Incorporating SIS Problem into Luby-Rackoff Cipher

Yu Morishima* and Masahiro Kaminaga †

Abstract

With the rise of quantum computing, the security of traditional cryptographic systems, especially those vulnerable to quantum attacks, is under threat. While public key cryptography has been widely studied in post-quantum security, symmetric-key cryptography has received less attention. This paper explores using the Ajtai-Micciancio hash function, based on the Short Integer Solution (SIS) problem, as a pseudorandom function in the Luby-Rackoff cipher. Since lattice-based problems like SIS are believed to resist quantum algorithms, this approach provides the potential for a quantum-resistant block cipher. We also propose a novel statistical method based on the Generalized Extreme Value distribution to evaluate the number of secure rounds and resistance to differential cryptanalysis.

Keywords: Luby-Rackoff cipher; Short integer solution problem; Differential cryptanalysis.

1 Introduction

Advances in quantum computing threaten conventional cryptographic systems, prompting active research into post-quantum cryptography designed to resist quantum attacks. Although public key cryptography has been studied extensively, symmetric-key cryptography has yet to receive much attention. This lack of focus is especially significant as symmetric-key cryptography is essential in lightweight, resource-efficient systems. Quantum attacks on symmetric-key algorithms [1][2] highlight the urgent need to secure these widely implemented systems.

In this context, Luby and Rackoff showed that the Feistel cipher is secure against chosen plaintext attacks (CPA) with three rounds and secure against chosen ciphertext attacks (CCA) with four rounds[3]. This security is achieved when the Feistel cipher is constructed using an ideal pseudorandom function (PRF). This result highlights the critical role of PRFs in ensuring the security of symmetric-key cryptography, particularly in Feistel structures. The performance and security of Feistel ciphers heavily depend on the design of the"F function," which is directly linked to the quality of the underlying PRF.

*Faculty of Engineering, Tohoku Gakuin University
†Graduate School of Engineering, Tohoku Gakuin University

PRFs are crucial not only for Feistel structures but also for symmetric-key cryptosystems. The Goldreich-Goldwasser-Micali (GGM) method [4] and the synthesizer technique [5] are well-known for constructing secure PRFs. While the GGM method offers strong theoretical security, the synthesizer technique supports efficient parallelization. Additionally, extensions to these methods have been proposed, including the use of lattice-based primitives, as seen in the synthesizer approach[6].

Based on hard lattice problems, Lattice cryptography is a leading candidate for post-quantum cryptography. It originated with Ajtai's seminal work[7], which showed that hard lattice problems in the worst case can be used to construct a one-way function with average computational hardness. This result is based on the Short Integer Solution (SIS) problem, a foundational lattice-based problem. Micciancio extended this to develop a collision-resistant hash function[8].

Learning with Errors (LWE) and SIS are known as the main lattice problems that form the basis of lattice cryptography. LWE and SIS have a duality[9]. If the solution vector in the SIS is identified with the error vector in the LWE, the two are reduced to the Closest Vector Problems (CVP) of the same class[10].

SIS is used for hash function construction[11] and digital signatures[12], and LWE is used for public key and homomorphic encryption[13][14]. A configuration using LWE has been cited as an example of applying the lattice problem to symmetric-key cryptography, and many configurations using LWE have been considered in existing research. On the other hand, studies on SIS have received little attention. However, SIS does not require a precise design of the error distribution in LWE and is simple to implement, making it suitable for theoretical analysis.

A variant of the synthesizer that incorporates the LWE through Learning with Rounding (LWR) as a primitive has been proposed [15], which, despite its benefits, faces challenges related to the synchronous execution of parallel processes and increased communication overhead. As an extension of this research, a further refined version of the synthesizer has been proposed [16], which enhances efficiency but requires substantially larger key sizes. While these are essential theoretical results, constructing symmetric-key cryptosystems based on SIS without these trivial constructions remains an open problem.

A key question arises: If the Ajtai-Micciancio hash function were adopted as the PRF, would the Luby-Rackoff cipher retain its resistance against chosen plaintext attacks? The cryptographic significance of exploring this approach is as follows: By employing the Ajtai-Micciancio hash function, a block cipher rooted in the SIS problem is introduced. Given that lattice-based problems like SIS are believed to be resistant to quantum algorithms, the Luby-Rackoff cipher may offer the potential to function as a quantum-resistant block cipher. Studies have demonstrated quantum attacks on Feistel structures, making this an important and timely problem[17]. To verify the security of the Luby-Rackoff cipher under quantum threats, it is crucial to evaluate its security against CPA. One of the most fundamental attack methods for CPA is differential cryptanalysis, which analyzes the differential characteristics of the cipher. However, directly performing a precise mathematical evaluation of differential characteristics is generally difficult. To address this, we propose a statistical approach.

This paper presents a method for evaluating the differential cryptanalysis resistance using a combination of Nyberg's results and the Generalized Extreme Value (GEV) distribution. By leveraging these statistical tools, we estimate the number of secure rounds required to resist differential attacks. This approach provides a novel and effective way to assess the differential characteristics, a critical factor in determining the CPA security of the cipher. To the best of our knowledge, no prior work has combined these methods for such an analysis, making this a valuable contribution to the field.

2 Preliminaries

2.1 Lattice problems

A lattice is a set of all integer linear combinations of n linearly independent column vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n$. The lattice $\mathcal{L}(\mathbf{B})$ generated by these vectors can be represented as $\mathcal{L}(\mathbf{B}) = \{\mathbf{B}\boldsymbol{x} \mid \boldsymbol{x} \in \mathbb{Z}^n\}$ by matrix $\mathbf{B} = \begin{pmatrix}\boldsymbol{b}_1 & \boldsymbol{b}_2 & \cdots & \boldsymbol{b}_n\end{pmatrix}$, where $\boldsymbol{x}$ denotes a column vector. In the following, vectors are assumed to be column vectors, and $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z}$ denotes the integers modulo q. The successive minima of the lattice are defined as follows.

Definition 1. *(Successive Minima) The successive minima $\lambda_1, \ldots, \lambda_n$ of the rank n lattice $\mathcal{L}$ are defined as follows: The i-th minimum $\lambda_i(\mathcal{L})$ is*

$$\lambda_i(\mathcal{L}) = \inf\{r \mid \dim(\mathrm{span}(\mathcal{L} \cap B(r))) \geq i\}. \tag{1}$$

Here, we denote the closed ball of centered at the origin and radius r as $B(r)$.

Lattice problems can be used in cryptography to discuss computational hardness and security. For example, the CVP, which finds a vector in $\mathcal{L}(\mathbf{B})$ closest to a given target vector $\boldsymbol{t} \notin \mathcal{L}(\mathbf{B})$, and the Shortest Vector Problem (SVP), which finds the shortest nonzero vector in $\mathcal{L}(\mathbf{B})$. The Shortest Independent Vectors Problem (SIVP) is another example of a lattice problem, with its computational hardness stemming from the difficulty of identifying a set of linearly independent vectors.

The lattice problems serve as the foundation for constructing strong ciphers. By using the technique of reducing worst case to average case hardness, we can build a cipher that exhibits strong resistance to attacks on average. SIS and LWE are exemplary problems that demonstrate such average resilience. The following provides a formal description of lattice problems relevant to this paper.

Definition 2. *(SIS) Given a uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a real number $\beta \geq 1$, find a nonzero integer vector $\boldsymbol{x} \in \mathbb{Z}^m$ such that $\mathbf{A}\boldsymbol{x} = \mathbf{0} \in \mathbb{Z}_q^n$ and $\|x\| \leq \beta$.*

Definition 3. *(SVP) Given a lattice $\mathcal{L}(\mathbf{A})$, find the shortest nonzero vector $\boldsymbol{v}$ in $\mathcal{L}(\mathbf{A})$. The parameter γ in the "γ-approximate SVP"(SVP$_\gamma$ for short) refers to the approximation factor, where the algorithm finds a vector $\boldsymbol{v}$ such that $\|\boldsymbol{v}\| \leq \gamma\lambda_1$, where λ_1 is the norm of the shortest nonzero vector in $\mathcal{L}(\mathbf{A})$.*

γ in SVP$_\gamma$ is a function of rank n of the lattice matrix. $\gamma = \sqrt{n}$ is called the Minkowski's bound, and SVP is known to have a nonzero solution. The LLL lattice reduction algorithm[18] can solve SVP$_\gamma$ where $\gamma = 2^{(n-1)/4}$ in polynomial time. If there is no algorithm to solve the SVP in probabilistic polynomial time, the SIS cannot be solved in probabilistic polynomial time either[7].

Definition 4. *($SIVP_\gamma$) Given a lattice $\mathcal{L}$ of rank n, find n linearly independent vectors* $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ *such that* $\max_i \|\boldsymbol{v}_i\| \leq \gamma\lambda_n(\mathcal{L})$.

2.2 Collision-resistant hash function family

Ajtai proposed a hash function family based on a computationally hard problem on a random lattice.

Definition 5. *(Ajitai's hash function family[7]) For $m > n\log_2 q$, Ajtai's hash function family f is defined as*

$$f(\boldsymbol{x}) = \mathbf{A}\boldsymbol{x} \bmod q, \tag{2}$$

where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ is uniformly selected at random and $\boldsymbol{x} \in \{0,1\}^m$.

This function has the parameters $n, m, q \in \mathbb{Z}^+$, where m and q are defined as functions of n. By considering the appropriate parameters and lattice problems, we can evaluate the computational hardness of this hash function. Ajtai demonstrated that this function can be a one-way hash function. These results indicate that the various computational hardness aspects of this function can be reduced to the average computational hardness of SIS. The average case hardness in lattice problems refers to the difficulty in solving these problems when the input is randomly sampled. The worst case hardness addresses the difficulty of solving the most challenging instances of lattice problems. There are many results regarding the selection of parameters and problems. The worst case hardness can be reduced within factor $O(\beta\sqrt{n})$ to the average hardness of the SIS with β for $q \leq \beta\omega(\sqrt{n\log n})$[19],[20], where $h(n) = \omega(g(n))$ implies that for any constant $c > 0$, $h(n)$ will eventually exceed $c \cdot g(n)$ as n increases.

Micciancio demonstrated that taking advantage of the computational hardness of SIS makes it possible to construct a family of collision-resistant hash functions.

Theorem 1. *(Collision-resistant hash function family[8]) For any sufficiently large polynomial q, if there exists no polynomial time algorithm for solving $SIVP_\gamma$ with $\gamma = O(n)$, which is almost linear in the rank of the lattice, then the hash function family defined in (2) is collision-resistant.*

Here, a large polynomial can be, for instance, chosen as n^3 or 2^n. For a more detailed discussion, please refer to [21].

The worst case computational hardness of SIVP_γ with $\gamma = O(n)$ is reduced to SIS average computational hardness with $q = \Omega(n^2)$, $\beta = O(\sqrt{m})$, $m \approx n\log q$ where $h(n) = \Omega(g(n))$ if there are constants $c > 0$ and n_0 such that $0 \leq c \cdot g(n) \leq h(n)$ for all $n \geq n_0$. This indicates that $g(n)$ is a lower bound on $h(n)$.

The fact that $f(\boldsymbol{x})$ is a collision-resistant hash function implies that the probability $p(m)$ for finding a pair $\boldsymbol{x}, \boldsymbol{x}'(\boldsymbol{x} \neq \boldsymbol{x}')$ such that $\mathbf{A}\boldsymbol{x} = \mathbf{A}\boldsymbol{x}'$ can be proven to be negligible with respect to m using a probabilistic polynomial time algorithm. In this context, negligible implies that $p(m)$ is satisfied $p(m) \leq 1/\mathrm{poly}(m)$ for sufficiently large m and any positive polynomial $\mathrm{poly}(\cdot)$. To construct a concrete hash function, it is necessary to specify q, m, and n, and following reference [8], we choose $q = 2^n$ and $m = 2n^2$ as reasonable values.

2.3 Luby-Rackoff cipher

A Luby-Rackoff cipher is a global structure for building block ciphers, like DES[3], and is based on a Feistel network. First, the input data is divided into halves $\boldsymbol{L}_0$ and $\boldsymbol{R}_0$, and $\boldsymbol{L}_0$ is scrambled by F, which is a nonlinear function of the input half data and round key $\boldsymbol{K}_1$ and EXORed with $\boldsymbol{R}_0$. The ciphertext is generated by executing the same round operation N times with round keys $\boldsymbol{K}_1, \boldsymbol{K}_2, \cdots, \boldsymbol{K}_N$.

The F function determines the strength of the cipher. Using the Luby-Rackoff construction with a PRF family, a class of block ciphers secure against chosen plaintext attacks can be constructed using PRFs. It is known that a family of PRFs can be constructed using a hash function with a one-way property[22]. One-wayness and collision-resistance are different concepts in computational complexity theory. However, the requirements for relaxed collision-resistance are known to be harder than the one-wayness[23]. Therefore, a secure Feistel cipher can be constructed using a collision-resistant hash function.

2.4 Differential cryptanalysis

Differential cryptanalysis is a chosen plaintext attack, a practical attack method against block ciphers[24]. Differential cryptanalysis uses the input plaintext pair $\boldsymbol{X}, \boldsymbol{X}'$ and their difference $\Delta\boldsymbol{X} = \boldsymbol{X} \oplus \boldsymbol{X}' \neq \boldsymbol{0}$. When the attacker can control the pairs, the round key is extracted by observing the bias of the difference $\Delta\boldsymbol{Y} = \boldsymbol{Y} \oplus \boldsymbol{Y}'$ of the output pair $\boldsymbol{Y}, \boldsymbol{Y}'$.

The number of plaintext and ciphertext pairs required for a successful differential cryptanalysis attack is proportional to the reciprocal probability of $\Delta\boldsymbol{Y}$ for the input difference $\Delta\boldsymbol{X}$. Therefore, the higher the probability of $\Delta\boldsymbol{Y}$ is, the easier the attack will be successful, and the more uniformly distributed the probability of $\Delta\boldsymbol{Y}$, the more difficult the attack will be. In other words, the security of a block cipher against differential cryptanalysis is evaluated by the maximum value of the probability of $\Delta\boldsymbol{Y}$, and the plaintext input difference $\Delta\boldsymbol{X}$ in N rounds. The maximum value P_N of the probability of the ciphertext output difference $\Delta\boldsymbol{Y}$ is defined by

$$P_N = \max_{\Delta\boldsymbol{X} \neq \boldsymbol{0}, \Delta\boldsymbol{Y}} P(\Delta\boldsymbol{Y} | \Delta\boldsymbol{X}), \tag{3}$$

where $P(\Delta\boldsymbol{Y}|\Delta\boldsymbol{X})$ denotes the conditional probability of event $\Delta\boldsymbol{Y}$ occurring for a given $\Delta\boldsymbol{X}$. (3) is called the maximum differential probability.

2.5 Extreme value distributions

Extreme value distributions describe the limiting distributions for the minimum or maximum independent random variables from the same distribution. $X_1, \ldots, X_n, \ldots$ be a sequence of independent and identically distributed random variables with a cumulative distribution function $F(x)$ and let $M_n = \max\{X_1, \ldots, X_n\}$ denote the maximum. The distribution of the maximum is given by $P(M_n \leq x) = P(X_1 \leq x) \cdots P(X_n \leq x) = F(x)^n$. We do not see the distribution of M_n for an unknown F, but as $n \to \infty$, we can find its limit distribution. The limit cumulative distribution function $G(x)$ of the extreme distribution is described by the generalized extreme value (GEV) distribution[25]:

$$G(x) = \exp\left\{-\left[1 + \xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}}\right\}, \tag{4}$$

defined on $\{x : 1 + \xi(x-\mu)/\sigma > 0\}$, where $\mu \in \mathbb{R}$, $\sigma > 0$ and $\xi \in \mathbb{R}$. This distribution has three parameters: μ represents location, σ scale, and ξ shape. Among these, depending on the value of shape parameter ξ, it can be divided into the three distributions corresponding to Gumbel at $\xi = 0$, Fréchet at $\xi > 0$, and Weibull at $\xi < 0$.

Theorem 2. *(The extreme value trinity theorem) There exist sequences of constants $a_n > 0$ and $b_n \in \mathbb{R}$ such that for $M_n^* = (M_n - b_n)/a_n$, $P(M_n^* \le x) \to G(x)$ as $n \to \infty$.*

We will represent the distribution of the maximum differential probability of the output of S-boxes in GEV.

3 Lattice-based Feistel cipher

A method using GGM[4] or a synthesizer[5] is known to construct a function family with pseudorandomness from a family of hash functions with one-wayness. However, these methods require processing delays due to circuit depth and repetitive processing in implementation; therefore, another approach is desirable. In this study, we propose a strategy based on Feistel construction. Since the Feistel cipher can always decrypt any F function, it is possible to construct a block cipher using a family of hash functions that is as good as the F function. This study presents the construction of a lattice-based Feistel cipher (LBF) using a family of hash functions as the F function. Fig. 1 shows the structure of the LBF round function. After m-bits input, the plaintext $\boldsymbol{X}$ is divided into $m/2$-bits $\boldsymbol{L}_0$ and $\boldsymbol{R}_0$ and input to the first-round function. This process repeats N rounds to generate the ciphertext $\boldsymbol{Y}$.

The F function in the round function consists of an expansion permutation E, EXOR with the round key $\boldsymbol{K}_i$, and the S-box. The extended permutation E concatenates the bit strings represented by $||$, corresponding to the expansion permutation in DES[24]. Here, E has a simple structure, and while it may appear overly simplistic compared to DES, there is no need to complicate E because $\mathbf{A}$ is random in LBF.

The hash function family $f(\boldsymbol{x})$ composed of (2), is used for the S-box (Fig. 2). For the input $\boldsymbol{x} \in \{0,1\}^m$ to the S-box, the output $\tilde{\boldsymbol{y}} \in \mathbb{Z}_q^n$ of $f(\boldsymbol{x})$, and by encoding $\tilde{y}_i \in \mathbb{Z}_q$ of $\tilde{\boldsymbol{y}}$ into a binary expression for each of $i = 1, 2, \ldots, n$ and concatenating it, the S-box output $\boldsymbol{y} \in \{0,1\}^{\frac{m}{2}}$ is obtained. The selection of round keys is arbitrary as long as the period is long enough to assume that they are uniformly distributed.

While PRFs can be constructed from one-way functions, collision-resistant functions are employed in the LBF. This is due to the fact that even one-way functions might lead to the leakage of round key information if a collision occurs. Consider when a pair of inputs to an S-box, $\boldsymbol{x}$ and $\boldsymbol{x}'$ (where $\boldsymbol{x} \neq \boldsymbol{x}'$), results in $\mathbf{A}\boldsymbol{x} = \mathbf{A}\boldsymbol{x}'$. Let the input to the round function be $\boldsymbol{L} = \boldsymbol{L}_j || \boldsymbol{L}_j$ and let the input pair of $\boldsymbol{L}$ be $\boldsymbol{L}'$. In this case, for the i-th bit of the vectors $\boldsymbol{z}$ and $\boldsymbol{w}$, the EXOR operation satisfies $z_i \oplus w_i = z_i + w_i - z_i w_i$.

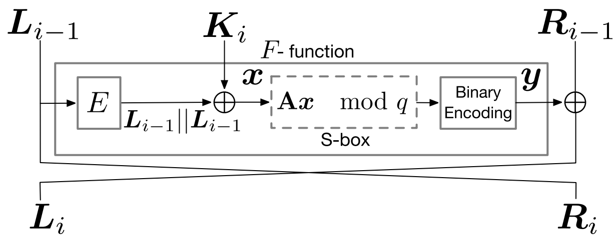

Figure 1: Round function of LBF

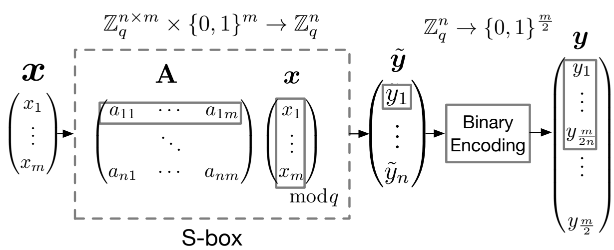

Figure 2: S-box and binary encoding of LBF

Here, we obtain the followings for the input pair $\boldsymbol{x} = \boldsymbol{K} \oplus \boldsymbol{L}$ and $\boldsymbol{x}' = \boldsymbol{K} \oplus \boldsymbol{L}'$, therefore $\mathbf{A}(\boldsymbol{x} - \boldsymbol{x}') = \mathbf{0}$.

Thus, for the i-th bit of $\boldsymbol{x} - \boldsymbol{x}'$, we derive

$$x_i - x'_i = k_i \oplus L_i - k_i \oplus L'_i \quad (5)$$
$$= (L_i - L'_i)(1 - k_i) = 0. \quad (6)$$

If $L_i - L'_i \neq 0$, we can determine $k_i = 1$. Therefore, a secure Feistel cipher can be constructed by using a collision-resistant hash function.

When the hash function family used in the S-box is collision-resistant, the probability of finding a pair of inputs such that $\mathbf{A}\boldsymbol{x} = \mathbf{A}\boldsymbol{x}'$ is negligible. There are various ways to determine $\mathbf{A}$; here, we consider a family where $\mathbf{A}$ is chosen randomly for each encryption but remains fixed across rounds. With the LBF constructed in this manner, the hash function that is difficult to invert is used as a large S-box, making differential cryptanalysis challenging. Moreover, this design allows flexible construction of ciphers with different block sizes in a single structure.

4 Differential cryptanalysis of LBF

In this section, we evaluate LBF's security against differential cryptanalysis. The maximum differential probability in Feistel ciphers determines the system's security, even with two or more rounds. While brute-force search can find this probability for small block sizes or few rounds, it is impractical for large block sizes and many rounds in real systems. Therefore, theoretical analysis is needed, and the following Theorem 3 from [26] provides the basis for estimating differential probabilities.

Theorem 3. *In Feistel cipher, when the round keys are uniformly and independently selected, an upper bound of the maximum differential probability P_N when $N \geq 4$ is given by the following using the maximum differential probability $P_{\max}$ of one round:*

$$P_N \leq 2P_{\max}^2. \tag{7}$$

(7) means that the security against differential cryptanalysis can be evaluated using the maximum differential probability of the round function, a result based on the property that the key selection is uniform and independent.

Below, we evaluate the maximum differential probability $P_{\max}$ of the round function of the LBF. Regarding the input $\boldsymbol{X} \in \{0,1\}^m$ and output $\boldsymbol{Y} \in \{0,1\}^m$ of a round function, if $\boldsymbol{X} = \boldsymbol{X}_L||\boldsymbol{X}_R$ and $\boldsymbol{Y} = \boldsymbol{Y}_L||\boldsymbol{Y}_R$ are blocks divided into $m/2$-bits, the following holds between the input difference $\Delta\boldsymbol{X}$ and the output difference $\Delta\boldsymbol{Y}$ of the round function:

$$\Delta\boldsymbol{Y} = \Delta\boldsymbol{Y}_L||\Delta\boldsymbol{Y}_R = \Delta\boldsymbol{y} \oplus \Delta\boldsymbol{X}_R||\Delta\boldsymbol{X}_L, \tag{8}$$

where $\Delta\boldsymbol{y} \in \{0,1\}^{\frac{m}{2}}$ is the output difference of the S-box (see Fig.2). Since $\Delta\boldsymbol{Y}_R = \Delta\boldsymbol{X}_L$ and the attacker can control the input difference, maximizing the probability of $\Delta\boldsymbol{Y}_L$ maximizes the probability of $\Delta\boldsymbol{Y}$. Note that since the maximum probability of $\Delta\boldsymbol{Y}_L$ is independent of $\Delta\boldsymbol{X}_R$, the maximum probability of $\Delta\boldsymbol{Y}_L$ is determined by $\Delta\boldsymbol{y}$. If the round key $\boldsymbol{K}_i \in \{0,1\}^m$ can be regarded as a uniform random, the S-box input $\boldsymbol{x}$ can also be regarded as a uniform one. Therefore, the maximum probability of $\Delta\boldsymbol{Y}_L$ is determined by the input difference $\Delta\boldsymbol{x}$ of the S-box and output difference $\Delta\boldsymbol{y}$ of the S-box. Furthermore, if the binary encoding of the S-box output $\tilde{\boldsymbol{y}}$ is a bijection, that is, the parameter is chosen such that $m = 2n\log_2 q$ holds, then $\boldsymbol{y}$ and $\tilde{\boldsymbol{y}}$ correspond one-to-one. In this case, $P_{\max}$ can be represented using the maximum differential probability of the S-box as follows:

$$P_{\max} \quad = \quad \max_{\Delta\boldsymbol{x}\neq\boldsymbol{0},\Delta\boldsymbol{y}} P(\Delta\boldsymbol{y}|\Delta\boldsymbol{x}, \mathbf{A}), \tag{9}$$

where $P(\Delta\boldsymbol{y}|\Delta\boldsymbol{x}, \mathbf{A})$ denotes the conditional probability of event $\Delta\boldsymbol{y}$ occurring for given $\Delta\boldsymbol{x}$ and $\mathbf{A}$.

In the following, the maximum differential probability of the S-box output of the LBF is theoretically derived. It is well known that the sum of uniformly distributed variables tends to a normal distribution. However, an exact distribution of the sum of discrete uniform distributions, when folded by modulo-q, cannot be directly derived. First, we present the following lemma for the uniform random variable used to construct the hash function to derive the S-box output difference distribution.

Lemma 4. *Let a_i for $i = 1, 2, \dots, n$ be independent and identically distributed random variables that obey a discrete uniform distribution over $\mathbb{Z}_q$. Then, the value of s_q, as defined by the following, obeys a discrete uniform distribution over $\mathbb{Z}_q$.*

$$s_q = \sum_{i=1}^{n} a_i \bmod q. \tag{10}$$

Proof. Consider n-tuple $\boldsymbol{a} = (a_1, a_2, \cdots, a_n)$ of independent uniform random variables $a_i \in \mathbb{Z}_q (i = 1, 2, \cdots, n)$. Since $\boldsymbol{a}$ is uniformly distributed over $\mathbb{Z}_q^n$, to find $P(s_q = k)$, it is sufficient to find the number of $\boldsymbol{a}$ such that

$$s := \sum_{i=1}^{n} a_i = l + iq, \tag{11}$$

where $l = 0, 1, \cdots, \lfloor \frac{n(q-1)-k}{q} \rfloor$. Therefore we have

$$\sum_{i=1}^{n-1} a_i = k + lq - a_n. \tag{12}$$

The left-hand side of (12) takes the value of $\{0, 1, \cdots, (n-1)(q-1)\}$, and the value of a_n is uniquely determined for $(n-1)$-tuple $(a_1, a_2, \cdots, a_{n-1})$, which (12) holds.

Subsequently, $\boldsymbol{a}$ for which (12) holds exists as q^{n-1} for given k, q, so that $P(s_q = k) = \frac{q^{n-1}}{q^n} = \frac{1}{q}$. □

Lemma 4 leads to the following theorem regarding the distribution of the S-box output pairs.

Theorem 5. *For a given $\Delta \boldsymbol{x} \in \{0,1\}^m$, let the input pairs be $\boldsymbol{x}, \boldsymbol{x}' \in \{0,1\}^m$, and let the output pairs of the S-box be, $\boldsymbol{y}, \boldsymbol{y}' \in \{0,1\}^{\frac{m}{2}}$. If $\mathbf{A}$ obeys a discrete uniform distribution over $\mathbb{Z}_q^{n\times m}$ and m is equal to $2n\log_2 q$, then the probability that the output pair $\boldsymbol{y}, \boldsymbol{y}'$ is obtained from a given $\Delta \boldsymbol{x}$ obeys a uniform distribution over $\{0,1\}^{\frac{m}{2}} \times \{0,1\}^{\frac{m}{2}}$.*

Proof. For a given input pair $(\boldsymbol{x}, \boldsymbol{x}') \in \{0,1\}^m \times \{0,1\}^m$, let $\Delta \boldsymbol{x} = \boldsymbol{x} \oplus \boldsymbol{x}' \in \{0,1\}^m$. Letting the i-th row of the row vector $\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_n$ of $\mathbf{A}$ be $\boldsymbol{a}_i = \begin{pmatrix} a_{i1} & a_{i2} & \cdots & a_{im} \end{pmatrix}$, the output pair $(\tilde{y}_i, \tilde{y}_i') \in \mathbb{Z}_q \times \mathbb{Z}_q$ of $f(\boldsymbol{x})$ corresponding to $\boldsymbol{a}_i$ can be represented as follows:

$$\begin{aligned} \tilde{y}_i &= \boldsymbol{a}_i \boldsymbol{x} \mod q & (13)\\ \tilde{y}_i' &= \boldsymbol{a}_i (\boldsymbol{x} \oplus \Delta \boldsymbol{x}) \mod q. & (14) \end{aligned}$$

Here, the k-th bit of $\Delta \boldsymbol{x}$ is represented as Δx_k and the set of indices where the bit is 0 or 1 is defined as follows:

$$\begin{aligned} \Delta \mathbb{I}_0 &= \{k \mid \Delta x_k = 0 \ (k = 1, 2, \cdots, m)\} & (15)\\ \Delta \mathbb{I}_1 &= \{k \mid \Delta x_k = 1 \ (k = 1, 2, \cdots, m)\}. & (16) \end{aligned}$$

Using the set of indices $\Delta \mathbb{I}_0$, $\Delta \mathbb{I}_1$ and the k-th bit x_k of $\boldsymbol{x}$, $\tilde{y}_i$ and $\tilde{y}_i'$ can be represented as follows:

$$\begin{aligned} \tilde{y}_i &= \sum_{k\in\Delta\mathbb{I}_0} a_{ik} x_k + \sum_{k\in\Delta\mathbb{I}_1} a_{ik} x_k \mod q & (17)\\ \tilde{y}_i' &= \sum_{k\in\Delta\mathbb{I}_0} a_{ik} x_k + \sum_{k\in\Delta\mathbb{I}_1} a_{ik} (x_k \oplus 1) \mod q. & (18) \end{aligned}$$

Since the elements of $\boldsymbol{a}_i$ are random variables that obey the uniform distribution over $\mathbb{Z}_q$, the first terms of (17) and (18) can be represented as random variables u that obey a discrete uniform distribution over $\mathbb{Z}_q$ from Lemma 4. Also, the second term in (17) is the sum of a_{ik} for k such that $x_k = 1$, and the second term of (18) is the sum of a_{ik} for k such that $x_k = 0$, and since a_{ik} in (1) and (2) do not overlap, their sums are mutually independent.

From Lemma 4, each sum is an independent random variable v, w that obey uniform distribution over $\mathbb{Z}_q$. $\tilde{y}_i, \tilde{y}'_i$ can be represented in the following form using independent random variables u, v, and w that obey discrete uniform distribution over $\mathbb{Z}_q$ as follows:

$$\tilde{y}_i = u + v \mod q \tag{19}$$
$$\tilde{y}'_i = u + w \mod q. \tag{20}$$

Since v, w, and u are independent, $\tilde{y}_i, \tilde{y}'_i$ are independent and uniformly distributed random variables over $\mathbb{Z}_q$.

As each row of the matrix $\mathbf{A}$ is independent, each row of the outputs $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{y}}'$ of the S-box are also independent. Therefore, the random variables $(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}')$ obey uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q^n$. Considering that the binary encoding $\mathbb{Z}_q^n \times \mathbb{Z}_q^n \to \{0,1\}^{\frac{m}{2}} \times \{0,1\}^{\frac{m}{2}}$ is a bijection for $m = 2n \log_2 q$, the S-box output pair $(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}')$ is also a random variable that obeys a uniform distribution over $\{0,1\}^{\frac{m}{2}} \times \{0,1\}^{\frac{m}{2}}$. □

Therefore, if the binary encoding of the LBF is a bijection, the difference of output $\Delta\boldsymbol{y} = \boldsymbol{y} \oplus \boldsymbol{y}'$ obeys the uniform distribution over $\{0,1\}^{\frac{m}{2}}$, then $P_{\max} = 1/2^{\frac{m}{2}}$.

This result and Theorem 5 lead to the estimate of N round maximum differential probability P_N given by

$$P_N \leq 2P_{\max}^2 = \frac{1}{2^{m-1}}. \tag{21}$$

5 Statistical analysis of differential cryptanalysis on LBF instances

In the evaluating cryptography, it is important to analyze a family of functions, but it is also necessary to examine specific instances for practical applications. In this study, we evaluate the typical security of LBF instances against differential cryptanalysis, where typical security refers to the security expected on average when focusing on individual instances in the family. We examine the specific instances and analyze the average properties within the LBF family. In addition, we use extreme value theory by GEV for the approximate model of the S-box to theoretically estimate the maximum differential characteristic probability and the practically secure number of rounds.

5.1 Differential cryptanalysis works well against LBF with small block sizes

We conducted simulations to examine whether LBF is vulnerable to attacks by differential cryptanalysis, focusing on cases with short block sizes. First, to eliminate the uncertainty caused by the random selection and examine the characteristics, we perform a computer simulation of differential cryptanalysis for the block size for which the candidate keys can be brute-force searched. When differential cryptanalysis is attempted on a Feistel cipher, the input difference $\Delta \boldsymbol{X}$ of the plaintexts is controlled, and the round key is estimated based on the output difference $\Delta \boldsymbol{Y}$ of the ciphertexts. For example, if the number of rounds is $N = 1$, the key can be obtained by the following procedure[27].

1. For the selected $\Delta \boldsymbol{X}$, select a pair of plaintext input pairs $\boldsymbol{X}, \boldsymbol{X}' = \boldsymbol{X} \oplus \Delta \boldsymbol{X}$ and find $\Delta \boldsymbol{Y}$.

2. Select a pair of input pairs $\boldsymbol{x}, \boldsymbol{x}'$ to the S-box such that the input difference of the the plaintext is $\Delta \boldsymbol{X}$, and determine $\Delta \boldsymbol{Y}$.

3. For the pairs $\boldsymbol{X}, \boldsymbol{X}', \Delta \boldsymbol{Y}$ obtained in 1) and the pairs $\boldsymbol{x}, \boldsymbol{x}', \Delta \boldsymbol{Y}$ obtained in 2), where $\Delta \boldsymbol{Y}$ is identical, the candidate key $\hat{\boldsymbol{K}}_1$ is derived from the relations $\boldsymbol{x} = \boldsymbol{L}_0 \parallel \boldsymbol{L}_0 \oplus \boldsymbol{K}_1$ and $\boldsymbol{X} = \boldsymbol{L}_0 \parallel \boldsymbol{R}_0$, and add it to the list of candidate keys.

The above procedure is repeated for multiple $\Delta \boldsymbol{X}$ to narrow down the keys. If the candidate keys can be searched for every $\boldsymbol{X}$and $\boldsymbol{x}$, the candidate keys can be narrowed down by taking the intersection of the candidate keys for each $\Delta \boldsymbol{X}$. If the number of rounds is $N = 3$, the candidate keys can be estimated using the same procedure by modifying the relation between the input and output differences to be used. If the input block size m is large, searching for all the candidates becomes difficult. Therefore, it is necessary to modify the candidate keys to estimate them from several randomly selected input pairs.

In the simulation, one instance on $\mathbb{Z}_q^{n \times m}$ that is a full rank matrix is chosen at random as $\mathbf{A}$ of the S-box, and a round key is searched for an instance of LBF for the selected $\mathbf{A}$ according to the differential cryptanalysis procedure described above. For the LBF with parameters $(n, m, q) = (2, 8, 4)$ and $(4, 32, 16)$, we searched for candidate keys by differential cryptanalysis. We found that the correct round key can be identified in a few hours in both cases where the number of rounds $N = 1$ and 3.

Table 1 shows the results of how the number of key candidates decreases for each instance of randomly generated $\mathbf{A}$ with $m = 32$ and $N = 3$ as the number of input differences increases from 1 to 3. The table shows the minimum, maximum, mean, and median number of key candidates for 100 instances. With only one difference, approximately one million possible keys can be found. However, with two differences, the number of possible keys decreases significantly to around a few hundred. With three differences, most instances are able to identify the correct key.

The results show that a linear regression was performed with the equation $y = c + dx$, where y is the base-2 logarithm of the number of key candidates, and x is the number of input differences. The estimated intercept, c, is 29.35747 (standard error: 0.16927, t-value: 173.4, $Pr(> |t|)$: <2e-16), and the slope, d, is -9.89946 (standard error: 0.07836, t-value: -126.3, $Pr(> |t|)$: < 2e-16). The R^2 is 0.9817, indicating a good fit of the regression equation to the results. With each additional difference, the number of candidates decreases to approximately $1/2^d \approx 1/1000$, corresponding to 9.9-bits.

These results show that when the block size m and the number of rounds N are small, the round key of the LBF can be identified using only a few input and output difference

Table 1: Statistical Results of the number of key candidates per number of difference used

# of difference	Minimum	Maximum	Mean	Median
1	587880	8388608	1203716.30	1051008
2	126	4132	397.61	274
3	1	5	1.33	1

pairs by using differential analysis. As observed here, differential cryptanalysis is practical when the block size and number of rounds are small. Therefore, it is necessary to determine block sizes and the number of rounds that differential cryptanalysis cannot solve.

5.2 Number of secure rounds

It is crucial to select a sufficiently large block size and number of rounds when designing block ciphers to ensure the required level of security. In this study, we analyze differential characteristics for randomly selected instances of $\mathbf{A}$ and statistically evaluate the number of secure rounds.

In Feistel ciphers, differential cryptanalysis becomes harder with more rounds, but processing time also increases, making it important to study the security-performance trade-off. While tracking active S-boxes enables efficient attacks on Feistel ciphers with many small S-boxes, this method does not apply to LBF due to its single S-box.

Another approach to the security evaluation of block ciphers with many rounds is differential characteristic probability, which estimates the differential probability of N rounds using the product of the differential probabilities for each round[24].

Definition 6. *Let $\Delta \boldsymbol{X}_{j-1}$ and $\Delta \boldsymbol{X}_j$ be the input and output differences in the j-th round, respectively, and $P(\Delta \boldsymbol{X}_j | \Delta \boldsymbol{X}_{j-1})$ be the conditional probability of the output difference given the input difference. The differential characteristic probability $P_{c,N}$ for N rounds is defined as*

$$P_{c,N} = \prod_{j=1}^{N} P(\Delta \boldsymbol{X}_j | \Delta \boldsymbol{X}_{j-1}), \tag{22}$$

with $\Delta \boldsymbol{X}_0 = \Delta \boldsymbol{X} \neq \mathbf{0}$ at the start.

For each round, the combination of realized values of the input difference $\Delta \boldsymbol{X}_{j-1}$ is called a *path*, and the differential characteristic probability is obtained by searching for the path that maximizes $P_{c,N}$.

When the differential characteristic probability satisfies $P_{c,N} \leq 2^{-m}$, the cipher is considered to be "practically secure" against differential cryptanalysis[28]. The smallest such N is the number of rounds the cipher secures against differential cryptanalysis. For block ciphers satisfying $P_{c,N} \leq 2^{-m}$, an attacker needs plaintext greater than or equal to all possible plaintext patterns to decrypt the cipher with differential cryptanalysis. The practically secure lower bound of $N(\mathbf{A}) = \min\{N \mid P_{c,N} \leq 2^{-m}\}$ is determined only by $\mathbf{A}$. We find the distribution of $N(\mathbf{A})$ by computing the differential characteristic

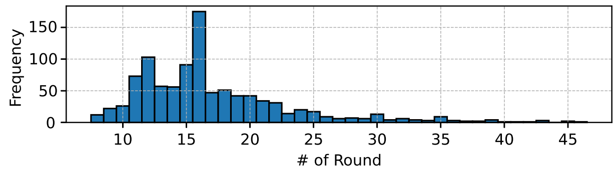

Figure 3: Number of rounds to achieve $P_{c,N} \leq 2^{-m}$

probability for uniformly random $\mathbf{A}$. For an LBF with the parameter $(n, m, q) = (2, 8, 4)$ and fixed round keys, we generate 1000 instances of $\mathbf{A}$ of full rank and determine $N(\mathbf{A})$. For small n, like $n = 2$, certain output differences can appear frequently and even become fixed; however, as n increases, such cases become rare, and in our simulations, no such instances occurred for $n \geq 4$, with their frequency expected to decrease further with larger n. Thus, these rare cases are unlikely to affect the overall analysis when n is sufficiently large.

Fig. 3 depicts the distribution of $N(\mathbf{A})$ obtained by the Monte Carlo simulation. The minimum value of $N(\mathbf{A})$ is 8, the maximum value is 46, and the average is 17.08. The results confirm that the number of secure rounds varies, corresponding to each instance.

5.3 Average properties on S-box output differential

Based on the previous discussion, this section studies the average characteristics of LBF instances. In an ideal S-box, the distribution of the output pairs obeys a uniform distribution. An ideal maximum difference characteristic is that the maximum output difference is small, and the probability of the output difference asymptotically obeys a uniform distribution.

It is difficult to demonstrate directly that the distribution of the maximum differential probability of the LBF approaches a uniform distribution. Using a method based on the generalized extreme value distribution, we propose that the distribution of maximum differential probability of the LBF asymptotically approaches that of an ideal distribution.

We approximate the distribution of the output pairs of the LBF S-box by a folded two-dimensional normal distribution and show that the average output characteristics of the LBF approach the ideal uniform distribution using a generalized extreme value distribution. When selecting an instance of $\mathbf{A}$, each row $\boldsymbol{a}_i \in \mathbb{Z}_q^m (i = 1, 2, \cdots, n)$ is independent, so the distribution of S-box output differences is a joint distribution of the distributions for each row.

First, for the S-box input pair $\boldsymbol{x}$, $\boldsymbol{x}'$ and $\mathcal{J}_{m,q} = \{0, 1, \cdots, m(q-1)\}$, define the S-box output pair $(\hat{y}_i, \hat{y}_i{}') = (\boldsymbol{a}_i \boldsymbol{x}, \boldsymbol{a}_i \boldsymbol{x}') \in \mathcal{J}_{m,q} \times \mathcal{J}_{m,q}$.

Fig. 4 shows the empirical distribution of output pairs $(\hat{y}_i, \hat{y}_i')$ for randomly generated instances of $\boldsymbol{a}_i$ by the Monte Carlo simulation. Note that the empirical distribution depends on the input difference. However, since the components of $\boldsymbol{a}_i$ are selected independently, we only need to consider the Hamming weight h_w of the input difference to obtain the empirical distribution. In Fig. 4, the top, middle, and bottom rows

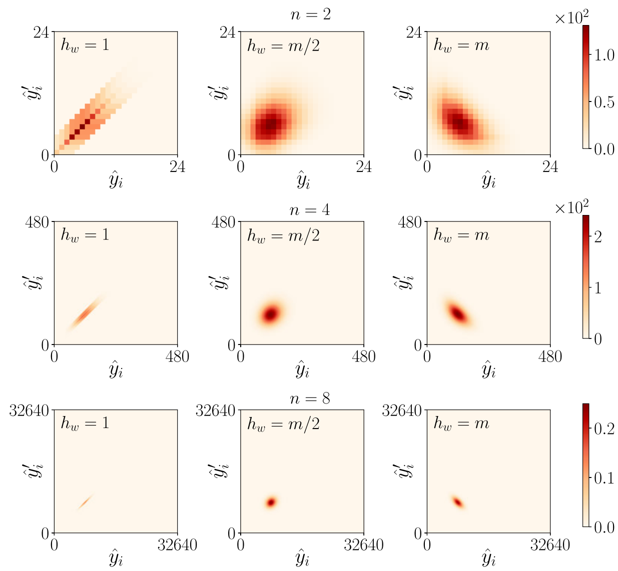

Figure 4: Emprical distribution of $(\hat{y}_i, \hat{y}'_i)$

correspond to $n = 2$, 4, and 8, respectively. From left to right across the columns, the figures correspond to the Hamming weights $h_w = 1$, $m/2$, and m (where m is the block size).

For the cases where $n = 2$, 4, and 8, the empirical distribution was obtained for 100000 instances under each condition, and the frequency was averaged for each instance. In this simulation, the empirical distribution of input $\boldsymbol{x}$ was created using all inputs for $n = 2$. In the case $n = 4$ and 8, the empirical distribution was obtained using 1000000 inputs selected uniformly at random with a fixed input difference $\Delta\boldsymbol{x}$ and a pair of inputs $\boldsymbol{x}' = \boldsymbol{x} \oplus \Delta\boldsymbol{x}$.

Since components of $\boldsymbol{a}_i$ are independent and obey a discrete uniform distribution, the output, which is the sum of them, is close to a normal distribution. As n increases from 2 to 8, the distribution is close to the two-dimensional normal distribution, especially when $h_w = m/2$. Moreover, as the parameter n increases, the number of random variables that obey the uniform distribution increases, so we expect that the distribution is approximately close to the normal distribution.

As n increases, the range of values for $(\hat{y}_i, \hat{y}_i{}')$ grows exponentially, while the concentration ellipse decreases in size, and regions far from the ellipse become rare events. Consequently, the data becomes zero-inflated categorical data. Such data can destabilize the χ^2 value in chi-square tests, making the uniformity test difficult[29]. Therefore, in the following, the joint distribution of the output pairs $(\hat{y}_i, \hat{y}'_i)$ is approximated by a two-dimensional normal distribution. This method is a standard approach for representing bivariate distributions with correlations. Using a folded two-dimensional normal distribution by modulo-q and the GEV, we demonstrate that the maximum differential probability asymptotically approaches an ideal S-box.

First, the distribution of the vector $\hat{\boldsymbol{y}} = (\hat{y}_i, \hat{y}'_i)$ representing the output pair is modeled as a two-dimensional normal distribution as follows:

$$
\begin{aligned}
p(\hat{\boldsymbol{y}}) &= \frac{1}{\sqrt{2\pi}^2 \sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\hat{\boldsymbol{y}} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\hat{\boldsymbol{y}} - \boldsymbol{\mu})\right) && (23) \\
\boldsymbol{\mu} &= \left(\tfrac{m}{4}(q-1), \quad \tfrac{m}{4}(q-1)\right) && (24) \\
\boldsymbol{\Sigma} &= \begin{pmatrix} \sigma^2 & \sigma^2 - \Delta \\ \sigma^2 - \Delta & \sigma^2 \end{pmatrix} && (25) \\
\sigma^2 &= \frac{m(q-1)}{4}\left(\frac{2q-1}{3} + \frac{m^2(q-1)(m-1)}{16}\right) && (26) \\
\Delta &= \frac{1}{12}(2q-1)(q-1)h_w(\Delta \boldsymbol{x}), && (27)
\end{aligned}
$$

where $\boldsymbol{\mu}$ is the mean vector, $\boldsymbol{\Sigma}$ is the covariance matrix, and $h_w(\Delta \boldsymbol{x})$ is the Hamming weight of the input difference. The derivation is obtained directly.

Next, we obtain the distribution folded by modulo-q, which models the distribution of S-box output pairs. By evaluating the maximum probability of this distribution, that is, the frequency of the mode, we can estimate the bias in the differential probability. When a one-dimensional normal distribution is folded by modulo-q, it asymptotically becomes a uniform distribution with a sufficiently small partition width[30]. This result can be applied to a multivariate normal distribution if each dimension is independent. However, it is not easy to extend this result directly because the variables in our approximate model are correlated.

In the following, we use Monte Carlo simulation to obtain the empirical distribution of the random variables $(\tilde{y}_i, \tilde{y}'_i)$, which are the output pairs $(\hat{y}_i, \hat{y}'_i)$ folded by modulo-q. Observing the frequency of the mode of this empirical distribution, we determine the empirical frequency distribution of the mode. By fitting the GEV to the empirical frequency distribution of the mode, we can estimate the maximum density of the S-box output pair.

N_s random output pairs that obey (23) are generated in the simulation. These random variable values are folded by modulo-q to obtain the empirical distribution of $(\tilde{y}_i, \tilde{y}'_i)$, and its frequency of the mode. This process is repeated N_m times to obtain the empirical frequency distribution. The parameters of the GEV were obtained by maximum likelihood estimation. We employed the ismev package of R to estimate the parameters[31]. Additionally, since the ideal S-box output pairs $(\tilde{y}_i, \tilde{y}'_i)$ obey a uniform distribution over $\mathbb{Z}_q \times \mathbb{Z}_q$, we compare its distribution with that of the output given by (23) using Monte Carlo simulation.

In Fig. 5, the top, middle, and bottom rows correspond to $n = 2$, 4, and 8, respectively. The sample size of the output pair is $N_s = 1000000$, the sample size of the frequency is $N_m = 100000$, and the Hamming weights of input differences are $h_w = 1, m/2$, and m. Fig. 5 shows the probability density function of the GEV with the estimated parameters, and Table 2–4 shows the estimated parameters. From the results, for $n = 2$, only the distribution for $h_w = 1$ (dash-dotted line) deviates from the others ($h_w = 4, 8$) and the uniform case. For $n = 4$, the distribution for $h_w = 1$ is slightly offset from the others. For $n = 8$, all the distributions are almost identical, confirming that the

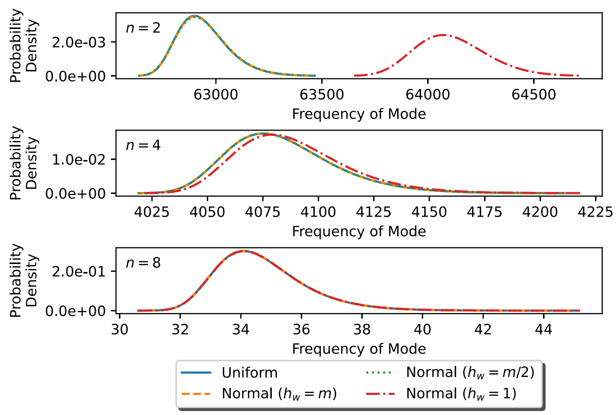

Figure 5: Estimated frequency distribution of mode

deviation decreases as n increases. We also confirmed that the distribution of $n = 2$ and $h_w = 1$ differs from other distributions. However, this discrepancy becomes smaller for larger $n = 4, 8$, and the larger n, the closer the distribution estimated by the normal distribution approximation becomes to that estimated by the uniform distribution.

Tables 2–4 show the estimated GEV parameters, location μ, scale σ, shape ξ, and their standard errors $(SE_\mu, SE_\sigma, SE_\xi)$ obtained by the maximum likelihood estimation. These estimated parameters show that the larger n is, the more asymptotic the normal distribution approximation result is to the uniform distribution characteristic, which is an ideal S-box. Considering the obtained standard error, if one examines the one-sided 95% confidence interval, the shape parameter is sufficiently less than zero, suggesting that the distribution of the output pair $(\tilde{y}_i, \tilde{y}'_i)$ of the S-box modeled by the normal distribution approximation obeys the Weibull distribution.

From the mode of the probability distribution of the S-box output pairs $(\tilde{y}_i, \tilde{y}'_i)$ obtained in this way, we can estimate the maximum differential probability of the LBF in the N round. Let $c(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}')$ be the number of occurrences of the output pair $(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}') \in \mathbb{Z}_q^n \times \mathbb{Z}_q^n$ of the S-box. The maximum differential probability $P_{\max}$ of the S-box is as follows:

$$P_{\max} = \max_{\Delta\tilde{\boldsymbol{y}}} P(\Delta\tilde{\boldsymbol{y}}) = \max_{\Delta\tilde{\boldsymbol{y}}} \sum_{(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}')\, s.t.\, \tilde{\boldsymbol{y}} \oplus \tilde{\boldsymbol{y}}' = \Delta\tilde{\boldsymbol{y}}} \frac{c(\tilde{\boldsymbol{y}}, \tilde{\boldsymbol{y}}')}{2^m},$$

where $\Delta\tilde{\boldsymbol{y}}$ is a formal notation representing the EXOR of $\tilde{\boldsymbol{y}}$ and $\tilde{\boldsymbol{y}}'$ after binary encoding, which represented as $\Delta\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{y}} \oplus \tilde{\boldsymbol{y}}'$. From the independence of each row in $\mathbf{A}$, this can be rewritten in the following form:

$$P_{\max} = \left(\max_{\Delta\tilde{y}_i} \sum_{(\tilde{y}_i, \tilde{y}'_i)\, s.t.\, \tilde{y}_i \oplus \tilde{y}'_i = \Delta\tilde{y}_i} \frac{c(\tilde{y}_i, \tilde{y}'_i)}{q^2} \right)^n, \tag{28}$$

where $c(\tilde{y}_i, \tilde{y}'_i)$ is the number of occurrences of the output pair $(\tilde{y}_i, \tilde{y}'_i)$, and $\tilde{y}_i \oplus \tilde{y}'_i = \Delta\tilde{y}_i$ represents the EXOR after binary encoding. Then, there are q output pairs $(\tilde{y}_i, \tilde{y}'_i)$ whose

Table 2: Estimated parameter ($n = 2$)

	Uniform	Normal ($h_w = m$)	Normal ($h_w = m/2$)	Normal ($h_w = 1$)
μ	62889.9804	62891.2570	62891.1632	64044.1023
SE_μ	0.366293	0.368066	0.371648	0.518169
σ	104.3645	105.4014	106.7196	155.1743
SE_σ	0.258920	0.263640	0.270961	0.353453
ξ	-0.086671	-0.092448	-0.094775	-0.167586
SE_ξ	0.002037	0.002018	0.002064	0.000806

Table 3: Estimated parameter ($n = 4$)

	Uniform	Normal ($h_w = 1$)	Normal ($h_w = m/2$)	Normal ($h_w = m$)
μ	4073.5561	4073.4771	4073.4372	4077.5444
SE_μ	0.073623	0.073703	0.073359	0.075155
σ	21.039072	21.061516	20.952711	21.469652
SE_σ	0.052006	0.051980	0.051752	0.053072
ξ	-0.077223	-0.078088	-0.078389	-0.078177
SE_ξ	0.001978	0.001971	0.001984	0.001981

Table 4: Estimated parameter ($n = 8$)

	Uniform	Normal ($h_w = 1$)	Normal ($h_w = m/2$)	Normal ($h_w = m$)
μ	34.0361	34.0343	34.0496	34.0439
SE_μ	0.004282	0.004259	0.004283	0.004301
σ	1.222491	1.214973	1.220174	1.226664
SE_σ	0.003036	0.003020	0.003043	0.003054
ξ	-0.049566	-0.051149	-0.047329	-0.046985
SE_ξ	0.001993	0.002010	0.002040	0.002016

output differences are $\Delta\tilde{y}_i$, and it is found that the upper bound of (28) can be evaluated using the frequency of the mode N_g for the number of occurrences $c(\tilde{y}_i, \tilde{y}'_i)$ and N_s:

$$P_{\max} = \left(\max_{\Delta\tilde{y}_i} \sum_{(\tilde{y}_i, \tilde{y}'_i) s.t. \tilde{y}_i \oplus \tilde{y}'_i = \Delta\tilde{y}_i} \frac{c(\tilde{y}_i, \tilde{y}'_i)}{q^2} \right)^n \leq \left(q \frac{N_g}{N_s} \right)^n .$$

The upper bound of the maximum differential characteristic probability of N rounds can be obtained as the power of N, and the number of rounds N satisfying the practically secure criterion can be estimated by determining the smallest N satisfying the following:

$$\left(\frac{q N_g}{N_s} \right)^{nN} \leq 2^{-m}. \tag{29}$$

For $N_s > qN_g$, the following can be derived

$$N \geq \frac{m}{\log_2 \frac{N_s}{qN_g}}. \tag{30}$$

According to the estimation of the number of rounds by (30), N required is maximized by the largest N_g. Since $(-\infty, \mu - \frac{\sigma}{\xi}]$ is supported on the negative axis for the GEV shape parameter $\xi < 0$, we consider $N_g = \mu - \frac{\sigma}{\xi}$, which has the largest mode, to estimate the upper bound on the number of rounds required. For $n = 2$, a large value of N_g results in $N_s < qN_g$, making it impossible to evaluate the number of rounds. However, for $n = 4$, $N = 16$, and for $n = 8$, $N = 6$, the upper bound of the required number of rounds can be estimated. To the best of the authors' knowledge, no examples of theoretical evaluation of secure rounds focusing on their probability distribution have been found. As a result, the fact that this approach yields a specific number of secure rounds is particularly noteworthy.

6 Conclusion

This paper evaluates the security of the Luby-Rackoff cipher when a hash function based on the computational hardness of the SIS problem is used as its pseudorandom generator. We derived an upper bound on the maximum differential probability to assess resistance to differential cryptanalysis and identified the required secure rounds for each block size m. Using the GEV distribution, our statistical analysis of S-box output bias shows that secure rounds for block sizes 32 and 128 are 16 and 6, respectively, demonstrating LBF's robustness against differential cryptanalysis. However, extending our method beyond $n = 8$ is challenging due to increased sample and computational demands. Further research is needed to explore alternative approaches and extend the analysis to linear cryptanalysis.

Acknowledgments

This work was supported by the Japan Society for the Promotion of Science KAKENHI Grant Number 20K11817.

References

[1] A. K. Malviya, N. Tiwari, and M. Chawla, "Quantum cryptanalytic attacks of symmetric ciphers: A review," *Computers and Electrical Engineering*, vol. 101, p. 108122, 2022.

[2] X. Dong, B. Dong, and X. Wang, "Quantum attacks on some feistel block ciphers," *Designs, Codes and Cryptography*, vol. 88, pp. 1179–1203, 2020.

[3] M. Luby and C. Rackoff, "How to construct pseudorandom permutations from pseudorandom functions," *SIAM Journal on Computing*, vol. 17, no. 2, pp. 373–386, April 1988.

[4] O. Goldreich, S. Goldwasser, and S. Micali, "How to construct random functions," *Journal of the ACM*, vol. 33, no. 4, pp. 792–807, August 1986.

[5] M. Naor and O. Reingold, "Synthesizers and their application to the parallel construction of Pseudo-Random functions," *Journal of Computer and System Sciences*, vol. 58, no. 2, pp. 336–375, April 1999.

[6] A. Banerjee, C. Peikert, and A. Rosen, "Pseudorandom functions and lattices," in *Advances in Cryptology – EUROCRYPT 2012*. Springer Berlin Heidelberg, 2012, pp. 719–737.

[7] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing*, July 1996, pp. 99–108.

[8] P. Q. Nguyen and B. Vallee, Eds., *The LLL Algorithm: Survey and Applications*. Berlin, Germany: Springer, December 2009.

[9] O. Regev, "The learning with errors problem (invited survey)," in *Proceedings of 2010 IEEE 25th Annual Conference on Computational Complexity*, June 2010, pp. 191–204.

[10] D. Micciancio, "(invited talk) duality in lattice based cryptography," in *Proceedings of 13th International Conference on Practice and Theory in Public Key Cryptography*, May 2010.

[11] O. Goldreich, S. Goldwasser, and S. Halevi, *Collision-Free Hashing from Lattice Problems*, O. Goldreich, Ed. Berlin, Heidelberg: Springer, August 2011.

[12] K. Kajita, G. Ohtake, K. Ogawa, K. Nuida, and T. Takagi, "Short lattice signature scheme with tighter reduction under Ring-SIS assumption," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E106.A, no. 3, pp. 228–240, March 2023.

[13] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM*, vol. 56, no. 6, pp. 1–40, September 2009.

[14] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*. IEEE, October 2011, pp. 97–106.

[15] A. Banerjee, "New constructions of cryptographic pseudorandom functions," Ph.D. dissertation, Georgia Institute of Technology, August 2015.

[16] H. Montgomery, "More efficient lattice prfs from keyed pseudorandom synthesizers," in *Proceedings of Progress in Cryptology – INDOCRYPT 2018*, D. Chakraborty and T. Iwata, Eds. Cham: Springer, December 2018, pp. 190–211.

[17] A. Hosoyamada and T. Iwata, "4-round luby-rackoff construction is a qprp," in *Proceedings of Advances in Cryptology – ASIACRYPT 2019*, S. D. Galbraith and S. Moriai, Eds. Cham: Springer International Publishing, 2019, pp. 145–174.

[18] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, December 1982.

[19] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing – STOC'08*. New York, NY, USA: Association for Computing Machinery, May 2008, p. 197–206.

[20] D. Micciancio and C. Peikert, "Hardness of SIS and LWE with small parameters," in *Proceedings of Advances in Cryptology – CRYPTO 2013*. Springer Berlin Heidelberg, August 2013, pp. 21–39.

[21] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on gaussian measures," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2007.

[22] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby, "A pseudorandom generator from any one-way function," *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1364–1396, January 1999.

[23] N. Bitansky, I. Haitner, I. Komargodski, and E. Yogev, "Distributional collision resistance beyond one-way functions," in *Proceedings of Advances in Cryptology – EUROCRYPT 2019*. Cham: Springer International Publishing, April 2019, pp. 667–695.

[24] E. Biham and A. Shamir, "Differential cryptanalysis of DES-like cryptosystems," *Journal of Cryptology*, vol. 4, pp. 3–72, January 1991.

[25] S. Coles, *An Introduction to Statistical Modeling of Extreme Values*. Springer London, August 2001.

[26] K. Nyberg and L. R. Knudsen, "Provable security against a differential attack," *Journal of Cryptology*, vol. 8, pp. 27–37, December 1995.

[27] D. R. Stinson, *Cryptography: theory and practice*. Boca Raton, Florida, U.S.A.: CRC Press, March 1995.

[28] L. R. Knudsen, *Practically secure Feistel ciphers. In: Fast Software Encryption. FSE 1993. Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1994.

[29] A. Agresti, *Categorical Data Analysis*. John Wiley & Sons, 2012.

[30] T. Suzuki and M. Kaminaga, "A true random number generator method embedded in wireless communication systems," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E103.A, no. 4, pp. 686–694, April 2020.

[31] O. S. functions written by Janet E. Heffernan with R port and R. documentation provided by Alec G. Stephenson., *ismev: An Introduction to Statistical Modeling of Extreme Values*, 2018, r package version 1.42.