Fast Correlation Attack Revisited

Cryptanalysis on Full Grain-128a, Grain-128, and Grain-v1

Yosuke Todo[1], Takanori Isobe[2], Willi Meier[3],
Kazumaro Aoki[1], and Bin Zhang[4,5]

[1] NTT Secure Platform Laboratories, Tokyo 180-8585, Japan
[2] University of Hyogo, Hyogo 650-0047, Japan
[3] FHNW, Windisch, Switzerland
[4] TCA Laboratory, SKLCS, Institute of Software, Chinese Academy of Sciences, Beijing, China
[5] State Key Laboratory of Cryptology, P.O.Box 5159, Beijing 100878, China

Abstract. A fast correlation attack (FCA) is a well-known cryptanalysis technique for LFSR-based stream ciphers. The correlation between the initial state of an LFSR and corresponding key stream is exploited, and the goal is to recover the initial state of the LFSR. In this paper, we revisit the FCA from a new point of view based on a finite field, and it brings a new property for the FCA when there are multiple linear approximations. Moreover, we propose a novel algorithm based on the new property, which enables us to reduce both time and data complexities. We finally apply this technique to the Grain family, which is a well-analyzed class of stream ciphers. There are three stream ciphers, Grain-128a, Grain-128, and Grain-v1 in the Grain family. As a result, we break them all, and especially for Grain-128a, the cryptanalysis on its full version is reported for the first time. Note that our attack is applied to the stream cipher mode of Grain-128a, and strong assumption is required to attack its authentication mode. Since ISO/IEC 29167-13 standardizes only authentication mode, our attack does not affect the practical use of the ISO/IEC standard.

Keywords: Fast correlation attack, Stream cipher, LFSR, Finite field, Multiple linear approximations, Grain-128a, Grain-128, Grain-v1

1 Introduction

Stream ciphers are a class of symmetric-key cryptosystems. They commonly generate a key stream of arbitrary length from a secret key and initialization vector (iv), and a plaintext is encrypted by XORing with the key stream. Many stream ciphers consist of an initialization and key-stream generator. The secret key and iv are well mixed in the initialization, where a key stream is never output, and the mixed internal state is denoted as the initial state in this paper. After the initialization, the key-stream generator outputs the key stream while updating the internal state. The initialization of stream ciphers generally requires much processing time, but the key-stream generator is very efficient.

LFSRs are often used in the design of stream ciphers, where the update function consists of one or more LFSRs and non-linear functions. Without loss of generality, the key-stream generator of LFSR-based stream ciphers can be represented as Fig. 1, where the binary noise e_t is generated by the non-linear function. LFSR-based stream ciphers share the feasibility to guarantee a long period in the key stream.

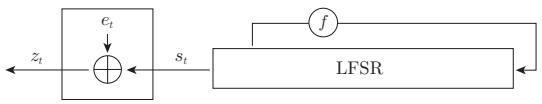

Fig. 1. Model of LFSR-based stream ciphers

A (fast) correlation attack is an important attack against LFSR-based stream ciphers. The initial idea was introduced by Siegenthaler [Sie84], and it exploits the bias of e_t. We guess the initial state $s^{(0)} = (s_0, s_1, \ldots, s_{n-1})$, compute s_t for $t = n, n+1, \ldots, N-1$, and XOR s_t with corresponding z_t. If we guess the correct initial state, highly biased e_t is acquired. Otherwise, we assume that the XOR behaves at random. When we collect an N-bit key stream and the size of the LFSR is n, the simple algorithm requires a time complexity of $N2^n$.

Following up the correlation attack, many algorithms have been proposed to avoid the exhaustive search of the initial state, and they are called as "fast correlation attack." The seminal work was proposed by Meier and Staffelbach [MS89], where the noise e_t is efficiently removed from z_t by using parity-check equations, and s_t is recovered. Several improvements of the original fast correlation attack have been proposed [ZYR90,MG90,CS91,JJ99b,JJ99a,CT00], but they have limitations such as the number of taps in the LFSR is significantly small or the bias of the noise is significantly high. Therefore, their applications are limited to experimental ciphers, and they have not been applied to modern concrete stream ciphers.

Another approach of the fast correlation attack is the so-called one-pass algorithm [CJS00,MFI01], and it has been successfully applied to modern concrete stream ciphers [BGM06,LLP08,ZXM15]. Similarly to the original correlation attack, we guess the initial state and recover the correct one by using parity-check equations. To avoid exhaustive search over the initial state, several methods have been proposed to decrease the number of secret bits in the initial state involved by parity-check equations [CJM02,ZF06]. In the most successful method, the number of involved secret bits decreases by XORing two different parity-check equations. Let $e_t = \langle s^{(0)}, a_t \rangle \oplus z_t$ be the parity-check equation, where $\langle s^{(0)}, a_t \rangle$ denotes an inner product between $s^{(0)}$ and a_t, and we assume that e_t is highly biased. Without loss of generality, we first detect a set of pairs (j_1, j_2) such that the first ℓ bits in $a_{j_1} \oplus a_{j_2}$ are 0, where such a set of pairs is efficiently detected from the birthday paradox. Then, $\langle s^{(0)}, a_{j_1} \oplus a_{j_2} \rangle \oplus z_{j_1} \oplus z_{j_2}$ is also highly biased, and the number of involved secret bits decreases from n to $n-\ell$. Later, this method is generalized by the generalized birthday problem [Wag02]. Moreover, an efficient algorithm was proposed to accelerate the one-pass algorithm [CJM02]. They showed that the guess and evaluation procedure can be regarded as a Walsh-Hadamard transform, and the fast Walsh-Hadamard transform (FWHT) can be applied to accelerate the one-pass algorithm. While the naive algorithm for the correlation attack requires $N2^n$, the FWHT enables us to evaluate it with the time complexity of $N + n2^n$. When the number of involved bits decreases from n to $n-\ell$, the time complexity also decreases to $N + (n-\ell)2^{n-\ell}$. The drawback of the one-pass algorithm with the birthday paradox is the increase of the noise. Let p be the probability that $e_t = 1$, and the correlation denoted by c is defined as $c = 1 - 2p$. If we use the XOR of parity-check equations to reduce the number of involved secret bits, the correlation of the modified equations drops to c^2. The increase of the noise causes the increase of the data complexity.

Revisiting Fast Correlation Attack. In this paper, we revisit the fast correlation attack. We first review the structure of parity-check equations from a new point of view based on a finite field, and the new viewpoint brings a new property for the fast correlation attack. A multiplication between $n \times n$ matrices and an n-bit fixed vector is generally used to construct parity-check equations. Our important observation is to show that this multiplication is "commutative" via the finite field, and it brings the new property for the fast correlation attack.

We first review the traditional wrong-key hypothesis, i.e., we observe correlation 0 when incorrect initial state is guessed. The new property implies that we need to reconsider the wrong-key hypothesis more carefully. Specifically, assuming that there are multiple high-biased linear masks, the traditional wrong-key hypothesis does not hold. We then show a modified wrong-key hypothesis.

The new property is directly useful to improve the efficiency of the fast correlation attack when there are multiple high-biased linear masks. In the previous fast correlation attack, the multiple approximations are only useful to reduce the data complexity but are not useful to reduce the time complexity [BGM06]. We propose a new algorithm that reduces both time and data complexities. Our new algorithm is a kind of the one-pass algorithm, but the technique to avoid the exhaustive search of the initial state is completely different from previous ones. The multiple linear masks are directly exploited to avoid the exhaustive search.

Table 1. Summary of results, where the key-stream generator and initialization are denoted as ksg and init, respectively.

Target		Attack	Assumption	Data	Time	Reference
Grain-128a	ksg	fast correlation attack	-	$2^{113.8}$	$2^{115.4}$	Sect. 5
Grain-128	init	dynamic cube attack	chosen IV	2^{63}	2^{90}	[DGP+11]
	init	dynamic cube attack	chosen IV	$2^{62.4}$	2^{84}	[FWC17]
	ksg	fast correlation attack	-	$2^{112.8}$	$2^{114.4}$	Sect. 6
Grain-v1	ksg	fast near collision attack	-	2^{19}	$2^{86.1}$ †	[ZXM18]
	ksg	fast correlation attack	-	$2^{75.1}$	$2^{76.7}$	Sect. 7

† In [ZXM18], the time complexity is claimed as $2^{75.7}$ but the unit of the time complexity is 1 update function of reference code on software implementation. Here we adjusted the time complexity for the fair comparison.

Applications. We apply our new algorithm to the Grain family, where there are three well-known stream ciphers: Grain-128a [ÅHJM11], Grain-128 [HJMM06], and Grain-v1 [HJM07]. The Grain family is amongst the most attractive stream ciphers, and especially Grain-v1 is in the eSTREAM portfolio and Grain-128a is standardized by ISO/IEC [ISO15]. Moreover the structure is recently used to design a lightweight hash function [AHMN13] and stream ciphers [AM15,MAM16].

Our new algorithm breaks each of full Grain-128a, Grain-128, and Grain-v1. Among them, this is the first cryptanalysis against full Grain-128a [6]. Regarding full Grain-128, our algorithm is the first attack against the key-stream generator. Regarding full Grain-v1, our algorithm is more efficient than the previous attack [ZXM18], and it breaks Grain-v1 obviously faster than the brute-force attack.

To realize the fast correlation attack against all of the full Grain family, we introduce novel linear approximate representations. They well exploit their structure and reveal a new important vulnerability of the Grain family.

Comparisons with Previous Attacks against Grain Family. To understand this paper, it is not necessary to understand previous attacks, but we summarize previous attacks against the Grain family.

Before Grain-v1, there is an original Grain denoted by Grain-v0 [HJM05], and it was broken by the fast correlation attack [BGM06]. Grain-v1 is tweaked to remove the vulnerability of Grain-v0. Nevertheless, our new fast correlation attack can break full Grain-v1 thanks to the new property.

The near collision attack is the important previous attack against Grain-v1 [ZLFL13], and very recently, an improvement called the fast near collision attack was proposed [ZXM18], where the authors claimed that the time complexity is $2^{75.7}$. However, this estimation is controversial because the unit of the time complexity is "1 update function of reference code on software implementation," and they estimated 1 update function to be $2^{10.4}$ cycles. Therefore, the pure time complexity is rather $2^{75.7+10.4} = 2^{86.1}$ cycles, which is greater than 2^{80}. On the other hand, the time complexity of the fast correlation attack is $2^{76.7}$, where the unit of the (dominant) time complexity is at most one multiplication with fixed values over the finite field. It is obviously faster than the brute-force attack, but it requires more data than the fast near collision attack.

Grain-128 is more aggressively designed than Grain-v1, where a quadratic function is adopted for the nonlinear feedback polynomial of the NFSR. Unfortunately, this low degree causes vulnerability against the dynamic cube attack [DS11]. While the initial work by Dinur and Shamir is a weak-key

[6] Grain-128a has two modes of operation: stream cipher mode and authenticated encryption mode. We assume that all output sequences of the pre-output function can be observed. This assumption naturally holds under the known-plaintext setting on the stream cipher mode. On the other hand, it is difficult to observe them under the reasonable assumption on the authentication mode because the half of the pre-output function is not used as the key stream. Therefore, we do not claim that the authenticated encryption mode is attacked, but remark that the designers of Grain-128a also considered the authentication will rely on the security of the pre-output stream [ÅHJM11].

attack, it was then extended to the single-key attack [DGP+11] and recently improved [FWC17]. The dynamic cube attack breaks the initialization, and the fast correlation attack breaks the key-stream generator. Note that different countermeasures are required for attacks against the key-stream generator and initialization. For example, we can avoid the dynamic cube attack by increasing the number of rounds in the initialization, but such countermeasure does not prevent the attack against the key-stream generator.

Grain-128a was designed to avoid the dynamic cube attack. The degree of the nonlinear feedback polynomial is higher than in Grain-128. No security flaws have been reported on full Grain-128a, but there are attacks against Grain-128a whose number of rounds in the initialization is reduced [LM12,TIHM17,WHT+18].

2 Preliminaries

2.1 LFSR-Based Stream Ciphers

The target of the fast correlation attack is LFSR-based stream ciphers, which are modeled as Fig. 1 simply. The LFSR generates an N-bit output sequence as $\{s_0, s_1, \ldots, s_{N-1}\}$, and the corresponding key stream $\{z_0, z_1, \ldots, z_{N-1}\}$ is computed as $z_t = s_t \oplus e_t$, where e_t is a binary noise.

Let

$$f(x) = c_0 + c_1 x^1 + c_2 x^2 + \cdots + c_{n-1} x^{n-1} + x^n$$

be the feedback polynomial of the LFSR and $s^{(t)} = (s_t, s_{t+1}, \ldots, s_{t+n-1})$ be an n-bit internal state of the LFSR at time t. Then, the LFSR outputs s_t, and the state is updated to $s^{(t+1)}$ as

$$s^{(t+1)} = s^{(t)} \times F = s^{(t)} \times \begin{pmatrix} 0 \cdots 0\ 0 & c_0 \\ 1 \cdots 0\ 0 & c_1 \\ \vdots\ \ddots\ \vdots\ \vdots & \vdots \\ 0 \cdots 1\ 0 & c_{n-2} \\ 0 \cdots 0\ 1 & c_{n-1} \end{pmatrix},$$

where F is an $n \times n$ binary matrix that represents the feedback polynomial $f(x)$. In concrete LFSR-based stream ciphers, the binary noise e_t is nonlinearly generated from the internal state or another internal state.

2.2 Fast Correlation Attack

The fast correlation attack (FCA) exploits high correlation between the internal state of the LFSR and corresponding key stream [Sie84,MS89]. We first show the most simple model, where we assume that e_t itself is highly biased. Let p be the probability of $e_t = 1$, and the correlation c is defined as $c = 1 - 2p$. We guess the initial internal state $s^{(0)}$, calculate $\{s_0, s_1, \ldots, s_{N-1}\}$ from the guessed $s^{(0)}$, and evaluate $\sum_{t=0}^{N-1} (-1)^{s_t \oplus z_t}$, where the sum is computed over the set of integers. If the correct initial state is guessed, the sum is equal to $\sum_{t=0}^{N-1} (-1)^{e_t}$ and follows a normal distribution $\mathcal{N}(Nc, N)$ [7]. On the other hand, we assume that the sum behaves at random when an incorrect initial state is guessed. Then, it follows a normal distribution $\mathcal{N}(0, N)$. To distinguish the two distributions, we need to collect $N \approx O(1/c^2)$ bits of the key stream.

The FCA can be regarded as a kind of a linear cryptanalysis [Mat93]. The output s_t is linearly computed from $s^{(0)}$ as $s_t = \langle s^{(0)}, A_t \rangle$, where A_t is the 1st row vector in the transpose of F^t denoted by ${}^{\mathrm{T}}F^t$. In other words, A_t is used as linear masks, and the aim of attackers is to find $s^{(0)}$ such that $\sum_{t=0}^{N-1} (-1)^{\langle s^{(0)}, A_t \rangle}$ is far from $N/2$.

[7] Accurately, when the correct initial state is guessed, it follows $\mathcal{N}(Nc, N - Nc^2)$. However, since N is huge and Nc^2 is small, the normal distribution $\mathcal{N}(Nc, N)$ is enough to approximate the distribution.

Usually, the binary noise e_t is not highly biased in modern stream ciphers, but we may be able to observe high correlation by summing optimally chosen linear masks. In other words, we can execute the FCA if

$$e'_t = \bigoplus_{i\in\mathbb{T}_s} \langle s^{(t+i)}, \Gamma_i \rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}$$

is highly biased by optimally choosing $\mathbb{T}_s$, $\mathbb{T}_z$, and Γ_i, where $s^{(t+i)}$ and Γ_i are n-bit vectors. Recall $s^{(t)} = s^{(0)} \times F^t$, and then, e'_t is rewritten as

$$\begin{aligned} e'_t &= \bigoplus_{i\in\mathbb{T}_s} \left\langle s^{(t+i)}, \Gamma_i \right\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i} \\ &= \bigoplus_{i\in\mathbb{T}_s} \left\langle s^{(0)} \times F^{t+i}, \Gamma_i \right\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i} \\ &= \left\langle s^{(0)}, \left(\bigoplus_{i\in\mathbb{T}_s} (\Gamma_i \times {}^{\mathrm{T}}F^i) \right) \times {}^{\mathrm{T}}F^t \right\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}. \end{aligned}$$

For simplicity, we introduce Γ denoted by $\Gamma = \bigoplus_{i\in\mathbb{T}_s} (\Gamma_i \times {}^{\mathrm{T}}F^i)$. Then, we can introduce the following parity-check equations as

$$e'_t = \left\langle s^{(0)}, \Gamma \times {}^{\mathrm{T}}F^t \right\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}. \tag{1}$$

We redefine p as the probability satisfying $e'_t = 1$ for all possible t, and the correlation c is also redefined from the corresponding p. Then, we can execute the FCA by using Eq. (1). Assuming that N parity-check equations are collected, we first guess $s^{(0)}$ and evaluate $\sum_{t=0}^{N-1} (-1)^{e'_t}$. While the sum follows a normal distribution $\mathcal{N}(0, N)$ in the random case, it follows $\mathcal{N}(Nc, N)$ if the correct $s^{(0)}$ is guessed.

The most straightforward algorithm requires the time complexity of $O(N2^n)$. Chose et al. showed that the guess and evaluation procedure can be regarded as a Walsh-Hadamard transform [CJM02]. The fast Walsh-Hadamard transform (FWHT) can be successfully applied to accelerate the algorithm, and it reduces the time complexity to $O(N + n2^n)$.

Definition 1 (Walsh-Hadamard Transform (WHT)). *Given a function $w : \{0,1\}^n \to \mathbb{Z}$, the WHT of w is defined as $\hat{w}(s) = \sum_{x\in\{0,1\}^n} w(x)(-1)^{\langle s,x\rangle}$.*

When we guess $s \in \{0,1\}^n$, the empirical correlation $\sum_{t=0}^{N-1} (-1)^{e'_t}$ is rewritten as

$$\begin{aligned} \sum_{t=0}^{N-1} (-1)^{e'_t} &= \sum_{t=0}^{N-1} (-1)^{\langle s, \Gamma \times {}^{\mathrm{T}}F^t \rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}} \\ &= \sum_{x\in\{0,1\}^n} \left(\sum_{t\in\{0,1,\ldots,N-1 | \Gamma \times {}^{\mathrm{T}}F^t = x\}} (-1)^{\langle s,x\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}} \right) \\ &= \sum_{x\in\{0,1\}^n} \left(\sum_{t\in\{0,1,\ldots,N-1 | \Gamma \times {}^{\mathrm{T}}F^t = x\}} (-1)^{\bigoplus_{i\in\mathbb{T}_z} z_{t+i}} \right) (-1)^{\langle s,x\rangle}. \end{aligned}$$

Therefore, from the following public function w as

$$w(x) := \sum_{t\in\{0,1,\ldots,N-1 | \Gamma \times {}^{\mathrm{T}}F^t = x\}} (-1)^{\bigoplus_{i\in\mathbb{T}_z} z_{t+i}},$$

we get $\hat{w}$ by using the FWHT, where $\hat{w}(s)$ is the empirical correlation when s is guessed.

3 Revisiting Fast Correlation Attack

We first review the structure of the parity-check equation by using a finite field and show that $\Gamma \times {}^{\mathrm{T}}F^t$ is "commutative." This new observation brings a new property for the FCA, and it is very important when there are multiple linear masks. As a result, we need to reconsider the wrong-key hypothesis carefully, i.e., there is a case that the most simple and commonly used hypothesis does not hold. Moreover, we propose a new algorithm that successfully exploits the new property to reduce the data and time complexities in the next section.

3.1 Reviewing Parity-Check Equations with Finite Field

We review $\Gamma \times {}^{\mathrm{T}}F^t$ by using a finite field $\mathrm{GF}(2^n)$, where the primitive polynomial is the feedback polynomial of the LFSR.

Recall the notation of $A_t \in \{0,1\}^n$, which was defined as the 1st row vector in ${}^{\mathrm{T}}F^t$, and then, the ith row vector of ${}^{\mathrm{T}}F^t$ is represented as A_{t+i-1}. Let α be a element as $f(\alpha) = 0$ and it is a primitive element of $\mathrm{GF}(2^n)$. We notice that α^t becomes natural conversion of $A_t \in \{0,1\}^n$. We naturally convert $\Gamma \in \{0,1\}^n$ to $\gamma \in \mathrm{GF}(2^n)$. The important observation is that $\Gamma \times {}^{\mathrm{T}}F$ also becomes natural conversion of $\gamma\alpha \in \mathrm{GF}(2^n)$ because of

$$\Gamma \times {}^{\mathrm{T}}F = \Gamma \times \begin{pmatrix} 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \\ c_0 & c_1 & \cdots & c_{n-2} & c_{n-1} \end{pmatrix}.$$

This trivially derives that $\Gamma \times {}^{\mathrm{T}}F^t$ is also natural conversion of $\gamma\alpha^t \in \mathrm{GF}(2^n)$, and of course, the multiplication is commutative, i.e., $\gamma\alpha^t = \alpha^t\gamma$. We finally consider a matrix multiplication corresponding to $\alpha^t\gamma$. Let M_γ be an $n \times n$ binary matrix, where the ith row vector of ${}^{\mathrm{T}}M_\gamma$ is defined as the natural conversion of $\gamma\alpha^{i-1}$. Then, $\alpha^t\gamma$ is the natural conversion of $A_t \times {}^{\mathrm{T}}M_\gamma$, and we acquire $\Gamma \times {}^{\mathrm{T}}F^t = A_t \times {}^{\mathrm{T}}M_\gamma$. The following shows an example to understand this relationship.

Example 1. Let us consider a finite field $\mathrm{GF}(2^8) = \mathrm{GF}(2)[x]/(x^8 + x^4 + x^3 + x^2 + 1)$. When $\Gamma =$ 01011011, the transpose matrix of the corresponding binary matrix M_γ is represented as

$${}^{\mathrm{T}}M_\gamma = \begin{pmatrix} 0&1&0&1&1&0&1&1 \\ 1&0&0&1&0&1&0&1 \\ 1&1&1&1&0&0&1&0 \\ 0&1&1&1&1&0&0&1 \\ 1&0&0&0&0&1&0&0 \\ 0&1&0&0&0&0&1&0 \\ 0&0&1&0&0&0&0&1 \\ 1&0&1&0&1&0&0&0 \end{pmatrix},$$

where the first row coincides with Γ and the second row is natural conversion of $\gamma\alpha$. Then, $\Gamma \times {}^{\mathrm{T}}F^t = A_t \times {}^{\mathrm{T}}M_\gamma$, and for example, when $t = 10$,

$$\Gamma \times {}^{\mathrm{T}}F^{10} = A_{10} \times {}^{\mathrm{T}}M_\gamma,$$

$$\Leftrightarrow \begin{pmatrix} 0&1&0&1&1&0&1&1 \end{pmatrix} \times \begin{pmatrix} 0&1&0&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1 \\ 1&0&1&1&1&0&0&0 \end{pmatrix}^{10} = \begin{pmatrix} 0&0&1&0&1&1&1&0 \end{pmatrix} \times \begin{pmatrix} 0&1&0&1&1&0&1&1 \\ 1&0&0&1&0&1&0&1 \\ 1&1&1&1&0&0&1&0 \\ 0&1&1&1&1&0&0&1 \\ 1&0&0&0&0&1&0&0 \\ 0&1&0&0&0&0&1&0 \\ 0&0&1&0&0&0&0&1 \\ 1&0&1&0&1&0&0&0 \end{pmatrix},$$

and the result is 00010101.

We review Eq. (1) by using the "commutative" feature as

$$\left\langle s^{(0)}, \Gamma \times {}^{\mathrm{T}}F^t \right\rangle = \left\langle s^{(0)}, A_t \times {}^{\mathrm{T}}M_\gamma \right\rangle = \left\langle s^{(0)} \times M_\gamma, A_t \right\rangle,$$

and Eq. (1) is equivalently rewritten as

$$e'_t = \left\langle s^{(0)} \times M_\gamma, A_t \right\rangle \oplus \bigoplus_{i \in \mathbb{T}_z} z_{t+i}.$$

The equation above implies the following new property.

Property 1. We assume that we can observe high correlation when we guess $s^{(0)}$ and parity-check equations are generated from $\Gamma \times {}^{\mathrm{T}}F^t$. Then, we can observe exactly the same high correlation even if we guess $s^{(0)} \times M_\gamma$ and parity-check equations are generated from A_t instead of $\Gamma \times {}^{\mathrm{T}}F^t$.

Hereinafter, $\gamma \in \mathrm{GF}(2^n)$ is not distinguished from $\Gamma \in \{0,1\}^n$, and we use γ as a linear mask for simplicity.

3.2 New Wrong-Key Hypothesis

We review the traditional and commonly used wrong-key hypothesis, where we assume that the empirical correlation behaves as random when an incorrect initial state is guessed. However, Property 1 implies that we need to consider this hypothesis more carefully.

We assume that the use of a linear mask Γ leads to high correlation, and we simply call such linear masks highly biased linear masks. When we generate parity-check equations from $\Gamma \times {}^{\mathrm{T}}F^t$, let us consider the case that we guess incorrect initial state $s'^{(0)} = s^{(0)} \times M_{\gamma'}$. From Property 1

$$\left\langle s'^{(0)}, \Gamma \times {}^{\mathrm{T}}F^t \right\rangle = \left\langle s^{(0)} \times M_{\gamma'}, A_t \times {}^{\mathrm{T}}M_\gamma \right\rangle = \left\langle s^{(0)}, A_t \times {}^{\mathrm{T}}M_{\gamma\gamma'} \right\rangle$$

In other words, it is equivalent to the case that $\gamma\gamma'$ is used as a linear mask instead of γ. If both γ and $\gamma\gamma'$ are highly biased linear masks, we also observe high correlation when we guess $s^{(0)} \times M_{\gamma'}$. Therefore, assuming that the target stream cipher has multiple linear masks with high correlation, the entire corresponding guessing brings high correlation.

We introduce a new wrong-key hypothesis based on Property 1. Assuming that there are m linear masks whose correlation is high and the others are correlation zero, we newly introduce the following wrong-key hypothesis.

Hypothesis 1 (New Wrong-Key Hypothesis) *Assume that there are m highly biased linear masks as $\gamma_1, \gamma_2, \ldots, \gamma_m$, and parity-check equations are generated from A_t. Then, we observe high correlation when we guess $s^{(0)} \times M_{\gamma_i}$ for any $i \in \{1, 2, \ldots, m\}$. Otherwise, we assume that it behaves at random, i.e., the correlation becomes 0.*

The new wrong-key hypothesis is a kind of extension from the traditional wrong-key hypothesis.

4 New Algorithm Exploiting New Property

Overview. We first show the overview before we detail our new attack algorithm. In this section, let n be the size of the LFSR in the target LFSR-based stream cipher, and we assume that there are m ($\ll 2^n$) highly biased linear masks denoted by $\gamma_1, \gamma_2, \ldots, \gamma_m$. The procedure consists of three parts: constructing parity-check equations, FWHT, and removing γ.

– We first construct parity-check equations. Parity-check equations of the traditional FCA are constructed from $\Gamma \times {}^{\mathrm{T}}F^t$ and $\bigoplus_{i \in \mathbb{T}_z} z_{t+i}$. In our new algorithm, we construct parity-check equations from A_t instead of $\Gamma \times {}^{\mathrm{T}}F^t$.

- We use the fast Walsh-Hadamard transform (FWHT) to get solutions with high correlation. In other words, we evaluate s such that $\langle s, A_t \rangle \oplus \bigoplus_{i \in \mathbb{T}_z} z_{t+i}$ is highly biased. As we explained in Sect. 3.1, we then observe high correlation when $s = s^{(0)} \times M_{\gamma_i}$, and there are m solutions with high correlation. Unfortunately, even if FWHT is applied, we have to guess n bits and it requires $n2^n$ time complexity. It is less efficient than the exhaustive search when the size of the LFSR is greater than or equal to the security level. To overcome this issue, we bypass some bits out of n bits by exploiting m linear masks. Specifically, we bypass β bits, i.e., we guess only $(n-\beta)$ bits and β bits are fixed to constant (e.g., 0). Even if β bits are bypassed, there are $m2^{-\beta}$ solutions with high correlation in average. Therefore, $m > 2^\beta$ is a necessary condition.
- We pick solutions whose empirical correlation is greater than a threshold, where some of solutions are represented as $s = s^{(0)} \times M_{\gamma_i}$. To remove M_{γ_i}, we exhaustively guess the applied γ_i and recover $s^{(0)}$. Assuming that N_p solutions are picked, the time complexity is $N_p \times m$. If the expected number of occurrences that the correct $s^{(0)}$ appears is significantly greater than that for incorrect ones, we can uniquely determine $s^{(0)}$. We simulate them by using the Poisson distribution in detail.

4.1 Detailed Algorithm

Let n be the state size of the LFSR and κ be the security level. We assume that there are $m_p\,(\ll 2^n)$ linear masks $\gamma_1, \gamma_2, \ldots, \gamma_{m_p}$ with positive correlation that is greater than a given c. Moreover we assume that there are $m_m\,(\ll 2^n)$ linear masks $\rho_1, \rho_2, \ldots, \rho_{m_m}$ with negative correlation that is smaller than $-c$. Note that c is close to 0, and $m = m_p + m_m$.

Constructing Parity-Check Equations. We first construct parity-check equations from A_t and $\bigoplus_{i \in \mathbb{T}_z} z_{t+i}$ for $t = 0, 1, \ldots, N-1$, and the time complexity is N. The empirical correlation follows $\mathcal{N}(Nc, N)$ and $\mathcal{N}(-Nc, N)$ when we guess one of $s^{(0)} \times M_{\gamma_i}$ and $s^{(0)} \times M_{\rho_i}$, respectively [8]. Otherwise we assume that the empirical correlation follows $\mathcal{N}(0, N)$.

FWHT with Bypassing Technique. We next pick $s \in \{0,1\}^n$ such that $|\frac{\sum_{t=0}^{N-1}(-1)^{e'_t}}{N}| \geq th$, where $e'_t = \langle s, A_t \rangle \oplus \bigoplus_{i \in \mathbb{T}_z} z_{t+i}$ and th (> 0) is a threshold. Let ϵ_1 be the probability that values following $\mathcal{N}(0, N)$ is greater than th, and let ϵ_2 be the probability that values following $\mathcal{N}(Nc, N)$ is greater than th. Namely,

$$\epsilon_1 = \frac{1}{\sqrt{2\pi N}} \int_{th}^{\infty} \exp\left(-\frac{x^2}{2N}\right) dx, \quad \epsilon_2 = \frac{1}{\sqrt{2\pi N}} \int_{th}^{\infty} \exp\left(-\frac{(x-Nc)^2}{2N}\right) dx.$$

Note that the probability that values following $\mathcal{N}(0, N)$ is smaller than $-th$ is also ϵ_1 and the probability that values following $\mathcal{N}(-Nc, N)$ is smaller than $-th$ is also ϵ_2. Let $\mathbb{S}_p$ and $\mathbb{S}_m$ be the set of picked solutions with positive and negative correlation, respectively. The expected size of $\mathbb{S}_p$ and $\mathbb{S}_m$ is $(2^n\epsilon_1 + m_p\epsilon_2)$ and $(2^n\epsilon_1 + m_m\epsilon_2)$, respectively, when the whole of n-bit s is guessed.

Unfortunately, if we guess the whole of n-bit s, the time complexity of FWHT is $n2^n$ and it is less efficient than the exhaustive search when $n \geq \kappa$. To reduce the time complexity, we assume multiple solutions. Instead of guessing the whole of s, we guess its partial $(n-\beta)$ bits, where bypassed β bits are fixed to constants, e.g., all 0. Then, the time complexity of the FWHT is reduced from $n2^n$ to $(n-\beta)2^{n-\beta}$. Even if β bits are bypassed, $m_p 2^{-\beta}\epsilon_2$ (resp. $m_m 2^{-\beta}\epsilon_2$) solutions represented as $s^{(0)} \times M_{\gamma_i}$ (resp. $s^{(0)} \times M_{\rho_i}$) remain. Moreover, the size of $\mathbb{S}_p$ and $\mathbb{S}_m$ also decreases to $(2^{n-\beta}\epsilon_1 + m_p 2^{-\beta}\epsilon_2)$ and $(2^{n-\beta}\epsilon_1 + m_m 2^{-\beta}\epsilon_2)$, respectively.

[8] The correlation c is the lower bound for all γ_i. Therefore, while the empirical correlation may not follow $\mathcal{N}(Nc, N)$, it does not affect the attack feasibility because it is far from $\mathcal{N}(0, N)$.

Removing γ. For all $s \in \mathbb{S}_p$ and all $j \in \{1, 2, \ldots, m_p\}$, we compute $s \times M_{\gamma_j}^{-1}$. It computes $s^{(0)} \times M_{\gamma_i} \times M_{\gamma_j}^{-1}$ and becomes $s^{(0)}$ when $i = j$. Since there are $m_p 2^{-\beta}\epsilon_2$ solutions represented as $s^{(0)} \times M_{\gamma_i}$ in $\mathbb{S}_p$, the correct $s^{(0)}$ appears $m_p 2^{-\beta}\epsilon_2$ times. On the other hand, every incorrect initial state appears about $m_p(2^{n-\beta}\epsilon_1 + m_p 2^{-\beta}\epsilon_2)2^{-n}$ times when we assume uniformly random behavior. In total, every incorrect initial state appears about

$$\begin{aligned}\lambda_1 &= m_p(2^{n-\beta}\epsilon_1 + m_p 2^{-\beta}\epsilon_2)2^{-n} + m_m(2^{n-\beta}\epsilon_1 + m_m 2^{-\beta}\epsilon_2)2^{-n} \\ &= (m2^{n-\beta}\epsilon_1 + (m_p^2 + m_m^2)2^{-\beta}\epsilon_2)2^{-n}\end{aligned}$$

times when we assume uniformly random behavior. On the other hand, the correct $s^{(0)}$ appears

$$\lambda_2 = (m_p + m_m)2^{-\beta}\epsilon_2 = m2^{-\beta}\epsilon_2$$

times.

The number of occurrences that every incorrect initial state appears follows the Poisson distribution with parameter λ_1, and the number of occurrences that the correct $s^{(0)}$ appears follows the Poisson distribution with parameter λ_2. To recover the unique correct $s^{(0)}$, we introduce a threshold th_p as

$$\sum_{k=th_p}^{\infty} \frac{\lambda_1^k e^{-\lambda_1}}{k!} < 2^{-n}.$$

The probability that the number of occurrences that $s^{(0)}$ appears is greater than th_p is estimated as $\sum_{k=th_p}^{\infty} \frac{\lambda_2^k e^{-\lambda_2}}{k!}$. Therefore, if the probability is close to one, we can uniquely recover $s^{(0)}$ with high probability.

4.2 Estimation of Time and Data Complexities

The procedure consists of three parts: constructing parity-check equations, FWHT, and removing γ. The first step requires the time complexity N, where the unit of the time complexity is a multiplication by α over $\mathrm{GF}(2^n)$ and $\bigoplus_{i \in \mathbb{T}_z} z_{t+i}$. The second step requires the time complexity $(n-\beta)2^{n-\beta}$, where the unit of the time complexity is an addition or subtraction [9]. The final step requires the time complexity $(m2^{n-\beta}\epsilon_1 + (m_p^2 + m_m^2)2^{-\beta}\epsilon_2)$, where the unit of the time complexity is a multiplication by fixed values over $\mathrm{GF}(2^n)$. These units of the time complexity are not equivalent, but at least, they are more efficient than the unit given by the initialization of stream ciphers. Therefore, for simplicity, we regard them as equivalent, and the total time complexity is estimated as

$$N + (n-\beta)2^{n-\beta} + m2^{n-\beta}\epsilon_1 + (m_p^2 + m_m^2)2^{-\beta}\epsilon_2.$$

Proposition 1. *Let n be the size of the LFSR in an LFSR-based stream cipher. We assume that there are m linear masks whose absolute value of correlation is greater than c. When the size of bypassed bits is β, we can recover the initial state of the LFSR with time complexity $3(n-\beta)2^{n-\beta}$ and the required number of parity-check equations is $N = (n-\beta)2^{n-\beta}$, where the success probability is $\sum_{k=th_p}^{\infty} \frac{\lambda_2^k e^{-\lambda_2}}{k!}$, where th_p is the minimum value satisfying*

$$\sum_{k=th_p}^{\infty} \frac{N^k e^{-N}}{k!} < 2^{-n},$$

and

$$\begin{aligned}\lambda_2 &= \frac{m2^{-\beta}}{\sqrt{2\pi N}} \int_{th}^{\infty} \exp\left(-\frac{(x-Nc)^2}{2N}\right) dx, \\ th &= \sqrt{2N} \times \mathrm{erfc}^{-1}\left(\frac{2(n-\beta)}{m}\right).\end{aligned}$$

[9] Since we only use $N < 2^n$ parity-check equations, it is enough to use additions or subtraction on n-bit registers.

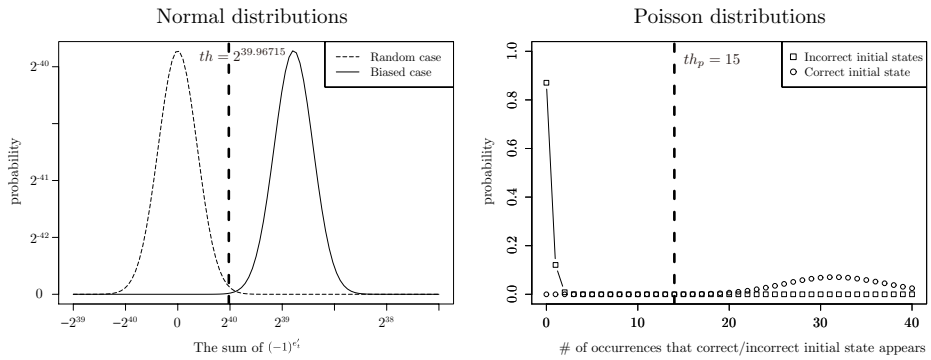

Fig. 2. Theoretical estimation for Example 2.

Proof. The total time complexity is estimated as

$$N + (n-\beta)2^{n-\beta} + m2^{n-\beta}\epsilon_1 + (m_p^2 + m_m^2)2^{-\beta}\epsilon_2.$$

In the useful attack parameter, since $(m_p^2 + m_m^2)2^{-\beta}\epsilon_2$ is significantly smaller than the others, we regard it as negligible. We consider the case that other three terms are balanced, i.e.,

$$N = (n-\beta)2^{n-\beta} = m2^{n-\beta}\epsilon_1,$$

where ϵ_1 is estimated as

$$\epsilon_1 = \frac{1}{\sqrt{2\pi N}} \int_{th}^{\infty} \exp\left(-\frac{x^2}{2N}\right) dx = \frac{1}{2} \times \mathrm{erfc}\left(\frac{th}{\sqrt{2N}}\right) = \frac{n-\beta}{m}.$$

Thus, when th is

$$th = \sqrt{2N} \times \mathrm{erfc}^{-1}\left(\frac{2(n-\beta)}{m}\right),$$

complexities of the three terms are balanced. We finally evaluate the probability that the initial state of the LFSR is uniquely recovered. The number of occurrences that each incorrect value appears follows the Poisson distribution with parameter $\lambda_1 = N2^{-n}$. To discard all $2^n - 1$ incorrect values, recall th_p satisfying $\sum_{k=th_p}^{\infty} \frac{\lambda_1^k e^{-\lambda_1}}{k!} < 2^{-n}$. Then, the success probability is $\sum_{k=th_p}^{\infty} \frac{\lambda_2^k e^{-\lambda_2}}{k!}$ where λ_2 is

$$\lambda_2 = m2^{-\beta}\epsilon_2 = \frac{m2^{-\beta}}{\sqrt{2\pi N}} \int_{th}^{\infty} \exp\left(-\frac{(x-Nc)^2}{2N}\right) dx$$

□

Example 2. Let us consider an attack against an LFSR-based stream cipher with 80-bit LFSR. We assume that there are 2^{14} linear masks whose correlation is greater than 2^{-36}. For $\beta = 9$, we use $N = (80-9) \times 2^{80-9} \approx 2^{77.1498}$ parity-check equations. The left figure of Fig. 2 shows two normal distributions: random and biased cases. If we use a following threshold

$$th = \sqrt{2N} \times \mathrm{erfc}^{-1}\left(\frac{2(n-\beta)}{m}\right) \approx 2^{39.9672},$$

$\epsilon_1 = (n-\beta)/m \approx 2^{-7.8503}$ and $\epsilon_2 = 0.99957$. The expected number of picked solutions is $2^{80-9}\epsilon_1 + 2^{14-9}\epsilon_2 \approx 2^{63.1498} + 31.98627 \approx 2^{63.1498}$. We apply 2^{14} inverse linear masks to the picked solutions and recover $s^{(0)}$, and the time complexity is $2^{63.1498+14} = 2^{77.1498}$.

The number of occurrences that each incorrect value appears follows the Poisson distribution with parameter $\lambda_1 = 2^{77.1498-80} = 2^{-2.8502}$. On the other hand, the number of occurrences that $s^{(0)}$ appears follows the Poisson distribution with parameter $\lambda_2 = 2^{14-9} \times 0.99957 \approx 31.98627$. The right figure of Fig. 2 shows two Poisson distributions. For example, when $th_p = 15$ is used, the probability that an incorrect value appears at least 15 is smaller than 2^{-80}. However, the corresponding probability for $s^{(0)}$ is 99.9%. As a result, the total time complexity is $3 \times 2^{77.1498} \approx 2^{78.7348}$.

5 Application to Grain-128a

We apply the new algorithm to the stream cipher Grain-128a [ÅHJM11], which has two modes of operations: stream cipher mode and authenticated encryption mode. We assume that all output sequences of the pre-output function can be observed. Under the known-plaintext scenario, this assumption is naturally realized for the stream cipher mode because the output is directly used as a key stream. On the other hand, this assumption is very strong for the authenticated encryption mode because only even-clock output is used as the key stream. Therefore, we do not claim that the authenticated encryption mode can be broken.

5.1 Specification of Grain-128a

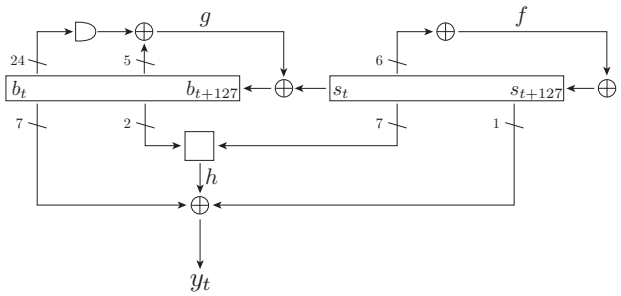

Fig. 3. Specification of Grain-128a

Let $s^{(t)}$ and $b^{(t)}$ be 128-bit internal states of the LFSR and NFSR at time t, respectively, and $s^{(t)}$ and $b^{(t)}$ are represented as $s^{(t)} = (s_t, s_{t+1}, \ldots, s_{t+127})$ and $b^{(t)} = (b_t, b_{t+1}, \ldots, b_{t+127})$. Let y_t be an output of the pre-output function at time t, and it is computed as

$$y_t = h(s^{(t)}, b^{(t)}) \oplus s_{t+93} \oplus \bigoplus_{j \in \mathbb{A}} b_{t+j}, \tag{2}$$

where $\mathbb{A} = \{2, 15, 36, 45, 64, 73, 89\}$, and $h(s^{(t)}, b^{(t)})$ is defined as

$$\begin{aligned} h(s^{(t)}, b^{(t)}) &= h(b_{t+12}, s_{t+8}, s_{t+13}, s_{t+20}, b_{t+95}, s_{t+42}, s_{t+60}, s_{t+79}, s_{t+94}) \\ &= b_{t+12}s_{t+8} \oplus s_{t+13}s_{t+20} \oplus b_{t+95}s_{t+42} \oplus s_{t+60}s_{t+79} \oplus b_{t+12}b_{t+95}s_{t+94}. \end{aligned}$$

Moreover, s_{t+128} and b_{t+128} are computed by

$$\begin{aligned} s_{t+128} &= s_t \oplus s_{t+7} \oplus s_{t+38} \oplus s_{t+70} \oplus s_{t+81} \oplus s_{t+96}, \\ b_{t+128} &= s_t \oplus b_t \oplus b_{t+26} \oplus b_{t+56} \oplus b_{t+91} \oplus b_{t+96} \oplus b_{t+3}b_{t+67} \oplus b_{t+11}b_{t+13} \\ &\quad \oplus b_{t+17}b_{t+18} \oplus b_{t+27}b_{t+59} \oplus b_{t+40}b_{t+48} \oplus b_{t+61}b_{t+65} \oplus b_{t+68}b_{t+84} \\ &\quad \oplus b_{t+88}b_{t+92}b_{t+93}b_{t+95} \oplus b_{t+22}b_{t+24}b_{t+25} \oplus b_{t+70}b_{t+78}b_{t+82}. \end{aligned}$$

Let z_t be the key stream at time t, and $z_t = y_t$ in the stream cipher mode. On the other hand, in the authenticated encryption mode, $z_t = y_{2w+2i}$, where w is the tag size. Figure 3 shows the specification of Grain-128a.

5.2 Linear Approximate Representation for Grain-128a

If there are multiple linear masks with high correlation, the new algorithm can be applied. In this section, we show that Grain-128a has many linear approximate representations, and they produce many linear masks.

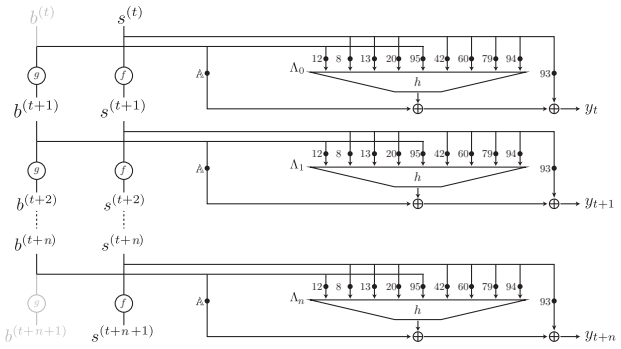

Fig. 4. Linear Approximate Representation for Grain-128a

Figure 4 shows the high-level view of the linear approximate representation. It involves from tth to $(t+n+1)$th rounds, where $b^{(t)}$ and $b^{(t+n+1)}$ must be linearly inactive to avoid involving the state of NFSR. Moreover, y_{t+i} is linearly active for $i \in \mathbb{T}_z$, and the linear mask of the input of the $(t+i)$-round h function denoted by Λ_i must be nonzero for $i \in \mathbb{T}_z$. Otherwise, it must be zero.

We focus on the structure of the h function, where the input consists of 7 bits from the LFSR and 2 bits from the NFSR. Then, non-zero Λ_i can take several values, and specifically, Λ_i can take 64 possible values (see Table 2) under the condition that a linear mask for 2 bits from NFSR is fixed. Since the sum of y_{t+i} for $i \in \mathbb{T}_z$ is used, it implies that there are $64^{|\mathbb{T}_z|}$ linear approximate representations. These many possible representations are obtained by exploiting the structure of the h function, and this structure is common for all ciphers in the Grain family. In other words, this is a new potential vulnerability of the Grain family.

We first consider $\mathbb{T}_z$ to construct the linear approximate representation, but it is difficult to find an optimal $\mathbb{T}_z$. Our strategy is heuristic and does not guarantee the optimality, but the found $\mathbb{T}_z$ is enough to break full Grain-128a. Once $\mathbb{T}_z$ is determined, we first evaluate the correlation of a linear approximate representation on fixed Λ_i for $i \in \{0, 1, \ldots, n\}$. The high-biased linear mask γ used in our new algorithm is constructed by Λ_i, and the correlation of γ is estimated from the correlation of Λ_i.

Finding Linear Masks with High Correlation. We focus on the sum of key stream bits, i.e., $\bigoplus_{i \in \mathbb{T}_z} y_{t+i}$. From Eq. (2), the sum is represented as

$$\begin{aligned}\bigoplus_{i \in \mathbb{T}_z} y_{t+i} &= \bigoplus_{i \in \mathbb{T}_z} \left(h(s^{(t+i)}, b^{(t+i)}) \oplus s_{t+i+93} \oplus \bigoplus_{j \in \mathbb{A}} b_{t+i+j} \right) \\ &= \bigoplus_{i \in \mathbb{T}_z} \left(h(s^{(t+i)}, b^{(t+i)}) \oplus s_{t+i+93} \right) \oplus \bigoplus_{j \in \mathbb{A}} \left(\bigoplus_{i \in \mathbb{T}_z} b_{t+j+i} \right).\end{aligned}$$

We first consider an appropriate set $\mathbb{T}_z$. We focus on $\bigoplus_{i \in \mathbb{T}_z} b_{t+j+i}$ and choose $\mathbb{T}_z$ such that $\bigoplus_{i \in \mathbb{T}_z} b_{t+j+i}$ is highly biased. Concretely, we tap 6 bits whose index corresponds to linearly tapped bits in the g

function, i.e., $\mathbb{T}_z = \{0, 26, 56, 91, 96, 128\}$. Then, for any j,

$$\begin{aligned}\bigoplus_{i\in\mathbb{T}_z} b_{t+j+i} &= b_{t+j} \oplus b_{t+j+26} \oplus b_{t+j+56} \oplus b_{t+j+91} \oplus b_{t+j+96} \oplus b_{t+j+128} \\ &= s_{t+j} \oplus g'(b^{(t+j)}),\end{aligned}$$

where

$$\begin{aligned}g'(b^{(t)}) = {} & b_{t+3}b_{t+67} \oplus b_{t+11}b_{t+13} \oplus b_{t+17}b_{t+18} \oplus b_{t+27}b_{t+59} \oplus b_{t+40}b_{t+48} \\ & \oplus b_{t+61}b_{t+65} \oplus b_{t+68}b_{t+84} \oplus b_{t+88}b_{t+92}b_{t+93}b_{t+95} \\ & \oplus b_{t+22}b_{t+24}b_{t+25} \oplus b_{t+70}b_{t+78}b_{t+82}.\end{aligned}$$

Note that all bits in $g'(b^{(t)})$ are nonlinearly involved, and the correlation may be high. Then

$$\begin{aligned}\bigoplus_{i\in\mathbb{T}_z} y_{t+i} &= \bigoplus_{i\in\mathbb{T}_z} \left(h(s^{(t+i)}, b^{(t+i)}) \oplus s_{t+i+93}\right) \oplus \bigoplus_{j\in\mathbb{A}} \left(s_{t+j} \oplus g'(b^{(t+j)})\right) \\ &= \bigoplus_{i\in\mathbb{T}_z} s_{t+i+93} \oplus \bigoplus_{j\in\mathbb{A}} s_{t+j} \oplus \bigoplus_{i\in\mathbb{T}_z} h(s^{(t+i)}, b^{(t+i)}) \oplus \bigoplus_{j\in\mathbb{A}} g'(b^{(t+j)}).\end{aligned}$$

We next consider a linear approximate representation of $h(s^{(t+i)}, b^{(t+i)})$. Let $\Lambda_i \in \{0,1\}^9$ be the input linear mask for the h function at time $t+i$, and $\Lambda_i = (\Lambda_i[0], \Lambda_i[1], \ldots, \Lambda_i[8])$. Then,

$$\begin{aligned}&h(s^{(t+i)}, b^{(t+i)}) \\ &\approx \Lambda_i[0]b_{t+i+12} \oplus \Lambda_i[4]b_{t+i+95} \oplus \langle \Lambda_i[1-3], (s_{t+i+8}, s_{t+i+13}, s_{t+i+20})\rangle \\ &\quad \oplus \langle \Lambda_i[5-8], (s_{t+i+42}, s_{t+i+60}, s_{t+i+79}, s_{t+i+94})\rangle,\end{aligned}$$

where $\Lambda_i[x-y]$ denotes a sub vector indexed from xth bit to yth bit. Let $cor_{h,i}(\Lambda_i)$ be the correlation of the h function at time $t+i$, and Table 2 summarizes them. From Table 2, $cor_{h,i}(\Lambda_i)$ is 0 or $\pm 2^{-4}$. We have 6 active h functions because $|\mathbb{T}_z| = 6$, and let $\Lambda_{\mathbb{T}_z} \in \{0,1\}^{9\times|\mathbb{T}_z|}$ be the concatenated linear mask, i.e., $\Lambda_{\mathbb{T}_z} = (\Lambda_0, \Lambda_{26}, \Lambda_{56}, \Lambda_{91}, \Lambda_{96}, \Lambda_{128})$. The total correlation from all active h functions depends on $\Lambda_{\mathbb{T}_z}$, and it is computed as $cor_h(\Lambda_{\mathbb{T}_z}) = (-1)^{|\mathbb{T}_z|+1} \prod_{i\in\mathbb{T}_z} cor_{h,i}(\Lambda_i)$ because of the piling-up lemma. Therefore, if Λ_i with correlation 0 is used for any $i \in \mathbb{T}_z$, $cor_h(\Lambda_{\mathbb{T}_z}) = 0$. Otherwise, $cor_h(\Lambda_{\mathbb{T}_z}) = \pm 2^{-24}$.

We guess all terms involved in the internal state of the LFSR in the FCA. Under the correlation $\pm 2^{-24}$, we get

$$\begin{aligned}\bigoplus_{i\in\mathbb{T}_z} y_{t+i} \approx {} & (\text{term by guessing } s^{(t)}) \\ & \oplus \bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[0]b_{t+i+12} \oplus \Lambda_i[4]b_{t+i+95}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right).\end{aligned}$$

Therefore, if

$$\begin{aligned}cor_g(\Lambda_{\mathbb{T}_z}) = {} & \Pr\left[\bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[0]b_{t+i+12} \oplus \Lambda_i[4]b_{t+i+95}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right) = 0\right] \\ & - \Pr\left[\bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[0]b_{t+i+12} \oplus \Lambda_i[4]b_{t+i+95}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right) = 1\right]\end{aligned}$$

is high, the FCA can be successfully applied. Note that $cor_g(\Lambda_{\mathbb{T}_z})$ is independent of $\Lambda_i[1-3, 5-8]$ for any $i \in \mathbb{T}_z$.

Appendix A shows the algebraic normal form of $\bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right)$. To evaluate its correlation, we divide $\bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right)$ into 20 terms, where only b_{t+67} and b_{t+137} are involved by multiple

Table 2. Correlation of the h function. The horizontal axis shows $\Lambda_{h,i}[1-3]$, the vertical axis shows $\Lambda_{h,i}[5-8]$, and $512 \times cor_{h,i}$ is shown in every cell.

	000	001	010	011	100	101	110	111
0000	-32	-32	-32	32	-32	-32	-32	32
0001	0	0	0	0	0	0	0	0
0010	-32	-32	-32	32	-32	-32	-32	32
0011	0	0	0	0	0	0	0	0
0100	-32	-32	-32	32	-32	-32	-32	32
0101	0	0	0	0	0	0	0	0
0110	32	32	32	-32	32	32	32	-32
0111	0	0	0	0	0	0	0	0
1000	-32	-32	-32	32	0	0	0	0
1001	0	0	0	0	-32	-32	-32	32
1010	-32	-32	-32	32	0	0	0	0
1011	0	0	0	0	-32	-32	-32	32
1100	-32	-32	-32	32	0	0	0	0
1101	0	0	0	0	-32	-32	-32	32
1110	32	32	32	-32	0	0	0	0
1111	0	0	0	0	32	32	32	-32

Case of $\Lambda_{h,i}[0,4] = 00$.

	000	001	010	011	100	101	110	111
0000	-32	-32	-32	32	-32	-32	-32	32
0001	0	0	0	0	0	0	0	0
0010	-32	-32	-32	32	-32	-32	-32	32
0011	0	0	0	0	0	0	0	0
0100	-32	-32	-32	32	-32	-32	-32	32
0101	0	0	0	0	0	0	0	0
0110	32	32	32	-32	32	32	32	-32
0111	0	0	0	0	0	0	0	0
1000	32	32	32	-32	0	0	0	0
1001	0	0	0	0	32	32	32	-32
1010	32	32	32	-32	0	0	0	0
1011	0	0	0	0	32	32	32	-32
1100	32	32	32	-32	0	0	0	0
1101	0	0	0	0	32	32	32	-32
1110	-32	-32	-32	32	0	0	0	0
1111	0	0	0	0	-32	-32	-32	32

Case of $\Lambda_{h,i}[0,4] = 01$.

	000	001	010	011	100	101	110	111
0000	-32	-32	-32	32	32	32	32	-32
0001	0	0	0	0	0	0	0	0
0010	-32	-32	-32	32	32	32	32	-32
0011	0	0	0	0	0	0	0	0
0100	-32	-32	-32	32	32	32	32	-32
0101	0	0	0	0	0	0	0	0
0110	32	32	32	-32	-32	-32	-32	32
0111	0	0	0	0	0	0	0	0
1000	-32	-32	-32	32	0	0	0	0
1001	0	0	0	0	32	32	32	-32
1010	-32	-32	-32	32	0	0	0	0
1011	0	0	0	0	32	32	32	-32
1100	-32	-32	-32	32	0	0	0	0
1101	0	0	0	0	32	32	32	-32
1110	32	32	32	-32	0	0	0	0
1111	0	0	0	0	-32	-32	-32	32

Case of $\Lambda_{h,i}[0,4] = 10$.

	000	001	010	011	100	101	110	111
0000	-32	-32	-32	32	32	32	32	-32
0001	0	0	0	0	0	0	0	0
0010	-32	-32	-32	32	32	32	32	-32
0011	0	0	0	0	0	0	0	0
0100	-32	-32	-32	32	32	32	32	-32
0101	0	0	0	0	0	0	0	0
0110	32	32	32	-32	-32	-32	-32	32
0111	0	0	0	0	0	0	0	0
1000	32	32	32	-32	0	0	0	0
1001	0	0	0	0	-32	-32	-32	32
1010	32	32	32	-32	0	0	0	0
1011	0	0	0	0	-32	-32	-32	32
1100	32	32	32	-32	0	0	0	0
1101	0	0	0	0	-32	-32	-32	32
1110	-32	-32	-32	32	0	0	0	0
1111	0	0	0	0	32	32	32	-32

Case of $\Lambda_{h,i}[0,4] = 11$.

terms. Then we try out 4 possible values of (b_{t+67}, b_{t+137}) and evaluate correlation independently. As a result, when $(b_{t+67}, b_{t+137}) = (0,0)$ and $(b_{t+67}, b_{t+137}) = (0,1)$, the correlation is $-2^{-33.1875}$ and $-2^{-33.4505}$, respectively. On the other hand, the correlation is 0 when $b_{t+67} = 1$. Therefore

$$cor_g(\Lambda_{\mathbb{T}_z}) = \frac{-2^{-33.1875} - 2^{-33.4505}}{4} = -2^{-34.313}$$

when $\Lambda_i[0,4] = 0$ for all $i \in \mathbb{T}_z$.

We similarly evaluate $cor_g(\Lambda_{\mathbb{T}_z})$ when $\Lambda_i[0,4] \neq 0$ for any $i \in \mathbb{T}_z$. If one of $\Lambda_0[0]$, $\Lambda_{26}[0]$, $\Lambda_{56}[0]$, $\Lambda_{91}[4]$, $\Lambda_{96}[4]$, and $\Lambda_{128}[4]$ is 1, the correlation is always 0 because b_{t+12}, b_{t+38}, b_{t+68}, b_{t+186}, b_{t+191}, and b_{t+223} are not involved to $\bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)})\right)$. Table 3 summarizes $cor_g(\Lambda_{\mathbb{T}_z})$ when $\Lambda_0[0]$, $\Lambda_{26}[0]$, $\Lambda_{56}[0]$, $\Lambda_{91}[4]$, $\Lambda_{96}[4]$, and $\Lambda_{128}[4]$ are 0.

For any fixed Λ_i, we can get the following linear approximate representation

$$\bigoplus_{i\in\mathbb{T}_z} y_{t+i} \approx \bigoplus_{i\in\mathbb{T}_z} s_{t+i+93} \oplus \bigoplus_{j\in\mathbb{A}} s_{t+j} \oplus \bigoplus_{i\in\mathbb{T}_z} \langle \Lambda_i[1-3], (s_{t+i+8}, s_{t+i+13}, s_{t+i+20}) \rangle \oplus \bigoplus_{i\in\mathbb{T}_z} \langle \Lambda_i[5-8], (s_{t+i+42}, s_{t+i+60}, s_{t+i+79}, s_{t+i+94}) \rangle. \quad (3)$$

Table 3. Summary of correlations when $\Lambda_i[0,4]$ is fixed. Let $*$ be arbitrary bit.

$\Lambda_0[4]$	$\Lambda_{26}[4]$	$\Lambda_{56}[4]$	$\Lambda_{91}[0]$	$\Lambda_{96}[0]$	$\Lambda_{128}[0]$	$cor_g(\Lambda_{\mathbb{T}_z})$
0	0	0	0	0	0	$-2^{-34.3130}$
0	0	0	0	0	1	$+2^{-36.1875}$
0	0	0	0	1	0	$-2^{-37.5860}$
0	0	0	0	1	1	$+2^{-39.4605}$
0	0	0	1	0	0	$-2^{-34.9230}$
0	0	0	1	0	1	$+2^{-36.7975}$
0	0	0	1	1	0	$+2^{-37.5860}$
0	0	0	1	1	1	$-2^{-39.4605}$
0	0	1	0	0	0	$-2^{-35.8980}$
0	0	1	0	0	1	$+2^{-37.7724}$
0	0	1	0	1	0	$-2^{-39.1710}$
0	0	1	0	1	1	$+2^{-41.0454}$
0	0	1	1	0	0	$-2^{-36.5080}$
0	0	1	1	0	1	$+2^{-38.3825}$
0	0	1	1	1	0	$+2^{-39.1710}$
0	0	1	1	1	1	$-2^{-41.0454}$
0	1	0	0	0	0	$-2^{-35.3636}$
0	1	0	0	0	1	$+2^{-37.2381}$
0	1	0	0	1	0	$-2^{-38.1710}$
0	1	0	0	1	1	$+2^{-40.0454}$
0	1	0	1	0	0	$-2^{-35.8490}$
0	1	0	1	0	1	$+2^{-37.7235}$
0	1	0	1	1	0	$+2^{-38.1710}$
0	1	0	1	1	1	$-2^{-40.0454}$
0	1	1	0	0	0	$-2^{-36.9486}$
0	1	1	0	0	1	$+2^{-38.8230}$
0	1	1	0	1	0	$-2^{-39.7559}$
0	1	1	0	1	1	$+2^{-41.6304}$
0	1	1	1	0	0	$-2^{-37.4340}$
0	1	1	1	0	1	$+2^{-39.3085}$
0	1	1	1	1	0	$+2^{-39.7559}$
0	1	1	1	1	1	$-2^{-41.6304}$
1	$*$	$*$	$*$	$*$	$*$	0

From the piling-up lemma, the correlation is computed as

$$-cor_g(\Lambda_{\mathbb{T}_z}) \times cor_h(\Lambda_{\mathbb{T}_z}),$$

where $cor_g(\mathbb{T}_z)$ is summarized in Table 3 and $cor_h(\Lambda_{\mathbb{T}_z}) = (-1)^{|\mathbb{T}_z|+1} \prod_{i\in\mathbb{T}_z} cor_{h,i}(\Lambda_i)$.

How to Find Multiple $\boldsymbol{\gamma}$. The correlation of the linear approximate representation on fixed Λ_i was estimated in the paragraph above. The linear mask γ used in the FCA directly is represented as

$$\gamma = \sum_{i\in\mathbb{T}_z} \big(\Lambda_i[1]\alpha^{i+8} + \Lambda_i[2]\alpha^{i+13} + \Lambda_i[3]\alpha^{i+20} + \Lambda_i[5]\alpha^{i+42} + \Lambda_i[6]\alpha^{i+60} + \Lambda_i[7]\alpha^{i+79} + \Lambda_i[8]\alpha^{i+94} + \alpha^{i+93}\big) + \sum_{j\in\mathbb{A}} \alpha^j.$$

If different $\Lambda_{\mathbb{T}_z}$s derive the same γ, we need to sum up corresponding correlations.

Clearly, since this linear approximate representation does not involve $\Lambda_i[0,4]$ for $i \in \mathbb{T}_z$, we need to sum up $2^{2\times|\mathbb{T}_z|} = 2^{12}$ correlations, where $\Lambda_i[1-3, 5-8]$ is identical and only $\Lambda_i[0,4]$ varies for $i \in \mathbb{T}_z$. Let V be a linear span whose basis is 12 corresponding unit vectors.

Moreover, there are special relationships. When we focus on $\Lambda_{56}[6]$ and $\Lambda_{96}[3]$, corresponding elements over GF(2^{128}) are identical because $\alpha^{56+60} = \alpha^{96+20} = \alpha^{116}$. In other words, $(\Lambda_{56}[6], \Lambda_{96}[3]) = (0,0)$ and $(\Lambda_{56}[6], \Lambda_{96}[3]) = (1,1)$ derive the same γ, and $(\Lambda_{56}[6], \Lambda_{96}[3]) = (1,0)$ and $(\Lambda_{56}[6], \Lambda_{96}[3]) = (0,1)$ also derive the same γ. We have 3 such relationships as follows.

- $\Lambda_{56}[6]$ and $\Lambda_{96}[3]$. Then, $\alpha^{56+60} = \alpha^{96+20} = \alpha^{116}$.
- $\Lambda_{91}[2]$ and $\Lambda_{96}[1]$. Then, $\alpha^{91+13} = \alpha^{96+8} = \alpha^{104}$.
- $\Lambda_{91}[7]$ and $\Lambda_{128}[5]$. Then, $\alpha^{91+79} = \alpha^{128+42} = \alpha^{170}$.

Therefore, from following three vectors

$$
\begin{aligned}
w1(\delta[0]) &= (0^9,0^9,000000100,000000000,000\overline{\delta[0]}00000,000000000),\\
w2(\delta[1]) &= (0^9,0^9,000000000,001000000,0\overline{\delta[1]}0000000,000000000),\\
w3(\delta[2]) &= (0^9,0^9,000000000,000000010,\quad 000000000,00000\overline{\delta[2]}000),
\end{aligned}
$$

a linear span $W(\delta) = \mathrm{span}(w1(\delta[0]), w2(\delta[1]), w3(\delta[2]))$ is defined, where $\overline{\delta[i]} = \delta[i] \oplus 1$. As a result, the correlation for γ denoted by cor_γ is estimated as

$$
cor_\gamma = \sum_{w \in W(\delta)} \sum_{v \in V} -cor_g(\Lambda_{\mathbb{T}_z} \oplus v) \times cor_h(\Lambda_{\mathbb{T}_z} \oplus v \oplus w).
$$

Note that cor_g is independent of $w \in W(\delta)$.

We heuristically evaluated γ with high correlation. As shown in Table 2, the number of possible Λ_i is at most 64. Otherwise, cor_h is always 0. Therefore, the search space is reduced from 2^{54} to 2^{36}. Moreover, Λ_0 is not involved in $W(\delta)$, and the absolute value of cor_γ is invariable as far as we use Λ_0 satisfying $cor_{h,0} = \pm 2^{-4}$. Therefore, we do not need to evaluate Λ_0 anymore, and the search space is further reduced from 2^{36} to 2^{30}. While Λ_{26} is also not involved to $W(\delta)$, we have non-zero correlation for both cases as $\Lambda_{26}[4] = 0$ and 1 (see Table 3). If the sign of $cor_{h,26}$ for $\Lambda_{26}[4] = 0$ is different from that for $\Lambda_{26}[4] = 1$, they cancel each other out. Therefore, we should use Λ_{26} such that the sign of correlation of Λ_{26} is equal to that of $\Lambda_{26} \oplus (000010000)$, and the number of such candidates is 32. Then, we do not need to evaluate Λ_{26} anymore, and the search space is further reduced from 2^{30} to 2^{24}. We finally evaluated 2^{24} $\Lambda_{\mathbb{T}_z}$ exhaustively. As a result, we found $49152 \times 64 \times 32 \approx 2^{26.58}$ γ whose absolute value of correlation is greater than $2^{-54.2381}$.

5.3 Estimation of Attack Complexity and Success Probability

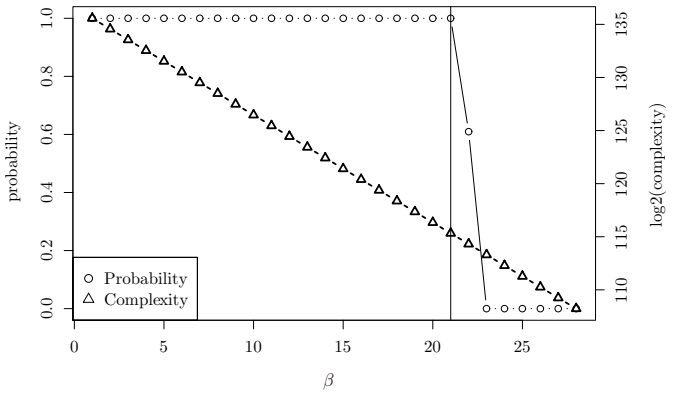

Fig. 5. Time complexity and success probability. FCA against Grain-128a.

We apply the attack algorithm described in Sect. 3, and Proposition 1 is used to estimate the attack complexity and success probability. Figure 5 shows the relationship between the time complexity, success probability, and the size of bypassed bits, where $(n, m, c) = (128, 49152 \times 64 \times 32, \pm 2^{-54.2381})$ is used. From Fig. 5, $\beta = 21$ is preferable. The time complexity is $3 \times (128 - 21) \times 2^{128-21} \approx 2^{115.3264}$ and the corresponding success probability is almost 100%. Moreover when $\beta = 22$, the time complexity is $2^{114.3129}$ and the success probability is 60.95%.

The estimation above only evaluates the time complexity to recover the initial state of the LFSR. To recover the secret key, we need to recover the whole of the initial state. Our next goal is to recover the initial state of the NFSR under the condition that the initial state of the LFSR is uniquely determined, but it is not difficult. We have several methods to recover the initial state and explain the most simple method.

The key stream is generated as Eq. (2). We focus on $(y_0, \ldots, y_{34})$, which involves 128 bits as $(b_2, \ldots, b_{129})$. We first guess 93 bits, and the remaining 35 bits are recovered by using corresponding Eq. (2). Specifically, we first guess $(b_{33}, \ldots, b_{75}, b_{80}, \ldots, b_{129})$. Then, $(b_{76}, \ldots, b_{79})$ are uniquely determined by using $(y_{31}, \ldots, y_{34})$. Similarly, we can uniquely determine the remaining 31 bits step by step. While we need to guess 93 bits, the time complexity is negligible compared with that for the FCA.

6 Application to Grain-128

Grain-128 is the preliminary version of Grain-128a. The dynamic cube attack is successfully applied to analyze full Grain-128 and well exploits the low-degree feedback polynomial of NFSR. Actually, a higher degree feedback polynomial is adopted for Grain-128a to avoid the dynamic cube attack.

The FCA is absolutely different from the dynamic cube attack. While the dynamic cube attack analyzes the initialization, the FCA analyzes the key-stream generator. As far as we know, no vulnerability on the key-stream generator has been reported.

The specification is simpler than Grain-128a. The feedback polynomial of the NFSR is more sparse and is specified as

$$\begin{aligned} b_{t+128} = {} & s_t \oplus b_t \oplus b_{t+26} \oplus b_{t+56} \oplus b_{t+91} \oplus b_{t+96} \oplus b_{t+3}b_{t+67} \oplus b_{t+11}b_{t+13} \\ & \oplus b_{t+17}b_{t+18} \oplus b_{t+27}b_{t+59} \oplus b_{t+40}b_{t+48} \oplus b_{t+61}b_{t+65} \oplus b_{t+68}b_{t+84}. \end{aligned}$$

Moreover there is a small tweak in the h function as

$$h(s^{(t)}, b^{(t)}) = b_{t+12}s_{t+8} \oplus s_{t+13}s_{t+20} \oplus b_{t+95}s_{t+42} \oplus s_{t+60}s_{t+79} \oplus b_{t+12}b_{t+95}s_{t+95},$$

where s_{t+95} is used instead of s_{t+94}.

Since Grain-128 is very similar to Grain-128a, we can use the same $\mathbb{T}_z$. Then $-cor_g = -2^{-32}$, where $\Lambda_{26}[4]$ and $\Lambda_{91}[0]$ can be chosen arbitrary but the others are 0.

We heuristically evaluated γ with high correlation, and we used the same strategy as the case of Grain-128a. As a result, we found $2^{15} \times 64 \times 32 = 2^{26}$ γ with correlation $\pm 2^{-51}$. We apply the attack algorithm described in Sect. 3, and Proposition 1 is used to estimate the attack complexity and success probability. Figure 6 shows the relationship between the time complexity, success probability, and the size of bypassed bits, where $(n, m, c) = (128, 2^{26}, \pm 2^{-51})$ is used. From Fig. 6, $\beta = 22$ is a preferable attack parameter. The time complexity is $3 \times (128 - 22) \times 2^{128-22} \approx 2^{114.3129}$ and the corresponding success probability is 99.0%.

7 Application to Grain-v1

7.1 Specification of Grain-v1

Let $s^{(t)}$ and $b^{(t)}$ be 80-bit internal states of the LFSR and NFSR at time t, respectively, and $s^{(t)}$ and $b^{(t)}$ are represented as $s^{(t)} = (s_t, s_{t+1}, \ldots, s_{t+79})$ and $b^{(t)} = (b_t, b_{t+1}, \ldots, b_{t+79})$, respectively. Then,

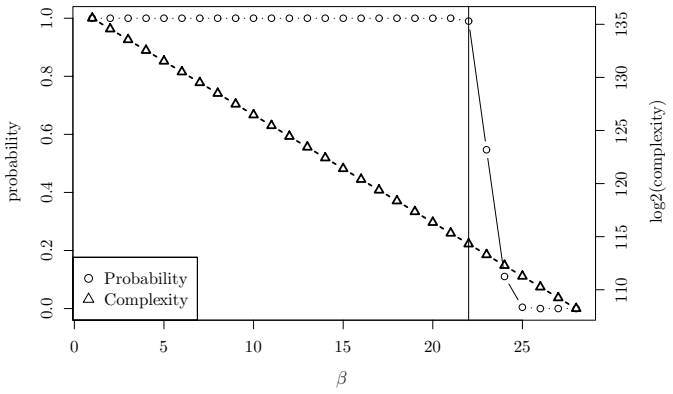

Fig. 6. Time complexity and success probability. FCA against Grain-128.

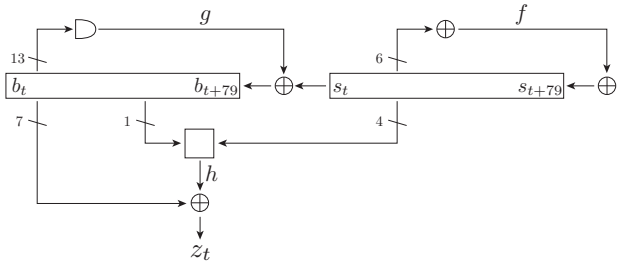

Fig. 7. Specification of Grain-v1

let z_t be a key stream at time t, and it is computed as

$$z_t = h(s^{(t)}, b^{(t)}) \oplus \bigoplus_{j \in \mathbb{A}} b_{t+j}, \tag{4}$$

where $\mathbb{A} = \{1, 2, 4, 10, 31, 43, 56\}$ and $h(s^{(t)}, b^{(t)})$ is defined as

$$\begin{aligned} h(s^{(t)}, b^{(t)}) &= h(s_{t+3}, s_{t+25}, s_{t+46}, s_{t+64}, b_{t+63}) \\ &= s_{t+25} \oplus b_{t+63} \oplus s_{t+3}s_{t+64} \oplus s_{t+46}s_{t+64} \oplus s_{t+64}b_{t+63} \\ &\quad \oplus s_{t+3}s_{t+25}s_{t+46} \oplus s_{t+3}s_{t+46}s_{t+64} \oplus s_{t+3}s_{t+46}b_{t+63} \\ &\quad \oplus s_{t+25}s_{t+46}b_{t+63} \oplus s_{t+46}s_{t+64}b_{t+63}. \end{aligned}$$

Moreover, s_{t+80} and b_{t+80} are computed by

$$\begin{aligned} s_{t+80} &= s_t \oplus s_{t+13} \oplus s_{t+23} \oplus s_{t+38} \oplus s_{t+51} \oplus s_{t+62}, \\ b_{t+80} &= s_t \oplus b_{t+62} \oplus b_{t+60} \oplus b_{t+52} \oplus b_{t+45} \oplus b_{t+37} \oplus b_{t+33} \oplus b_{t+28} \oplus b_{t+21} \\ &\quad \oplus b_{t+14} \oplus b_{t+9} \oplus b_t \oplus b_{t+63}b_{t+60} \oplus b_{t+37}b_{t+33} \oplus b_{t+15}b_{t+9} \\ &\quad \oplus b_{t+60}b_{t+52}b_{t+45} \oplus b_{t+33}b_{t+28}b_{t+21} \oplus b_{t+63}b_{t+45}b_{t+28}b_{t+9} \\ &\quad \oplus b_{t+60}b_{t+52}b_{t+37}b_{t+33} \oplus b_{t+63}b_{t+60}b_{t+21}b_{t+15} \\ &\quad \oplus b_{t+63}b_{t+60}b_{t+52}b_{t+45}b_{t+37} \oplus b_{t+33}b_{t+28}b_{t+21}b_{t+15}b_{t+9} \\ &\quad \oplus b_{t+52}b_{t+45}b_{t+37}b_{t+33}b_{t+28}b_{t+21}. \end{aligned}$$

Figure 7 shows the specification of Grain-v1.

Table 4. Correlation of the h function, where $32 \times cor_{h,i}$ is shown in every cell.

	$\Lambda_i[0-3]$															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
$\Lambda_i[4]=0$	0	0	0	0	0	-8	0	8	0	8	0	-8	-8	8	-8	8
$\Lambda_i[4]=1$	0	-8	0	8	-8	-8	-8	-8	0	0	0	0	0	-8	0	8

7.2 Fast Correlation Attack against Grain-v1

When we use $\mathbb{T}_z = \{0, 14, 21, 28, 37, 45, 52, 60, 62, 80\}$, we focus on the sum of the key stream bits, i.e., $z_{t+0} \oplus z_{t+14} \oplus z_{t+21} \oplus z_{t+28} \oplus z_{t+37} \oplus z_{t+45} \oplus z_{t+52} \oplus z_{t+60} \oplus z_{t+62} \oplus z_{t+80}$.

$$\bigoplus_{i\in\mathbb{T}_z} z_{t+i} = \bigoplus_{i\in\mathbb{T}_z} h(s^{(t+i)}, b^{(t+i)}) \oplus \bigoplus_{j\in\mathbb{A}} \left(\bigoplus_{i\in\mathbb{T}_z} b_{t+j+i} \right).$$

For any j,

$$\bigoplus_{i\in\mathbb{T}_z} b_{t+j+i} = s_{t+j} \oplus g'(b^{(t+j)}),$$

where $g'(b^{(t)})$ is defined as

$$\begin{aligned} g'(b^{(t)}) = {} & b_{t+33} \oplus b_{t+9} \oplus b_{t+63}b_{t+60} \oplus b_{t+37}b_{t+33} \oplus b_{t+15}b_{t+9} \oplus b_{t+60}b_{t+52}b_{t+45} \\ & \oplus b_{t+33}b_{t+28}b_{t+21} \oplus b_{t+63}b_{t+45}b_{t+28}b_{t+9} \oplus b_{t+60}b_{t+52}b_{t+37}b_{t+33} \\ & \oplus b_{t+63}b_{t+60}b_{t+21}b_{t+15} \oplus b_{t+63}b_{t+60}b_{t+52}b_{t+45}b_{t+37} \\ & \oplus b_{t+33}b_{t+28}b_{t+21}b_{t+15}b_{t+9} \oplus b_{t+52}b_{t+45}b_{t+37}b_{t+33}b_{t+28}b_{t+21}. \end{aligned}$$

Then

$$\begin{aligned} \bigoplus_{i\in\mathbb{T}_z} z_{t+i} &= \bigoplus_{i\in\mathbb{T}_z} h(s^{(t+i)}, b^{(t+i)}) \oplus \bigoplus_{j\in\mathbb{A}} \left(s_{t+j} \oplus g'(b^{(t+j)}) \right) \\ &= \bigoplus_{j\in\mathbb{A}} s_{t+j} \oplus \bigoplus_{i\in\mathbb{T}_z} h(s^{(t+i)}, b^{(t+i)}) \oplus \bigoplus_{j\in\mathbb{A}} g'(b^{(t+j)}). \end{aligned}$$

We next consider a linear approximate representation of $h(s^{(t+i)}, b^{(t+i)})$. Let Λ_i be the input linear mask for the h function at time $t+i$. Then

$$\begin{aligned} & h(s^{(t+i)}, b^{(t+i)}) \\ & \approx \Lambda_i[4]b_{t+i+63} \oplus \langle \Lambda_i[0-3], (s_{t+i+3}, s_{t+i+25}, s_{t+i+46}, s_{t+i+64}) \rangle. \end{aligned}$$

Let $cor_{h,i}(\Lambda_i)$ be the correlation of the h function at time $t+i$, and Table 4 summarizes them. From Table 4, $cor_{h,i}(\Lambda_i)$ is 0 or $\pm 2^{-2}$. Since we have $|\mathbb{T}_z| = 10$ active h functions, the total correlation from all active h functions is computed as $(-1)^{|\mathbb{T}_z|+1} \prod_{i\in\mathbb{T}_z} cor_{h,i}(\Lambda_i) = \pm 2^{-20}$ because of the piling-up lemma. Note that $\Lambda_i[0-3]$ is independent from the state of the NFSR.

All terms involved in the internal state of the LFSR can be guessed in the FCA. Therefore, under the correlation $\pm 2^{-20}$, we get

$$\bigoplus_{i\in\mathbb{T}_z} z_{t+i} = (\text{term by guessing}) \oplus \bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[4]b_{t+i+63}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)}) \right).$$

Therefore, if

$$\begin{aligned} cor_g(\Lambda_{\mathbb{T}_z}) = {} & \Pr\left[\bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[4]b_{t+i+63}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)}) \right) = 0 \right] \\ & - \Pr\left[\bigoplus_{i\in\mathbb{T}_z} (\Lambda_i[4]b_{t+i+63}) \oplus \bigoplus_{j\in\mathbb{A}} \left(g'(b^{(t+j)}) \right) = 1 \right] \end{aligned}$$

Table 5. Summary of correlations when $\Lambda_i[4]$ is fixed.

$\Lambda_{14}[4]$	$\Lambda_{21}[4]$	$\Lambda_{28}[4]$	$\Lambda_{45}[4]$	$cor_g(\Lambda_{\mathbb{T}_z})$
0	0	0	0	$-2^{-39.7159}$
0	0	0	1	$-2^{-43.4500}$
0	0	1	0	$-2^{-39.6603}$
0	0	1	1	$-2^{-43.7260}$
0	1	0	0	$+2^{-45.1228}$
0	1	0	1	$-2^{-42.9025}$
0	1	1	0	$+2^{-44.3802}$
0	1	1	1	$-2^{-42.6875}$
1	0	0	0	$+2^{-41.9519}$
1	0	0	1	$+2^{-43.5233}$
1	0	1	0	$+2^{-41.8662}$
1	0	1	1	$+2^{-43.6420}$
1	1	0	0	$-2^{-44.9114}$
1	1	0	1	$+2^{-42.8544}$
1	1	1	0	$-2^{-44.5232}$
1	1	1	1	$+2^{-42.7302}$

is high, the FCA can be successfully applied.

Similarly to the case of Grain-128a, we evaluate $cor_g(\Lambda_{\mathbb{T}_z})$. If one of $\Lambda_0[4]$, $\Lambda_{37}[4]$, $\Lambda_{52}[4]$, $\Lambda_{60}[4]$, $\Lambda_{62}[4]$, and $\Lambda_{80}[4]$ is 1, the correlation is always 0 because b_{t+63}, b_{t+100}, b_{t+115}, b_{t+123}, b_{t+125}, and b_{t+143} are not involved in $\bigoplus_{j\in\mathbb{A}}\left(g'(b^{(t+j)})\right)$. Table 5 summarizes $cor_g(\Lambda_{\mathbb{T}_z})$ when $\Lambda_i[4] = 0$ for $i \in \{0, 37, 52, 60, 62, 80\}$.

For any fixed Λ_i, we can get the following linear approximate representation

$$\bigoplus_{i\in\mathbb{T}_z} z_{t+i} \approx \bigoplus_{j\in\mathbb{A}} s_{t+j} \oplus \bigoplus_{i\in\mathbb{T}_z} \langle \Lambda_i[0-3], (s_{t+i+3}, s_{t+i+25}, s_{t+i+46}, s_{t+i+64})\rangle. \tag{5}$$

From the piling-up lemma, the correlation is computed as $-cor_g(\Lambda_{\mathbb{T}_z}) \times cor_h(\Lambda_{\mathbb{T}_z})$.

How to Find Multiple γ. The correlation of the linear approximate representation on fixed Λ_i was estimated in the paragraph above. The linear mask γ used in the FCA directly is represented as

$$\gamma = \sum_{i\in\mathbb{T}_z} \left(\Lambda_i[0]\alpha^{i+3} + \Lambda_i[1]\alpha^{i+25} + \Lambda_i[2]\alpha^{i+46} + \Lambda_i[3]\alpha^{i+64}\right) + \sum_{j\in\mathbb{A}} \alpha^j.$$

If different Λ_h have the same γ, we need to sum up corresponding correlations.

This linear approximate representation does not use $\Lambda_i[4]$ for $i \in \mathbb{T}_z$. Therefore, we need to sum up $2^{|\mathbb{T}_z|} = 2^{10}$ correlations, where $\Lambda_i[0-3]$ is identical and only $\Lambda_i[5]$ varies for $i \in \mathbb{T}_z$. Let V be a linear span whose basis is 12 corresponding unit vectors.

Moreover, there are special relationships similar to the case of Grain-128a, and we have four such relationships as

- $\Lambda_{37}[2]$ and $\Lambda_{80}[0]$. Then, $\alpha^{37+46} = \alpha^{80+3} = \alpha^{83}$.
- $\Lambda_{62}[3]$ and $\Lambda_{80}[2]$. Then, $\alpha^{62+64} = \alpha^{80+46} = \alpha^{126}$.
- $\Lambda_0[2]$ and $\Lambda_{21}[1]$. Then, $\alpha^{0+46} = \alpha^{21+25} = \alpha^{46}$.
- $\Lambda_{21}[3]$ and $\Lambda_{60}[1]$. Then, $\alpha^{21+64} = \alpha^{60+25} = \alpha^{85}$.

Therefore, from following four vectors

$$w1(\delta[0]) = (00000,0^5,\quad 00000,0^5,00100,0^5,0^5,00000,\quad 00000,\overline{\delta[0]}0000),$$
$$w2(\delta[1]) = (00000,0^5,\quad 00000,0^5,00000,0^5,0^5,00000,\quad 00010,00\overline{\delta[1]}00),$$
$$w3(\delta[2]) = (00100,0^5,0\overline{\delta[2]}000,0^5,00000,0^5,0^5,00000,\quad 00000,00000),$$
$$w4(\delta[3]) = (00000,0^5,\quad 00010,0^5,00000,0^5,0^5,0\overline{\delta[3]}000,00000,00000),$$

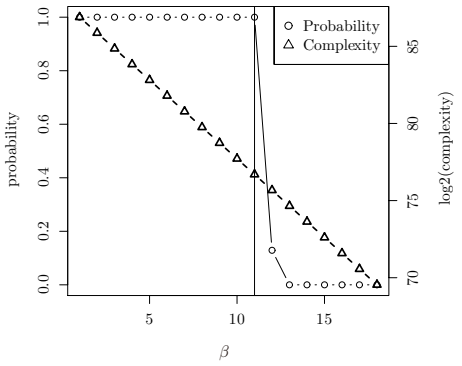

Fig. 8. Time complexity and success probability. FCA against Grain-v1.

a linear span $W(\delta) = \mathrm{span}(w1(\delta[0]), w2(\delta[1]), w3(\delta[2]), w4(\delta[3]))$ is defined, where $\overline{\delta[i]} = \delta[i] \oplus 1$. Then, let cor_γ be the correlation of γ, and

$$cor_\gamma = \sum_{w \in W(\delta)} \sum_{v \in V} -cor_g(\Lambda_{\mathbb{T}_z} \oplus v) \times cor_h(\Lambda_{\mathbb{T}_z} \oplus v \oplus w).$$

We heuristically evaluated γ with high correlation. For every element in $\mathbb{T}_z$, since the subset $\{14, 28, 45, 52\}$ is independent of the special relationship, we first focus on the subset. Since $b_{t+63+52}$ is not involved in $\bigoplus_{j \in \mathbb{A}} \left(g'(b^{(t+j)})\right)$, $\Lambda_{52}[4]$ must be 0. Therefore, $\Lambda_{52}[0-3]$ should be chosen as

$$\Lambda_{52}[0-3] \in \{0101, 0111, 1001, 1011, 1100, 1101, 1110, 1111\},$$

and cor_γ is invariable as far as we use Λ_{52} satisfying $cor_{h,52} = \pm 2^{-2}$. We do not need to evaluate Λ_{52} anymore, and the search space is reduced from 2^{40} to 2^{36}. For $i \in \{14, 28, 45\}$, corresponding masks should be chosen as

$$\Lambda_i[0-3] \in \{0101, 0111, 1001, 1011, 1100, 1101, 1110, 1111\}$$

because $cor_g(\Lambda_{\mathbb{T}_z})$ is high when $(\Lambda_{14}[4], \Lambda_{21}[4], \Lambda_{28}[4], \Lambda_{45}[4])$ is 0010 or 0000. Let us focus on Table 5. We have three-type linear masks as

- $\Lambda_i[0-3] \in \{1001, 1011, 1100, 1110\}$, where $cor_{h,i} = \pm 2^{-2}$ for $\Lambda_i[4] = 0$ but $cor_{h,i} = 0$ for $\Lambda_i[4] = 1$.
- $\Lambda_i[0-3] \in \{0111, 1101\}$, where the sign of $cor_{h,i}$ is different in each case of $\Lambda_i[4] = 0$ or 1.
- $\Lambda_i[0-3] \in \{0101, 1111\}$, where the sign of $cor_{h,i}$ is the same in both cases of $\Lambda_i[4] = 0$ and 1.

Since cor_γ is invariable in each case, it is enough to evaluate one from each case. Therefore, the search space is reduced from 2^{36} to $3^3 \times 2^{24}$. We finally evaluated 9×2^{24} $\Lambda_{\mathbb{T}_z}$ exhaustively. As a result, we found about 442368 γ whose absolute value of correlation is greater than 2^{-36}.

Estimating Attack Complexity and Success Probability. We apply the attack algorithm described in Sect. 3, and Proposition 1 is used to estimate the attack complexity and success probability. Figure 8 shows the relationship between the time complexity, success probability, and the size of bypassed bits, where $(n, m, c) = (80, 442368, \pm 2^{-36})$ is used. From Fig. 8, $\beta = 11$ is preferable, and the time complexity is $3 \times (80 - 11) \times 2^{80-11} \approx 2^{76.6935}$ and the corresponding success probability is almost 100%.

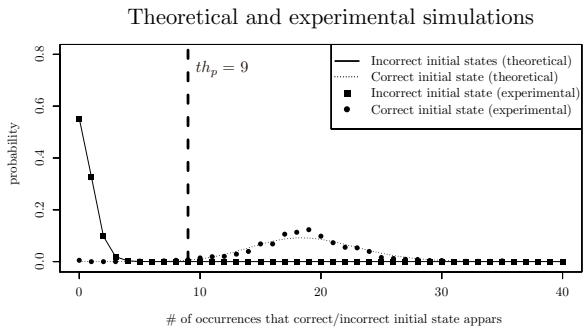

Fig. 9. Comparison between the theoretical and experimental estimations.

8 Verifications, Observations, and Countermeasures

8.1 Experimental Verification

We verify our algorithm by applying it to a toy Grain-like cipher, where the sizes of the LFSR and NFSR are 24 bits, and s_{t+24}, b_{t+24}, and z_t are computed as

$$\begin{aligned} s_{t+24} &= s_t \oplus s_{t+1} \oplus s_{t+2} \oplus s_{t+7}, \\ b_{t+24} &= s_t \oplus b_t \oplus b_{t+5} \oplus b_{t+14} \oplus b_{t+20}b_{t+21} \oplus b_{t+11}b_{t+13}b_{t+15}, \\ z_t &= h(s_{t+3}, s_{t+7}, s_{t+15}, s_{t+19}, b_{t+17}) \oplus \bigoplus_{j\in\{1,3,8\}} b_{t+j}, \end{aligned}$$

where the h function is as the one used in Grain-v1.

Similarly to the case of Grain-128a, $\mathbb{T}_z$ is used by tapping linear part of the feedback polynomial of NFSR, i.e., $\mathbb{T}_z = \{0, 5, 14, 24\}$. Then, the sum of the key stream is

$$\bigoplus_{i\in\mathbb{T}_z} z_{t+i} = \bigoplus_{i\in\mathbb{T}_z} h(s^{(t+i)}, b^{(t+i)}) \oplus \bigoplus_{j\in\{1,3,8\}} \left(s_{t+j} + g'(b^{(t+j)})\right),$$

where $g'(b^{(t)}) = b_{t+20}b_{t+21} \oplus b_{t+11}b_{t+13}b_{t+15}$. The ANF of the h function involves b_{t+17}, b_{t+22}, b_{t+31}, and b_{t+41}. If $\Lambda_i[4] = 1$ is used for $i \in \{0, 14, 24\}$, the correlation is always 0 because $\bigoplus_{j\in\{1,3,8\}} g'(b^{(t+j)})$ does not involve b_{t+17}, b_{t+31}, and b_{t+41}. Only b_{t+22} is involved to $\bigoplus_{j\in\{1,3,8\}} g'(b^{(t+j)})$. Therefore, we evaluated correlations of $\bigoplus_{j\in\{1,3,8\}} g'(b^{(t+j)})$ and $\bigoplus_{j\in\{1,3,8\}} g'(b^{(t+j)}) \oplus b_{t+22}$, and they have the correlation $2^{-3.41504}$. For $i \in \{0, 14, 24\}$, we have 8 possible linear masks. Moreover, we should use 0101 and 1111 for the linear mask $\Lambda_{14}[0-3]$ because the sign of the correlation is the same in either case of $\Lambda_{14}[4] = 0$ and $\Lambda_{14}[4] = 1$. As a result, we have $8 \times 8 \times 8 \times 2 = 1024$ linear masks whose absolute value of correlations is $2 \times 2^{-8-3.41504} = 2^{-10.41504}$, where the factor 2 is derived from the sum of correlations for $\Lambda_{14}[4] = 0$ and $\Lambda_{14}[4] = 1$.

For example, when $\beta = 5$, the data complexity is $(24-5) \times 2^{24-5} \approx 2^{23.25}$. From Proposition 1, when we use $th = 6579$ as the threshold for the normal distribution, the complexities for three steps of the attack algorithm are balanced. Moreover, when we use $th_p = 9$ as the threshold for the Poisson distribution, the probability that incorrect initial state appears at least th_p times is $2^{-26} < 2^{-24}$.

We randomly choose the initial state and repeat the attack algorithm 1000 times. Figure 9 shows the comparison of the Poisson distributions between the theoretical and experimental ones. From this figure, our experimental results almost follow the theoretical one.

8.2 Unified Representation with Finite Field

The "commutative" property of $\Gamma \times {}^{\mathrm{T}}F^t$ is exploited in our new fast correlation attack, where $\Gamma \in \{0,1\}^n$ and $F^t \in \{0,1\}^{n\times n}$ are regarded as $\gamma \in \mathrm{GF}(2^n)$ and $\alpha^t \in \mathrm{GF}(2^n)$, respectively. We further consider the finite field representation of $s^{(0)} \in \{0,1\}^n$.

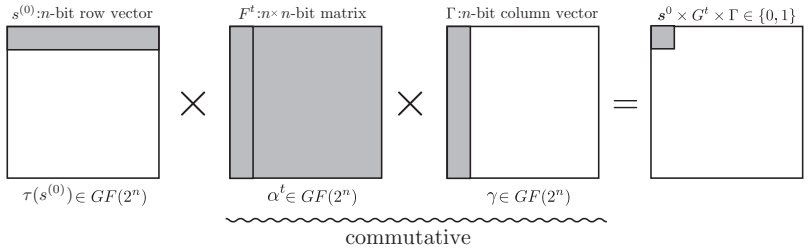

Fig. 10. "Commutative" property

Recall Eq.(1), the parity-check equation is represented as

$$\begin{aligned} e'_t &= \left\langle s^{(0)}, \Gamma \times {}^{\mathrm{T}}F^t \right\rangle \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i} \\ &= s^{(0)} \times F^t \times {}^{\mathrm{T}}\Gamma \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i}. \end{aligned}$$

We equivalently transform $F^t \times {}^{\mathrm{T}}\Gamma$ into $\alpha^t\gamma$ in our new algorithm. We further consider the equivalent representation of $s^{(0)}$ over $\mathrm{GF}(2^n)$, which is denoted by $\tau(s^{(0)})$, and Eq.(1) is rewritten as

$$e'_t = (\tau(s^{(0)})\gamma\alpha^t)[0] \oplus \bigoplus_{i\in\mathbb{T}_z} z_{t+i},$$

where $(\tau(s^{(0)})\gamma\alpha^t)[0]$ is the first coefficient of $\tau(s^{(0)})\gamma\alpha^t$, and Fig. 10 shows the overview.

The conversion function $\tau : \{0,1\}^n \to \mathrm{GF}(2^n)$ is a bit trickier than conversions for F^t and Γ. It is not natural because $s^{(0)}$ is an n-bit **row** vector, and therefore, we need to introduce a conversion function τ as follows.

Definition 2 (Conversion function τ). *For any $y \in \mathrm{GF}(2^n)$, let us consider an $n \times n$ matrix $[y, \alpha y, \alpha^2 y, \ldots, \alpha^{n-1} y]$. Then $\tau^{-1}(y)$ is the first row n-bit vector in this matrix, and τ is the inversion of τ^{-1}.*

The following is an example in the case of $\mathrm{GF}(2^8) = \mathrm{GF}(2)[x]/(x^8 + x^4 + x^3 + x^2 + 1)$.

Example 3. We consider the conversion τ for $\mathrm{GF}(2^8) = \mathrm{GF}(2)[x]/(x^8 + x^4 + x^3 + x^2 + 1)$. When $y = \alpha(= 01000000)$ and $y = \alpha + \alpha^3 + \alpha^4 + \alpha^6 + \alpha^7(= 01011011)$, the first row of the matrix $[y, \alpha y, \alpha^2 y, \ldots, \alpha^7 y]$ is 00000001 and 01101001, respectively, because

$$\left(\begin{array}{c|cccccccc} 0 & 0&0&0&0&0&0&0&1 \\ \hline 1 & 0&0&0&0&0&0&0&0 \\ 0 & 1&0&0&0&0&0&0&1 \\ 0 & 0&1&0&0&0&0&0&1 \\ 0 & 0&0&1&0&0&0&0&1 \\ 0 & 0&0&0&1&0&0&0&0 \\ 0 & 0&0&0&0&1&0&0&0 \\ 0 & 0&0&0&0&0&1&0&0 \\ 0 & 0&0&0&0&0&0&1&0 \end{array}\right) \text{ and } \left(\begin{array}{c|cccccccc} 0 & 1&1&0&1&0&0&1&1 \\ \hline 1 & 0&1&1&0&1&0&0&0 \\ 0 & 0&1&1&0&0&1&1&1 \\ 1 & 1&1&1&0&0&0&0&0 \\ 1 & 0&0&1&0&0&0&1&1 \\ 0 & 1&0&0&1&0&0&0&0 \\ 1 & 0&1&0&0&1&0&0&0 \\ 1 & 1&0&1&0&0&1&0&0 \end{array}\right).$$

Therefore $\tau(00000001) = \alpha = 01000000$ and $\tau(01101001) = \alpha + \alpha^3 + \alpha^4 + \alpha^6 + \alpha^7 = 01011011$.

8.3 Experimental Path Search Algorithm

An unified representation with the finite field is shown in Sect. 8.2, where $s^{(0)}$ is also represented by the corresponding element over the finite field. This representation enables us to reveal highly biased linear masks experimentally.

We have to enumerate high-biased linear masks and their correlations before we execute fast correlation attacks. In the application to Grain family, such masks and correlations were theoretically simulated under the plausible assumption. Here, we demonstrate another method based on an experimental approach.

Our current goal is not to recover $s^{(0)}$, and rather, we choose $s^{(0)}$ at random and aim to enumerate high-biased linear masks and their correlations. Therefore, we randomly choose $s^{(0)}$ and execute the first and second steps of our algorithm. Then, we observe high correlation by guessing $s^{(0)} \times M_{\gamma_i} = \tau^{-1}(\tau(s^{(0)})\gamma_i)$ if γ_i is one of highly-biased linear masks. Assuming that guessing s brings high correlation, the corresponding high-biased linear mask is calculated as $\gamma_i = \tau(s)(\tau(s^{(0)}))^{-1}$ because

$$\begin{aligned} s &= \tau^{-1}(\tau(s^{(0)})\gamma_i) \\ &= \tau^{-1}(\tau(s^{(0)})\tau(s)(\tau(s^{(0)}))^{-1}) \\ &= \tau^{-1}(\tau(s)). \end{aligned}$$

In other words, we can enumerate high-biased linear masks experimentally by exploiting known correct initial state. The complexity of our experimental path search algorithm is almost equivalent with the complexity of our fast correlation attack. Therefore, when we assume attackers who can execute our fast correlation attack practically, they can also enumerate high-biased linear masks experimentally.

In fact, we applied the experimental path search algorithm to the toy Grain-like cipher described in Sect. 8.1. Our theoretical estimation indicates 1024 high-biased linear masks, and our experimental path search algorithms also indicated the same linear masks.

8.4 Another View to Find Preferable $\mathbb{T}_z$

In our strategy, we first searched for $\mathbb{T}_z$, which brings the best linear characteristic. A mixed integer linear programming (MILP) is often applied to search for the best linear characteristics of block ciphers [MWGP11,SHW+14], and this method is naturally applied to search for the best linear characteristic of the fast correlation attack. We first generate an MILP model to represent linear trail with specific number of rounds R. Then, we maximize the probability of the linear characteristic under the condition that $b^{(0)}$ and $b^{(R)}$ are linearly inactive.

We used $\mathbb{T}_z = \{0, 26, 56, 91, 96, 128\}$ and $\mathbb{T}_z = \{0, 14, 21, 28, 37, 45, 52, 60, 62, 80\}$ for Grain-128a and Grain-v1, respectively, and they bring the best linear characteristic. For Grain-128a and Grain-v1, the correlation of the linear characteristic are $\pm 2^{-80.159}$ and $\pm 2^{-38.497}$, respectively. It is not enough to estimate the correlation only from the best characteristic because we need to take into account of the effect by multiple characteristics. For example, assuming that there are two characteristics whose absolute values of correlations are the same but their signs are different, these two characteristics cancel each other. On the other hand, if their signs are the same, we can observe double correlations. Especially, it is very interesting that Grain-128a has significant gain from the best linear characteristic. While the MILP is useful to find the best characteristic, there is no method to find multiple linear characteristics without repeating MILPs. Therefore, we used the MILP only to detect a preferable $\mathbb{T}_z$, and the corresponding correlation is estimated as explained in Sects. 5, 6, and 7.

8.5 Possible Countermeasure against Our New Attack

The simplest countermeasure is to suppress the output at every second position when the key stream is output. For example, the authenticated encryption mode of Grain-128a has such structure, where the key stream is output only in the even clock. When we attack Grain-128a, we want to use $\mathbb{T}_z = \{0, 26, 56, 91, 96, 128\}$, but we cannot tap 91. As far as we search, we cannot detect a preferable $\mathbb{T}_z$ under the condition that the tapped indices are only even numbers. On the other hand, this countermeasure leads to low throughput.

Another countermeasure would be to limit the length of the key stream for each pair of secret key and iv. It would become difficult to collect enough parity-check equations to execute the FCA.

Lightweight stream ciphers often have such restriction, e.g., Plantlet outputs only 2^{30}-bit key stream for each pair of secret key and iv [MAM16]. On the other hand, the advantage of stream ciphers can keep high performance once the initialization finishes, and such restriction does not use the advantage very well.

Acknowledgments. The authors thank the anonymous CRYPTO 2018 reviewers for careful reading and many helpful comments. Takanori Isobe was supported in part by Grant-in-Aid for Young Scientist (B) (KAKENHI 17K12698) for Japan Society for the Promotion of Science. Bin Zhang is supported by the National Key R&D Research programm (Grant No. 2017YFB0802504), the program of the National Natural Science Foundation of China (Grant No. 61572482), National Cryptography Development Fund (Grant No. MMJJ20170107).

References

ÅHJM11. Martin Ågren, Martin Hell, Thomas Johansson, and Willi Meier. Grain-128a: a new version of Grain-128 with optional authentication. *IJWMC*, 5(1):48–59, 2011.

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

AM15. Frederik Armknecht and Vasily Mikhalev. On lightweight stream ciphers with shorter internal states. In Gregor Leander, editor, *FSE 2015*, volume 9054 of *LNCS*, pages 451–470. Springer, 2015.

BGM06. Côme Berbain, Henri Gilbert, and Alexander Maximov. Cryptanalysis of Grain. In Matthew J. B. Robshaw, editor, *FSE 2006*, volume 4047 of *LNCS*, pages 15–29. Springer, 2006.

CJM02. Philippe Chose, Antoine Joux, and Michel Mitton. Fast correlation attacks: An algorithmic point of view. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 209–221. Springer, 2002.

CJS00. Vladimir V. Chepyzhov, Thomas Johansson, and Ben J. M. Smeets. A simple algorithm for fast correlation attacks on stream ciphers. In Bruce Schneier, editor, *FSE 2000*, volume 1978 of *LNCS*, pages 181–195. Springer, 2000.

CS91. Vladimir V. Chepyzhov and Ben J. M. Smeets. On A fast correlation attack on certain stream ciphers. In Donald W. Davies, editor, *EUROCRYPT '91*, volume 547 of *LNCS*, pages 176–185. Springer, 1991.

CT00. Anne Canteaut and Michaël Trabbia. Improved fast correlation attacks using parity-check equations of weight 4 and 5. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 573–588. Springer, 2000.

DGP+11. Itai Dinur, Tim Güneysu, Christof Paar, Adi Shamir, and Ralf Zimmermann. An experimentally verified attack on full Grain-128 using dedicated reconfigurable hardware. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 327–343. Springer, 2011.

DS11. Itai Dinur and Adi Shamir. Breaking Grain-128 with dynamic cube attacks. In Antoine Joux, editor, *FSE 2011*, volume 6733 of *LNCS*, pages 167–187. Springer, 2011.

FWC17. Ximing Fu, Xiaoyun Wang, and Jiazhe Chen. Determining the nonexistent terms of non-linear multivariate polynomials: How to break Grain-128 more efficiently. *IACR Cryptology ePrint Archive*, 2017:412, 2017.

HJM05. Martin Hell, Thomas Johansson, and Willi Meier. Grain - a stream cipher for constrained environments, 2005. `http://www.ecrypt.eu.org/stream`.

HJM07. Martin Hell, Thomas Johansson, and Willi Meier. Grain: a stream cipher for constrained environments. *IJWMC*, 2(1):86–93, 2007.

HJMM06. Martin Hell, Thomas Johansson, Alexander Maximov, and Willi Meier. A stream cipher proposal: Grain-128. In *IEEE International Symposium on Information Theory (ISIT 2006),*, pages 1614–1618. IEEE, 2006.

ISO15. ISO/IEC. JTC1: ISO/IEC 29167-13: Information technology – automatic identification and data capture techniques – part 13: Crypto suite Grain-128A security services for air interface communications, 2015.

JJ99a. Thomas Johansson and Fredrik Jönsson. Fast correlation attacks based on turbo code techniques. In Michael J. Wiener, editor, *CRYPTO '99*, volume 1666 of *LNCS*, pages 181–197. Springer, 1999.

JJ99b. Thomas Johansson and Fredrik Jönsson. Improved fast correlation attacks on stream ciphers via convolutional codes. In Jacques Stern, editor, *EUROCRYPT '99*, volume 1592 of *LNCS*, pages 347–362. Springer, 1999.

LLP08. Jung-Keun Lee, Dong Hoon Lee, and Sangwoo Park. Cryptanalysis of Sosemanuk and SNOW 2.0 using linear masks. In Josef Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 524–538. Springer, 2008.

LM12. Michael Lehmann and Willi Meier. Conditional differential cryptanalysis of Grain-128a. In Josef Pieprzyk, Ahmad-Reza Sadeghi, and Mark Manulis, editors, *CANS 2012*, volume 7712 of *LNCS*, pages 1–11. Springer, 2012.

MAM16. Vasily Mikhalev, Frederik Armknecht, and Christian Müller. On ciphers that continuously access the non-volatile key. *IACR Trans. Symmetric Cryptol.*, 2016(2):52–79, 2016.

Mat93. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *EUROCRYPT '93*, volume 765 of *LNCS*, pages 386–397. Springer, 1993.

MFI01. Miodrag J. Mihaljevic, Marc P. C. Fossorier, and Hideki Imai. Fast correlation attack algorithm with list decoding and an application. In Mitsuru Matsui, editor, *FSE 2001*, volume 2355 of *LNCS*, pages 196–210. Springer, 2001.

MG90. Miodrag J. Mihaljevic and Jovan Dj. Golic. A fast iterative algorithm for a shift register initial state reconstruction given the nosiy output sequence. In Jennifer Seberry and Josef Pieprzyk, editors, *AUSCRYPT '90*, volume 453 of *LNCS*, pages 165–175. Springer, 1990.

MS89. Willi Meier and Othmar Staffelbach. Fast correlation attacks on certain stream ciphers. *J. Cryptology*, 1(3):159–176, 1989.

MWGP11. Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In Chuankun Wu, Moti Yung, and Dongdai Lin, editors, *Inscrypt 2011*, volume 7537 of *LNCS*, pages 57–76. Springer, 2011.

SHW+14. Siwei Sun, Lei Hu, Peng Wang, Kexin Qiao, Xiaoshuang Ma, and Ling Song. Automatic security evaluation and (related-key) differential characteristic search: Application to SIMON, PRESENT, LBlock, DES(L) and other bit-oriented block ciphers. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 158–178. Springer, 2014.

Sie84. Thomas Siegenthaler. Correlation-immunity of nonlinear combining functions for cryptographic applications. *IEEE Trans. Information Theory*, 30(5):776–780, 1984.

TIHM17. Yosuke Todo, Takanori Isobe, Yonglin Hao, and Willi Meier. Cube attacks on non-blackbox polynomials based on division property. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part III*, volume 10403 of *LNCS*, pages 250–279. Springer, 2017.

Wag02. David A. Wagner. A generalized birthday problem. In Moti Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 288–303. Springer, 2002.

WHT+18. Qingju Wang, Yonglin Hao, Yosuke Todo, Chaoyun Li, Takanori Isobe, and Willi Meier. Improved division property based cube attacks exploiting algebraic properties of superpoly. *CRYPTO 2018*, 2018. Accepted at CRYPTO 2018, `http://eprint.iacr.org/2017/1063`.

ZF06. Bin Zhang and Dengguo Feng. Multi-pass fast correlation attack on stream ciphers. In Eli Biham and Amr M. Youssef, editors, *SAC 2006*, volume 4356 of *LNCS*, pages 234–248. Springer, 2006.

ZLFL13. Bin Zhang, Zhenqi Li, Dengguo Feng, and Dongdai Lin. Near collision attack on the Grain v1 stream cipher. In Shiho Moriai, editor, *FSE 2013*, volume 8424 of *LNCS*, pages 518–538. Springer, 2013.

ZXM15. Bin Zhang, Chao Xu, and Willi Meier. Fast correlation attacks over extension fields, large-unit linear approximation and cryptanalysis of SNOW 2.0. In Rosario Gennaro and Matthew Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 643–662. Springer, 2015.

ZXM18. Bin Zhang, Chao Xu, and Willi Meier. Fast near collision attack on the Grain v1 stream cipher. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part II*, volume 10821 of *LNCS*, pages 771–802. Springer, 2018.

ZYR90. Kencheng Zeng, Chung-Huang Yang, and T. R. N. Rao. An improved linear syndrome algorithm in cryptanalysis with applications. In Alfred Menezes and Scott A. Vanstone, editors, *CRYPTO '90*, volume 537 of *LNCS*, pages 34–47. Springer, 1990.

A Algebraic Normal Form of $\bigoplus_{j\in\mathbb{A}}\left(g'(\boldsymbol{b}^{t+j})\right)$

$$
\begin{aligned}
&\bigoplus_{j\in\mathbb{A}}\left(g'(\boldsymbol{b}^{t+j})\right)\\
&= g'(\boldsymbol{b}^{t+2}) \oplus g'(\boldsymbol{b}^{t+12}) \oplus g'(\boldsymbol{b}^{t+36}) \oplus g'(\boldsymbol{b}^{t+45}) \oplus g'(\boldsymbol{b}^{t+64}) \oplus g'(\boldsymbol{b}^{t+73}) \oplus g'(\boldsymbol{b}^{t+89})\\
&= b_{t+24}b_{t+26}b_{t+27} \oplus b_{t+26}b_{t+28}\\
&\quad \oplus b_{t+29}b_{t+61} \oplus b_{t+58}b_{t+60}b_{t+61} \oplus b_{t+56}b_{t+58}\\
&\quad \oplus (b_{t+85}b_{t+93}b_{t+97} \oplus b_{t+85}b_{t+93} \oplus b_{t+90}b_{t+94}b_{t+95}b_{t+97} \oplus b_{t+97}b_{t+101}\\
&\qquad \oplus b_{t+95}b_{t+97}b_{t+98} \oplus b_{t+63}b_{t+95} \oplus b_{t+90}b_{t+91} \oplus b_{t+63}\underline{b_{t+67}} \oplus b_{t+55}b_{t+63}\\
&\qquad \oplus b_{t+62}b_{t+63} \oplus b_{t+91}b_{t+123} \oplus b_{t+115}b_{t+123}b_{t+127})\\
&\quad \oplus (b_{t+5}b_{t+69} \oplus \underline{b_{t+67}}b_{t+69}b_{t+70} \oplus b_{t+70}b_{t+86} \oplus b_{t+86}b_{t+88}b_{t+89} \oplus b_{t+84}b_{t+86}\\
&\qquad \oplus b_{t+72}b_{t+80}b_{t+84} \oplus b_{t+76}b_{t+84} \oplus b_{t+76}b_{t+80} \oplus b_{t+72}b_{t+104} \oplus b_{t+76}b_{t+140}\\
&\qquad \oplus b_{t+104}b_{t+120} \oplus b_{t+104}b_{t+112} \oplus b_{t+133}\underbrace{b_{t+137}}b_{t+138}b_{t+140} \oplus b_{t+48}b_{t+112}\\
&\qquad \oplus b_{t+134}b_{t+138} \oplus b_{t+134}b_{t+142}b_{t+146})\\
&\quad \oplus b_{t+13}b_{t+15}\\
&\quad \oplus b_{t+19}b_{t+20}\\
&\quad \oplus b_{t+42}b_{t+50} \oplus b_{t+42}b_{t+74}\\
&\quad \oplus (b_{t+106}b_{t+114}b_{t+118} \oplus b_{t+106}b_{t+110} \oplus b_{t+106}b_{t+107} \oplus b_{t+111}b_{t+113}b_{t+114}\\
&\qquad \oplus b_{t+103}b_{t+107}b_{t+108}b_{t+110} \oplus b_{t+113}b_{t+129} \oplus b_{t+113}b_{t+121} \oplus b_{t+39}b_{t+103}\\
&\qquad \oplus b_{t+124}b_{t+128}b_{t+129}b_{t+131} \oplus b_{t+125}b_{t+129} \oplus b_{t+129}\underbrace{b_{t+137}}\\
&\qquad \oplus b_{t+37}b_{t+39}b_{t+40} \oplus \underline{b_{t+67}}b_{t+131})\\
&\quad \oplus b_{t+143}b_{t+151}b_{t+155}\\
&\quad \oplus b_{t+18}b_{t+82} \oplus b_{t+81}b_{t+82}\\
&\quad \oplus b_{t+32}b_{t+33}\\
&\quad \oplus b_{t+83}b_{t+99}\\
&\quad \oplus (b_{t+92}b_{t+156} \oplus b_{t+152}b_{t+156}b_{t+157}b_{t+159} \oplus b_{t+141}b_{t+157} \oplus b_{t+157}b_{t+173}\\
&\qquad \oplus b_{t+159}b_{t+167}b_{t+171})\\
&\quad \oplus b_{t+132}b_{t+148} \oplus b_{t+100}b_{t+132} \oplus b_{t+116}b_{t+148} \oplus b_{t+100}b_{t+102}\\
&\quad \oplus b_{t+47}b_{t+49}\\
&\quad \oplus b_{t+53}b_{t+54}\\
&\quad \oplus b_{t+75}b_{t+77}\\
&\quad \oplus b_{t+161}b_{t+165}b_{t+166}b_{t+168}\\
&\quad \oplus b_{t+150}b_{t+154}\\
&\quad \oplus b_{t+177}b_{t+181}b_{t+182}b_{t+184}
\end{aligned}
$$

Table 6. Correlation of $\bigoplus_{j\in\mathbb{A}} \left(g'(\boldsymbol{b}^{t+j})\right)$.

No.	Term of Boolean function	correlation
1	$b_{t+24}b_{t+26}b_{t+27} \oplus b_{t+26}b_{t+28}$	-0.5
2	$b_{t+29}b_{t+61} \oplus b_{t+58}b_{t+60}b_{t+61} \oplus b_{t+56}b_{t+58}$	-0.25
3	$b_{t+85}b_{t+93}b_{t+97} \oplus b_{t+85}b_{t+93} \oplus b_{t+90}b_{t+94}b_{t+95}b_{t+97}$ $\oplus b_{t+97}b_{t+101} \oplus b_{t+95}b_{t+97}b_{t+98} \oplus b_{t+63}b_{t+95} \oplus b_{t+90}b_{t+91}$ $\oplus b_{t+63}\underline{b_{t+67}} \oplus b_{t+55}b_{t+63} \oplus b_{t+62}b_{t+63} \oplus b_{t+91}b_{t+123}$ $\oplus b_{t+115}b_{t+123}b_{t+127}$	-0.046875
4	$b_{t+5}b_{t+69} \oplus \underline{b_{t+67}}b_{t+69}b_{t+70} \oplus b_{t+70}b_{t+86} \oplus b_{t+86}b_{t+88}b_{t+89}$ $\oplus b_{t+84}b_{t+86} \oplus b_{t+72}b_{t+80}b_{t+84} \oplus b_{t+76}b_{t+84} \oplus b_{t+76}b_{t+80}$ $\oplus b_{t+72}b_{t+104} \oplus b_{t+76}b_{t+140} \oplus b_{t+104}b_{t+120} \oplus b_{t+104}b_{t+112}$ $\oplus b_{t+133}\underline{\underline{b_{t+137}}}b_{t+138}b_{t+140} \oplus b_{t+48}b_{t+112} \oplus b_{t+134}b_{t+138}$ $\oplus b_{t+134}b_{t+142}b_{t+146}$	if $b_{t+137}=0$, $-2^{-6.41504}$ if $b_{t+137}=1$, $-2^{-6.67807}$
5	$b_{t+13}b_{t+15}$	-0.5
6	$b_{t+19}b_{t+20}$	-0.5
7	$b_{t+42}b_{t+50} \oplus b_{t+42}b_{t+74}$	-0.5
8	$b_{t+106}b_{t+114}b_{t+118} \oplus b_{t+106}b_{t+110} \oplus b_{t+106}b_{t+107}$ $\oplus b_{t+111}b_{t+113}b_{t+114} \oplus b_{t+103}b_{t+107}b_{t+108}b_{t+110} \oplus b_{t+113}b_{t+129}$ $\oplus b_{t+113}b_{t+121} \oplus b_{t+39}b_{t+103} \oplus b_{t+124}b_{t+128}b_{t+129}b_{t+131}$ $\oplus b_{t+125}b_{t+129} \oplus b_{t+129}\underline{\underline{b_{t+137}}} \oplus \underline{b_{t+67}}b_{t+131} \oplus b_{t+37}b_{t+39}b_{t+40}$	if $b_{t+67}=0$, $-2^{-4.14202}$ if $b_{t+67}=1$, 0
9	$b_{t+143}b_{t+151}b_{t+155}$	-0.75
10	$b_{t+18}b_{t+82} \oplus b_{t+81}b_{t+82}$	-0.5
11	$b_{t+32}b_{t+33}$	-0.5
12	$b_{t+83}b_{t+99}$	-0.5
13	$b_{t+92}b_{t+156} \oplus b_{t+152}b_{t+156}b_{t+157}b_{t+159} \oplus b_{t+141}b_{t+157}$ $\oplus b_{t+157}b_{t+173} \oplus b_{t+159}b_{t+167}b_{t+171}$	-0.1875
14	$b_{t+132}b_{t+148} \oplus b_{t+100}b_{t+132} \oplus b_{t+116}b_{t+148} \oplus b_{t+100}b_{t+102}$	-0.25
15	$b_{t+47}b_{t+49}$	-0.5
16	$b_{t+53}b_{t+54}$	-0.5
17	$b_{t+75}b_{t+77}$	-0.5
18	$b_{t+161}b_{t+165}b_{t+166}b_{t+168}$	-0.875
19	$b_{t+150}b_{t+154}$	-0.5
20	$b_{t+177}b_{t+181}b_{t+182}b_{t+184}$	-0.875

B Examples of γ for Grain-v1

As we show in Sect. 7, the linear mask γ is represented as

$$\gamma = \sum_{i \in \mathbb{T}_z} \left(\Lambda_i[0]\alpha^{i+3} + \Lambda_i[1]\alpha^{i+25} + \Lambda_i[2]\alpha^{i+46} + \Lambda_i[3]\alpha^{i+64} \right) + \sum_{j \in \mathbb{A}} \alpha^j.$$

As an example, we use following linear masks and $\delta = 0000$. Then the following linear approximate representations have the same γ.

	$\Lambda_i[0-3]$										
$W(\delta)$	0	14	21	28	37	45	52	60	62	80	correlation
	0101	0111	1010	0101	0101	0101	0101	1011	0100	0111	0
$w4$	0101	0111	1011	0101	0101	0101	0101	1111	0100	0111	0
$w3$	0111	0111	1110	0101	0101	0101	0101	1011	0100	0111	0
$w3$ $w4$	0111	0111	1111	0101	0101	0101	0101	1111	0100	0111	0
$w2$	0101	0111	1010	0101	0101	0101	0101	1011	0101	0101	0
$w2$ $w4$	0101	0111	1011	0101	0101	0101	0101	1111	0101	0101	$-2^{-38.2558}$
$w2$ $w3$	0111	0111	1110	0101	0101	0101	0101	1011	0101	0101	$-2^{-38.2558}$
$w2$ $w3$ $w4$	0111	0111	1111	0101	0101	0101	0101	1111	0101	0101	$-2^{-38.0837}$
$w1$	0101	0111	1010	0101	0111	0101	0101	1011	0100	1111	0
$w1$ $w4$	0101	0111	1011	0101	0111	0101	0101	1111	0100	1111	0
$w1$ $w3$	0111	0111	1110	0101	0111	0101	0101	1011	0100	1111	0
$w1$ $w3$ $w4$	0111	0111	1111	0101	0111	0101	0101	1111	0100	1111	0
$w1$ $w2$	0101	0111	1010	0101	0111	0101	0101	1011	0101	1101	0
$w1$ $w2$ $w4$	0101	0111	1011	0101	0111	0101	0101	1111	0101	1101	$-2^{-38.2558}$
$w1$ $w2$ $w3$	0111	0111	1110	0101	0111	0101	0101	1011	0101	1101	$-2^{-38.2558}$
$w1$ $w2$ $w3$ $w4$	0111	0111	1111	0101	0111	0101	0101	1111	0101	1101	$-2^{-38.0837}$
total											$-2^{-35.6112}$

If $\Lambda_{21}[0-3] = 1010$, the correlation is 0. Moreover $\Lambda_{62}[4]$ must be 0, and $\Lambda_{62} = 01000$ is correlation 0.