi}{3}\mu n + \frac{2\pi}{3}\right) + \mathcal{O}\left(\exp\left(\frac{-\pi^2\sigma^2}{2}n\right)\right).$$

□

C Proofs of Section 5

C.1 Proof of Lemma 4

Proof. We let $h \coloneqq \beta n$ be the Hamming weight of the key and $w \coloneqq \alpha n$ be the Hamming weight of $x \oplus x'$. We start by partitioning the index set $\{0, \dots, n-1\}$ into three disjoint subsets:

- $\{0\}$, this index is under the constraint that $x_0 = y_0$.
- $D = I \cup J,\ |D| = w$.
- $C = \{i \in [n] \setminus \{1\} : x_i = y_i\},\ |C| = n - 1 - w$.

Therefore, we can express the scalar products $\langle k, x\rangle, \langle k, x'\rangle$ as:

$$\begin{aligned}\langle k, x\rangle &= k_0 x_0 + \langle k, x\rangle_C + \langle k, x\rangle_D \\ \langle k, x'\rangle &= k_0 x_0 + \langle k, x'\rangle_C + \langle k, x'\rangle_D\end{aligned}$$

By definition of C, we have $\langle k, x\rangle_C = \langle k, x'\rangle_C \coloneqq s$. Then, let $u = \sum_{i \in D} k_i$ be the weight of the key in D and $v = \langle k, x\rangle_D$. Then $\langle k, x'\rangle_D = u - v$. Thus:

$$\langle k, x\rangle = k_0 x_0 + s + v, \quad \langle k, x'\rangle = k_0 x_0 + s + (u - v)$$

and

$$\begin{aligned}F(k, x) &= H(\langle k, x\rangle) = H(k_0 x_0 + s + v) \\ F(k, x') &= H(\langle k, x'\rangle) = H(k_0 x_0 + s + u - v)\,.\end{aligned}$$

From this decomposition, we can write:

$$\mathbb{E}_{(x,x') \in \mathcal{X}_\alpha}(\mathbb{1}_{\{F(k,x')=0\}}) = \mathbb{E}_{x_0,u,v,s}(\mathbb{1}_{\{H(k_0 x_0 + s + u - v)=0\}})$$

Let $X_0, \mathcal{S}, \mathcal{U}, \mathcal{V}$ be the distributions of x_0, s, u, v respectively. X_0 is independent of the three other laws and follows the law $\mathcal{B}(1/2)$.

Distribution of $\mathcal{U}$. $\mathcal{U}$ counts the number of ones in the support of the difference D and follows $\mathcal{H}(n-1, w, h - k_0)$:

$$\mathbb{P}\left(\mathcal{U} = u\right) = \frac{\binom{w}{u}\binom{n-1-w}{h-k_0-u}}{\binom{n-1}{h-k_0}}$$

Distribution of $\mathcal{V} \mid \mathcal{U}$. Given that there are u bits of the key in D, and given the constraint that x has $w/2$ bits in D, $\mathcal{V} \mid \mathcal{U}$ follows $\mathcal{H}(w, u, w/2)$:

$$\mathbb{P}\left(\mathcal{V} = v \mid \mathcal{U} = u\right) = \frac{\binom{u}{v}\binom{w-u}{w/2-v}}{\binom{w}{w/2}}$$

Distribution of Shared Sum $\mathcal{S} \mid \mathcal{U}, \mathcal{V}$ Since the shared coordinates are uniformly drawn at random and that there are $h - k_0 - u$ active bits in C then $\mathcal{S} \mid \mathcal{U}, \mathcal{V} \sim \mathcal{B}(h - k_0 - u, 1/2)$.

By successive conditioning, the expectancy $\mathbb{E}_{x_0,u,v,s}(\mathbb{1}_{\{H(k_0x_0+s+u-v)=0\}})$ is:

$$\sum_{\substack{x_0 \in \mathbb{F}_2 \\ 0 \le u \le w \\ 0 \le v \le w/2}} \frac{1}{2} \frac{\binom{w}{u}\binom{n-1-w}{h-k_0-u}}{\binom{n-1}{h-k_0}} \frac{\binom{u}{v}\binom{w-u}{w/2-v}}{\binom{w}{w/2}} \sum_{s=0}^{h-k_0-u} \binom{h-k_0-u}{s} \frac{2^{u+k_0}}{2^h} \mathbb{1}_{\{H(k_0x_0+s+u-v)=0\}}$$

This concludes the proof of Lemma 4. A similar proof can be driven to obtain a closed formula for the probability of $\{F(k,x') = F(k,x) = 0\}$. It follows the exact same structure but $\mathbb{1}_{\{F(k,x')=0\}}$ should be replaced by $\mathbb{1}_{\{F(k,x)=0\}}\mathbb{1}_{\{F(k,x')=0\}}$. □

C.2 Proof of Proposition 2

In this proof, we reuse the notations introduced in the proof of Lemma 4. Moreover, the sketch of the proof is quite similar to the one of Lemma 1. We will rewrite the expectancies in terms of evaluations of characteristic functions for some given laws and then derive the final formulas by asymptotically estimating their values. Let us define $\mathcal{Y} = 2k_0X_0 + 2\mathcal{S} + \mathcal{U}$ and $\mathcal{Z} = 2\mathcal{V} - \mathcal{U}$. They correspond to the laws of $\langle x,k\rangle + \langle x',k\rangle$ and $\langle x,k\rangle - \langle x',k\rangle$. Therefore, $\langle x,k\rangle = (\mathcal{Y}+\mathcal{Z})/2$ and $\langle x',k\rangle = (\mathcal{Y}-\mathcal{Z})/2$. The reason why we introduce those variables is because $\mathcal{Y}$ and $\mathcal{Z}$ behave like independent gaussian distributions when n goes to infinity, under the constraint $X_0 = x_0$ for some $x_0 \in \{0,1\}$. This is stated in Assumption 2.

Assumption 2 *Under the condition $X_0 = x_0$ for some value $x_0 \in \{0,1\}$, we can asymptotically approximate the characteristic functions of $\mathcal{Y}$ and $\mathcal{Z}$ by that of $\mathcal{N}\Big(2k_0x_0 + h - k_0, n\beta(1-\alpha)\Big)$ and $\mathcal{N}\Big(0, n\alpha\beta(1-\beta)\Big)$, respectively. Furthermore, asymptotically $\mathcal{Y}$ and $\mathcal{Z}$ behave like independent laws. This has been tested experimentally as it can be seen in Figure 7.*

We can rewrite $\mathbb{1}_{\{H(x)=0\}}$ with Fourier analysis, that is:

$$\mathbb{1}_{\{H(x)=0\}} = \sum_{j=0}^{5} c_j \omega^{jx} \tag{7}$$

where $c_0 = \frac{1}{2}, c_1 = \frac{1}{3}e^{-\frac{i\pi}{3}}, c_2 = 0, c_3 = \frac{1}{6}, c_4 = 0, c_5 = \overline{c_1}$ and $\omega = e^{\frac{i\pi}{3}}$

We want to compute

$$\mathop{\mathbb{E}}_{x_0,u,v,s}(\mathbb{1}_{\{H(\langle x,k\rangle)=0\}}\mathbb{1}_{\{H(\langle x',k\rangle)=0)\}})$$

which, according to the law of total expectation, is equal to

$$\frac{1}{2}\sum_{x_0\in\{0,1\}} \mathop{\mathbb{E}}_{u,v,s}\left[\mathbb{1}_{\{H(\frac{\mathcal{Y}+\mathcal{Z}}{2})=0\}}\mathbb{1}_{\{H(\frac{\mathcal{Y}-\mathcal{Z}}{2})=0\}} \mid X_0 = x_0\right]$$

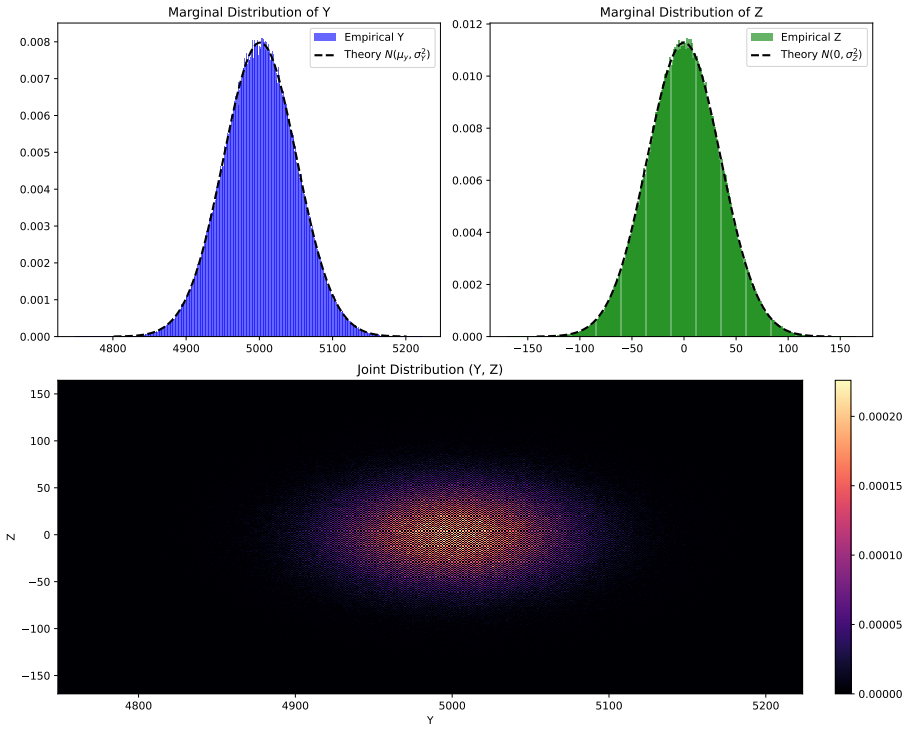


Fig. 7. Distributions of $\mathcal{Y}$, $\mathcal{Z}$ and their joint distribution $(\mathcal{Y}, \mathcal{Z})$ for the parameters $n = 10,000$, $\alpha = \beta = 0.5, k_0 = x_0 = 1$ with $500,000$ random samples.

For a fixed value of x_0 we have:

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{u,v,s}\left[\mathbb{1}_{\{H(\frac{\mathcal{Y}+\mathcal{Z}}{2})=0\}}\mathbb{1}_{\{H(\frac{\mathcal{Y}-\mathcal{Z}}{2})=0\}} \mid X_0 = x_0\right] \\
&= \mathop{\mathbb{E}}_{u,v,s}\left[\sum_{j=0}^{5} c_j \omega^{j\frac{\mathcal{Y}+\mathcal{Z}}{2}} \times \sum_{k=0}^{5} c_k \omega^{k\frac{\mathcal{Y}-\mathcal{Z}}{2}} \mid X_0 = x_0\right] \\
&= \sum_{0\le j,k\le 5} c_j c_k \mathbb{E}\left[\omega^{j\frac{\mathcal{Y}+\mathcal{Z}}{2}+k\frac{\mathcal{Y}-\mathcal{Z}}{2}} \mid X_0 = x_0\right] \\
&= \sum_{0\le j,k\le 5} c_j c_k \mathbb{E}\left[\omega^{\mathcal{Y}\frac{(j+k)}{2}} \omega^{\mathcal{Z}\frac{(j-k)}{2}} \mid X_0 = x_0\right] \\
&= \sum_{0\le j,k\le 5} c_j c_k \mathbb{E}\left[\omega^{\mathcal{Y}\frac{(j+k)}{2}} \omega^{\mathcal{Z}\frac{(j-k)}{2}} \mid X_0 = x_0\right] \\
&= \sum_{0\le j,k\le 5} \underbrace{c_j c_k \phi_{\mathcal{Y}}\left(\frac{\pi}{6}(j+k)\right)\phi_{\mathcal{Z}}\left(\frac{\pi}{6}(j-k)\right)}_{:=D(j,k)}
\end{aligned}
$$

where $\phi_{\mathcal{Y}}, \phi_{\mathcal{Z}}$ respectively denote the characteristic functions of $\mathcal{Y}$ and $\mathcal{Z}$ under the condition $X_0 = x_0$. The last equality uses the assumed independence of those two laws.

Now, by properties of a characteristic function we have $D(-j \bmod 6, -k \bmod 6) = \overline{D(j,k)}$. This means that the the sum can be rewritten by:

$$\sum_{0\leq j,k\leq 5} D(j,k) = \sum_{(j,k)\in\{0,1,3,-1\}^2} \Re\left(D(j,k)\right) \tag{8}$$

Lemma 5. *Under Assumption 2, we have that:*

$$\begin{aligned}\Re(D(j,k)) \underset{n\to+\infty}{\sim} |c_j||c_k| \cos\left(\frac{\pi}{6}\big(A_j + A_k + (j+k)(2k_0x_0 + h - k_0)\big)\right) \\ \times \exp\left(-\frac{\pi^2}{72}\left[(j+k)^2\sigma_{\mathcal{Y}}^2 + (j-k)^2\sigma_{\mathcal{Z}}^2\right]\right)\end{aligned} \tag{9}$$

Proof. Using Assumption 2, we can replace the characteristic functions of $\mathcal{Y}$ and $\mathcal{Z}$ under the condition $X_0 = x_0$ by that of their gaussian approximations:

$$\phi_{\mathcal{Y}}(t) \underset{n\to+\infty}{\sim} \exp\left(it\mu_{\mathcal{Y}} - \frac{1}{2}t^2\sigma_{\mathcal{Y}}^2\right), \quad \mu_{\mathcal{Y}} = 2k_0x_0 + h - k_0, \quad \sigma_{\mathcal{Y}}^2 = n\beta(1-\alpha)$$

$$\phi_{\mathcal{Z}}(t) \underset{n\to+\infty}{\sim} \exp\left(-\frac{1}{2}t^2\sigma_{\mathcal{Z}}^2\right), \quad \mu_{\mathcal{Z}} = 0, \quad \sigma_{\mathcal{Z}}^2 = n\alpha\beta(1-\beta)$$

This yields

$$\begin{aligned}&\Re(D(j,k)) \\ &\underset{n\to+\infty}{\sim} \Re\left[c_jc_k \exp\left(i\frac{\pi}{6}(j+k)\mu_{\mathcal{Y}} - \frac{1}{2}\left(\frac{\pi}{6}(j+k)\right)^2\sigma_{\mathcal{Y}}^2\right)\exp\left(-\frac{1}{2}\left(\frac{\pi}{6}(j-k)\right)^2\sigma_{\mathcal{Z}}^2\right)\right] \\ &\underset{n\to+\infty}{\sim} \Re\left[c_jc_k \exp\left(i\frac{\pi}{6}(j+k)\mu_{\mathcal{Y}}\right)\right]\exp\left(-\frac{\pi^2}{72}\left[(j+k)^2\sigma_{\mathcal{Y}}^2 + (j-k)^2\sigma_{\mathcal{Z}}^2\right]\right)\end{aligned}$$

By replacing the Fourier coefficients c_j by their polar form $|c_j|e^{i\frac{\pi}{6}A_j}$ we can further refine the left term of the formula:

$$\Re\left(c_jc_k\exp\left(i\frac{\pi}{6}(j+k)\mu_{\mathcal{Y}}\right)\right) = |c_j||c_k|\cos\left(\frac{\pi}{6}\big(A_j + A_k + (j+k)(2k_0x_0 + h - k_0)\big)\right)$$

Putting all of this together we obtain the claimed asymptotical approximation for $\Re(D(j,k))$. □

Now, we still need to average this formula over x_0, by angle addition formulas we have:

$$\begin{aligned}\frac{1}{2}\sum_{x_0\in\{0,1\}} \Re(D(j,k)) =& |c_j||c_k| \exp\left(-\frac{\pi^2}{72}\left[(j+k)^2\sigma_{\mathcal{Y}}^2 + (j-k)^2\sigma_{\mathcal{Z}}^2\right]\right) \\ &\times \cos\left(\frac{\pi}{6}(j+k)k_0\right)\cos\left(\frac{\pi}{6}\big[(j+k)h + A_j + A_k\big]\right)\end{aligned} \tag{10}$$

Finally, to obtain the most complete formula we can sum Equation (10) over all pairs $(j,k) \in \{0,1,3,-1\}$. Yet, we can see that the decay rate of each pair is governed by the value of $(j+k)^2$ and $(j-k)^2$. From this, we can extract four leading terms.

Case $j = k = 0$**:**
This case contributes by a term equal to $\frac{1}{4}$.

Case $(j,k) \in \{(0,1),(0,-1),(1,0),(-1,0)\}$**:**
Those terms correspond to $j+k = \pm 1$ and $j-k = \pm 1$ and therefore, by parity of the cosine function, they all have the same contribution of $\frac{1}{6}\cos(\frac{\pi}{6}\beta n - \frac{\pi}{3})\cos(\frac{\pi}{6}k_0)\exp(\frac{-\pi^2}{72}n\beta(1-\alpha\beta))$. Summing this four times yields the coefficient B_1.

Case $(j,k) \in \{(1,1),(-1,-1)\}$**:**
Those terms correspond to $j+k = \pm 2$ and $j-k = 0$. Again, by parity of the cosine function, they all have the same contribution of $\frac{1}{9}\cos(\frac{\pi}{3}\beta n - \frac{2\pi}{3})\cos(\frac{\pi}{3}k_0)\exp(\frac{-\pi^2}{18}n\beta(1-\alpha))$. Summing this twice yields the coefficient B_2.

Case $(j,k) \in \{(1,-1),(-1,1)\}$**:**
Those terms correspond to $j+k = 0$ and $j-k = \pm 2$ and as before they all have the same contribution of $\frac{1}{9}\exp(\frac{-\pi^2}{18}n\alpha\beta(1-\beta))$. Summing this twice yields the coefficient B_3.

Remaining terms $(3,0),(0,3),(3,1),(1,3),(-1,3),(3,-1),(3,3)$**:**
By careful examination of the exponential decays of those remaining terms one can conclude the proof.

C.3 Proof of Proposition 3

As for Proposition 2, we also use the notations introduced in the proof of Lemma 4. Here we are interested in computing $\mathop{\mathbb{E}}_{x_0,u,v,s}(\mathbb{1}_{\{H(x_0k_0+s+v)=0\}})$. Using Fourier analysis we can express the indicator function $\mathbb{1}_{\{H(x)=0\}}$ as $\sum_{j=0}^{5} c_j \omega^{jx}$ with $c_0 = \frac{1}{2}$, $c_1 = \frac{1}{3}e^{-i\frac{\pi}{3}}, c_2 = c_4 = 0, c_3 = \frac{1}{6}, c_5 = \frac{1}{3}e^{i\frac{\pi}{3}}$ and $\omega = e^{i\frac{\pi}{3}}$.

Therefore, we can rewrite the expectancy as:

$$\begin{aligned}\mathop{\mathbb{E}}_{x_0,u,v,s}\left[\mathbb{1}_{\{H(x_0k_0+s+v)=0\}}\right] &= \mathop{\mathbb{E}}_{x_0,u,v,s}\left[\sum_{j=0}^{5} c_j\omega^{j(x_0k_0+s+v)}\right] \\ &= \sum_{j=0}^{5} c_j \mathop{\mathbb{E}}_{x_0,u,v,s}\left[\omega^{j(x_0k_0+s+v)}\right] \\ &= \sum_{j\in\{0,1,3,-1\}} \Re\left(c_j \mathop{\mathbb{E}}_{x_0,u,v,s}\left[\omega^{j(x_0k_0+s+v)}\right]\right)\end{aligned}$$

where the last equality comes from $c_{-j \bmod 6} = \overline{c_j}$ and by property of the characteristic function.

Moreover,

$$\mathop{\mathbb{E}}_{x_0,u,v,s}\left[\omega^{j(x_0k_0+s+v)}\right] = \mathop{\mathbb{E}}_{x_0}[\omega^{j(x_0k_0)}] \mathop{\mathbb{E}}_{u,v,s}\left[\omega^{j(s+v)}\right]$$

and $\mathop{\mathbb{E}}_{x_0}[\omega^{j(x_0k_0)}] = \dfrac{1+w^{jk_0}}{2} = e^{i\frac{\pi jk_0}{6}}\cos(\dfrac{\pi}{6}jk_0)$. We are now interested in finding the value of $\mathop{\mathbb{E}}_{u,v,s}\left[\omega^{j(s+v)}\right]$. We can condition everything on u by using the law of total expectation which yields:

$$\mathop{\mathbb{E}}_{u,v,s}\left[\omega^{j(s+v)}\right] = \mathop{\mathbb{E}}_{u}\left[\mathop{\mathbb{E}}_{(v,s)|u}\left[\omega^{j(s+v)}\right]\right]$$

Assumption 3 *We assume that $(\mathcal{S}+\mathcal{V} \mid \mathcal{U})$ can be asymptotically approximated by $\mathcal{N}(\dfrac{\beta n - k_0}{2}, \dfrac{\beta(1-\alpha\beta)n}{4})$ This assumption has been tested experimentally as it can be seen in Figure 8.*

By replacing the characteristic function of $(\mathcal{S}+\mathcal{V}) \mid \mathcal{U}$ by that of its gaussian approximation, we have:

$$\begin{aligned}\mathop{\mathbb{E}}_{u,v,s}\left[\omega^{j(s+v)}\right] &\underset{n\to+\infty}{\sim} \mathbb{E}_u\left[\exp\left(i\frac{\pi}{3}j\frac{\beta n-k_0}{2} - \frac{\pi^2}{72}j^2\beta(1-\alpha\beta)n\right)\right] \\ &\underset{n\to+\infty}{\sim} \exp\left(i\frac{\pi}{3}j\frac{\beta n-k_0}{2} - \frac{\pi^2}{72}j^2\beta(1-\alpha\beta)n\right)\end{aligned}$$

In the end, the contribution of the term j is given by:

$$\begin{aligned}&\Re\left(c_j e^{i\frac{\pi jk_0}{6}}\cos(\frac{\pi}{6}jk_0)\exp\left(i\frac{\pi}{3}j\frac{\beta n-k_0}{2} - \frac{\pi^2}{72}j^2\beta(1-\alpha\beta)n\right)\right) \\ &= |c_j|\cos(\frac{\pi}{6}jk_0)\cos(\frac{\pi}{6}(j\beta n + A_j))\exp(-\frac{\pi^2}{72}j^2\beta(1-\alpha\beta)n)\end{aligned}$$

where $c_j = |c_j|e^{i\frac{\pi}{6}A_j}$

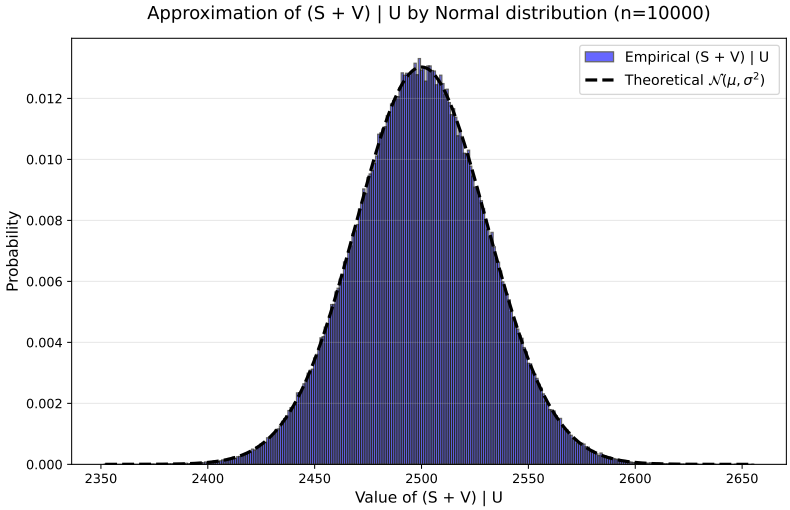


Fig. 8. Distribution of $(\mathcal{S}+\mathcal{V}) \mid \mathcal{U}$ for $n = 10000, \alpha = \beta = 0.5$ with $500,000$ samples.

Case $j = 0$:

This case contributes by a term equal to $\frac{1}{2}$.

Case $j \in \{1, -1\}$:

Those terms contributes with the same value of

$$\frac{1}{3}\cos(\frac{\pi}{6}k_0)\cos(\frac{\pi}{6}(\beta n - 2))\exp(-\frac{\pi^2}{72}\beta(1-\alpha\beta)n)$$

which yields the term C_1.

Case $j = 3$:

This term contribute with a term equal to

$$\frac{1}{6}\cos(\frac{\pi}{2}k_0)\cos(\frac{\pi}{2}\beta n)\exp(-\frac{9\pi^2}{72}\beta(1-\alpha\beta)n)$$

which is a $\mathcal{O}(C_1^9)$.

This concludes the proof of Proposition 3.

D Asymptotic time complexity of Section 5.4

E Proof of Theorem 1

We recall our theorem here.

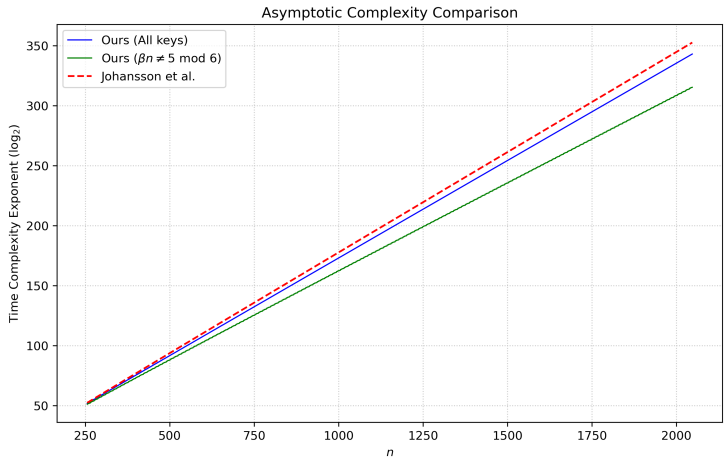


Fig. 9. Comparison of asymptotic time complexities between our attacks and [11].

Theorem 1. *There exists a constant $C > 0$ such that, for any parameters N, n, s, α, β satisfying $s \geq 17$ and*

$$N \geq C \frac{s}{\max\limits_{m \in \mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m)^2} \frac{2^{n-s}}{\binom{n-s}{\alpha(n-s)}},$$

There exists an algorithm (Algorithm 3 with some additional stopping conditions) that distinguishes with constant advantage in

$$\textbf{\textit{Data}} = N, \ \textbf{\textit{Time}} = \mathcal{O}\left(s2^s + N\right), \ \textbf{\textit{Memory}} = \mathcal{O}\left(s2^s\right).$$

Proof. As mentioned earlier, we only have to show the correctness of the algorithm. The algorithm in the Theorem is simply Algorithm 3 with the additional instructions that (i) the algorithm stops and returns "weakPRF" with probability 0.5 if the cardinality of L is smaller than its expectancy

$$E := N \frac{\binom{n-s}{\alpha(n-s)}}{2^{n-s}}$$

and (ii) the samples are discarded as soon as the list L is of size E (L is truncated to be of size E). This ensures that if the list L is not big enough (smaller than E) then the algorithm is not run, and, if the list is big enough then its cardinality is exactly E. With this said, and using the fact that $\mathbb{P}\left(|L| > E\right) > 1/4$ we only have to prove that, conditioned on the event that $|L| = E$, then the advantage of the algorithm is constant to prove our result. This is what do in what follows and we will condition all our probabilities to the event that $|L| = E$.

The proof relies on Hoeffding's inequality. We begin by observing that each term appearing in the score function which we recall is defined as

$$R^{(z)} := \sum_{(x,y) \in L} (-1)^y \eta_{(\alpha,\beta,n-s)}\left(\langle z, x_I \rangle\right), \quad \text{where } I := \{0, \ldots, s-1\}$$

is bounded in absolute value by

$$\eta_{\max} \stackrel{\text{def}}{=} \max_{m \in \mathbb{Z}_6} |\eta_{(\alpha,\beta,n-s)}(m)|.$$

Indeed, for every $(x, y) \in L$ and every $z \in \mathbb{F}_2^s$,

$$\left|(-1)^b \eta_{(\alpha,\beta,n-s)}(\langle x_I, z\rangle)\right| \leq \eta_{\max}.$$

We now recall that the threshold was defined as $T \stackrel{\text{def}}{=} -\frac{|L|\eta_{\max}^2}{64} = -\frac{E\eta_{\max}^2}{64}$ and that we want to bound in each case, when the oracle is random and when the oracle is a weakPRF, the probability that the algorithm returns weakPRF, that is that there exist a candidate $z \in \mathbb{F}_2^s$ such that $R^{(z)} < T$. We will reuse the notation

$$I = \{0, \ldots, s-1\}, \qquad J = \{s, \ldots, n-1\}.$$

Case 1 (uniform distribution). Assume first that L is composed of E independent samples (x, y) where the x's are taken independently uniformly at random in $\mathbb{F}_2^n$ such that $\mathrm{wt}(x_J) = \alpha(n-s)$ and the y's are drawn independently uniformly in $\mathbb{F}_2$. We show that provided that C is sufficiently big,

$$\mathbb{P}\left(\min_{z \in \mathbb{F}_2^s} R^{(z)} < T\right) \leq 0.1 \tag{11}$$

Fix some $z \in \mathbb{F}_2^s$. Since y is uniform in $\mathbb{F}_2$, we have $\mathbb{E}\left(R^{(z)}\right) = 0$. Moreover, $R^{(z)}$ is a sum of N independent random variables whose absolute value are bounded by $\eta_{\max}$. Hoeffding's inequality therefore gives

$$\begin{aligned}
\mathbb{P}\left(R^{(z)} < T\right) &\leq \mathbb{P}\left(\left|R^{(z)}\right| > |T|\right) \\
&\leq 2\exp\left(-T^2/2E\eta_{\max}^2\right) \\
&\leq 2\exp\left(-\frac{E\eta_{\max}^2}{2(64^2)}\right) \\
&\leq 2\exp\left(-\frac{Cs}{2(64^2)}\right)
\end{aligned}$$

Applying the union bound over all 2^s possible values of z yields

$$\mathbb{P}\left(\max_{z \in \mathbb{F}_2^s} R^{(z)} > T\right) \leq 2^s 2\exp\left(-\frac{Cs}{2(64^2)}\right)$$

Provided that C is as a sufficiently large constant we have (11).

Case 2 (Weak PRF case). In this case k is drawn uniformly at random in $\mathbb{F}_2^n$ such that $\mathrm{wt}(k_J) = \beta(n-s)$ (as described in the assumption of algorithm 2) and L is composed of N independent samples $(x, H(\langle x, k\rangle))$ where the x's are taken independently uniformly at random in $\mathbb{F}_2^n$ and such that $\mathrm{wt}(x_J) = \alpha(n-s)$.

We have that

$$\mathbb{P}\left(R^{(k_I)} < T\right) \geq \sum_{w=6}^{n} \mathbb{P}\left(R^{(k_I)} < T \mid \mathrm{wt}(k_I) = w\right) \mathbb{P}\left(\mathrm{wt}(k_I) = w\right) \tag{12}$$

Let $w \geq 6$. We will now upper bound $\mathbb{P}\left(R^{(k_I)} < T \mid \mathrm{wt}(k_I) = w\right)$ by using Hoeffding's inequality. We first compute the expected value of $R^{(k_I)}$, we denote it by

$$V := \mathbb{E}\left(R^{(k_I)} \mid \mathrm{wt}(k_I) = w\right).$$

We have

$$V = -2E \sum_{m \in \mathbb{Z}_6} \mathbb{P}\left(\langle x_I, k_I \rangle = m \mid \mathrm{wt}(k_I) = w\right) \eta_{(\alpha,\beta,n-s)}(m)^2.$$

Since $w \geq 6$, each residue modulo 6 appears with probability at least a positive constant. In particular,

$$\forall m \in \mathbb{Z}_6, \qquad \mathbb{P}\left(\langle x_I, k_I \rangle = m \mid \mathrm{wt}(k_I) = w\right) \geq \frac{1}{32}.$$

Therefore,

$$V \leq -2\frac{E}{32} \max_{m \in \mathbb{Z}_6} \eta_{(\alpha,\beta,n-s)}(m)^2 = 2T \tag{13}$$

$$\begin{aligned}
\mathbb{P}\left((R^{(k_I)} < T \mid \mathrm{wt}(k_I) = w\right) &= \mathbb{P}\left(R^{(k_I)} - V < T - V \mid \mathrm{wt}(k_I) = w\right) \\
&\geq \mathbb{P}\left(R^{(k_I)} - V < -T \mid \mathrm{wt}(k_I) = w\right) \\
&= 1 - \mathbb{P}\left(R^{(k_I)} - V \geq -T \mid \mathrm{wt}(k_I) = w\right) \\
&\geq 1 - \exp(-T^2/2E\eta_{\max}^2) \\
&= 1 - \exp(-Cs/2(64)^2) \\
&\geq 0.9
\end{aligned}$$

provided that C is big enough. Plugging this last equation in eq. (12) we get

$$\begin{aligned}
\mathbb{P}\left(R^{(k_I)} < T\right) &\geq 0.9 \sum_{w=6}^{n} \mathbb{P}\left(\mathrm{wt}(k_I) = w\right) \\
&= 0.9(1 - \mathbb{P}\left(\mathrm{wt}(k_I) < 6\right))
\end{aligned}$$

Using the assumption that $s \geq 17$, we have that $|I| \geq 17$ and using the assumption that k_I is uniform in $\mathbb{F}_2^s$ we can easily show that

$$\mathbb{P}\left(\mathrm{wt}(k_I) < 6\right) < 0.1.$$

Hence we finally have that

$$\begin{aligned}\mathbb{P}\left(R^{(k_I)} < T\right) &\geq 0.9^2 \\ &= 0.81.\end{aligned}$$

Clearly from the study of these last two case the advantage of the distinguisher is greater than some constant.

F Fast computation of the score function

The goal of this section is to prove proposition 1, that is, given a list L of couples of the form $(x, y) \in \mathbb{F}_2^n \times \mathbb{F}_2$, and some parameters n, s, α, β, our goal is to make a procedure computing for all $z \in \mathbb{F}_2^s$ the scores

$$R^{(z)} := \sum_{(x,y)\in L} (-1)^y \eta_{(\alpha,\beta,n-s)}\left(\langle z, x_J\rangle\right), \qquad \text{where } J := \{0,\ \ldots, s-1\ \}$$

in total time $\mathcal{O}\left(s2^s + |L|\right)$. To simplify further expression we denote simply for all $m \in \mathbb{Z}_6$

$$\eta(m) := \eta_{(\alpha,\beta,n-s)}\left(m\right).$$

Our strategy for computing this function first make appear the Fourier transform $\widehat{\eta}$ of η. This is a standard technique and will allow making visible the character of the group $\mathbb{Z}_6^s$, which we will then use to exhibit recurrence relations allowing the efficient computation of the $R^{(z)}$'s.

Lemma 6. *First, let us define for any $s \in \mathbb{N}$ and any $d \in \mathbb{Z}_6$ and any function f from $\mathbb{F}_2^s$ to $\mathbb{C}$ the quantity $M_{(f,d)}$ defined as*

$$M_{(f,d)}\left(z\right) := \sum_{a\in\mathbb{F}_2^s} f(a)\omega^{d\langle a,z\rangle}.$$

We have that

$$R^{(z)} = \frac{1}{6}\sum_{d\in\mathbb{Z}_6} \widehat{\eta}\left(d\right) M_{(f,d)}\left(z\right)$$

where $\omega := e^{-2i\pi/6}$ and

$$\forall a \in \mathbb{F}_2^s, \qquad f(a) := \sum_{(x,y)\in L : x_J = a} (-1)^y.$$

Proof. Recalling that by definition

$$R^{(z)} := \sum_{(x,y)\in L} (-1)^b \eta\left(\langle x_J, z\rangle\right). \tag{14}$$

Using the definition of the Fourier Transform $\widehat{\eta}$ of η we have that

$$\eta\left(\langle x_J, z\rangle\right) = \frac{1}{6}\sum_{d\in\mathbb{Z}_6} \widehat{\eta}\left(d\right)\omega^{d\langle x_J, z\rangle}$$

Plugging this last equation into eq. (14) yields the lemma.

The algorithm for computing $M_{f,d}$. Writing $z \in \mathbb{F}_2^s$ as $z_1||z'$ where $z_1 \in \mathbb{F}_2$ and $z' \in \mathbb{F}_2^{s-1}$, our algorithm leverages the following recurrence relation.

$$\begin{aligned} M_{f,d}(z_1||z') &= \sum_{a \in \mathbb{F}_2^s} f(a)\omega^{d\langle a,z\rangle} \\ &= \sum_{a' \in \mathbb{F}_2^{s-1}} f(0||a')\omega^{d\langle a',z'\rangle} + \omega^{dz_1} \sum_{a \in \mathbb{F}_2^{s-1}} f(1||a')\omega^{d\langle a',z'\rangle} \\ &= M_{f_0,d}(z') + \omega^{dz_1} M_{f_1,d}(z') \end{aligned}$$

where

$$\forall a' \in \mathbb{F}_2^{s-1}, b \in \mathbb{F}_2, \qquad f_b\left(a'\right) := f\left(b||a'\right).$$

Namely our algorithm is recursive and given as follows.

Algorithm 4 Fast Computation of $M_{f,d}$ when $f : \mathbb{F}_2^s \longrightarrow \mathbb{R}$ and $d \in \mathbb{Z}_6$

procedure ComputeM(f, d, s)
  $\omega \leftarrow \exp\left(-2i\pi/6\right)$
  if $s = 0$ **then**
    Return f
  $f_0(a') \leftarrow f(0||a')$ for all $a \in \mathbb{F}^{s-1}$
  $M_{f_0,d} \leftarrow$ ComputeM($f_0, d, s-1$)
  $f_1(a') \leftarrow f(1||a')$ for all $a \in \mathbb{F}^{s-1}$
  $M_{f_1,d} \leftarrow$ ComputeM($f_1, d, s-1$)
  $M_{f,d}\left(z\right) \leftarrow M_{f_0,d}(z') + \omega^{dz_1} M_{f_1,d}(z')$ for all $(z_1, z') \in \mathbb{F}_2 \times \mathbb{F}_2^{s-1}$
  Return the function $z \in \mathbb{F}_2^s \mapsto M_{f,d}(z) \in \mathbb{C}$

It is easy to show by induction that the time and memory complexity of ComputeM(f, d, s) is given by $\mathcal{O}\left(s2^s\right)$.

The algorithm the score function. We are finally able to write the procedure computing efficiently the whole score function. It is described in Algorithm 5 and its correctness follows from Lemma 6.

Algorithm 5 Fast Score Computation

procedure ComputeScoreFunction(L, α, β, n, s)
  $\eta(m) \leftarrow \eta_{(\alpha,\beta,n-s)}, \quad \forall m \in (\mathbb{Z}_6)^2$
  Compute Fourier Transform $\widehat{\eta}$ of η
  $R^{(z)} \leftarrow 0, \quad \forall z \in \mathbb{F}_2^s$
  $f(a) \leftarrow \sum_{(x,y)\in L: x_{\{0,\ldots,s-1\}}=a} (-1)^y$ for all $a \in \mathbb{F}_2^s$
  for $d \in \mathbb{Z}_6$ **do**
    $M_{f,d} \leftarrow$ ComputeM(f, s, d)
    for $z \in \mathbb{F}_2^s$ **do**
      $R^{(z)} \leftarrow R^{(z)} + \frac{\widehat{\eta}(d)}{6} M_{f,d}(z)$
  Return R

The proof of Proposition 1 follows from the algorithm.