Vectorial Decoding Algorithm for Fast Correlation Attack and Its Applications to Stream Cipher Grain-128a*

ZhaoCun Zhou[1,2], DengGuo Feng[1,3], and Bin Zhang[1]

[1] TCA Laboratory, SKLCS, Institute of Software, Chinese Academy of Science, Beijing, China
[2] University of Chinese Academy of Science, Beijing, China
[3] State Key Laboratory of Computer Science, ISCAS, Beijing, China
zhaocun@iscas.ac.cn
martin_zhangbin@hotmail.com

Abstract. Fast correlation attacks, pioneered by Meier and Staffelbach, is an important cryptanalysis tool for LFSR-based stream cipher, which exploits the correlation between the LFSR state and key stream and targets at recovering the initial state of LFSR via a decoding algorithm. In this paper, we develop a vectorial decoding algorithm for fast correlation attack, which is a natural generalization of original binary approach. Our approach benefits from the contributions of all correlations in a subspace. We propose two novel criterions to improve the iterative decoding algorithm. We also give some cryptographic properties of the new FCA which allows us to estimate the efficiency and complexity bounds. Furthermore, we apply this technique to well-analyzed stream cipher Grain-128a. Based on a hypothesis, an interesting result for its security bound is deduced from the perspective of iterative decoding. Our analysis reveals the potential vulnerability for LFSRs over generic linear group and also for nonlinear functions with high SEI multidimensional linear approximations such as Grain-128a.

Keywords: Linear Approximation · Fast Correlation Attack· Iterative Decoding · Grain-128a.

1 Introduction

Stream ciphers are a widely used class of symmetric-key cryptosystem. A key stream sequence is generated from the initial state derived from the key. The plaintext is encrypted by XORing with the key stream in the same length.

Linear feedback shift register(LFSR) based stream ciphers form an important class of stream-cipher system, in which one or more LFSRs are often used. LFSRs could be defined over different algebraic structures, such as finite fields and generic linear group. Besides for LFSR, these ciphers usually adopt a nonlinear

* Supported by organization x.

filter function or a finite state automata(FSM) with nonlinear update function. The history of these ciphers can be traced back to decades ago, e.g., LILI-128[11], SNOW family [14, 15, 34, 16] and Grain family etc.

Grain family includes three well-known stream ciphers: Grain-128a[2], Grain-128[20] and Grain-v1[21]. Grain-v1 is in the eSTREAM portfolio and Grain-128a is standardized by ISO/IEC[1]. All Grain family members share a similar structure. Several lightweight ciphers proposed recently also adopt similar structures[4, 5, 31]. An important attack for Grain-v1 is near collision attack[37], which is improved in [38]. Since Grain-128 adopts quadratic function, the dynamic cube attack plays an important role in its cryptanalysis[13]. To avoid the dynamic cube attack, Grain-128a adopts a nonlinear function with higher degree. However, Grain family is reported to be vulnerable for fast correlation attacks(FCA) in CRYPTO 18[33].

FCA is pioneered by Meier and Staffelbach in 1989[27]. Generally speaking, FCA exploits the correlation between the key stream and the state or the outputs of LFSR. The problem of recovering initial state of LFSR is transformed into a decoding problem. The linear part of the stream cipher is treated as a linear code, and the nonlinear part of the stream cipher is treated as noise.

According to the differences of decoding strategies, these FCA approaches can be roughly divided into two classes. One class adopts one-pass decoding algorithm. For example, Johansson et. al. proposed the FCA based on convolution codes and Viterbi decoding algorithm[24], and sooner improved it by turbo codes[23]. Another FCA adopts maximum likelihood decoding on a reduced set of information bits[8]. The parity checks are usually folded to eliminate partial bits. List decoding and polynomial reconstruction can also be applied in FCA [28, 25]. An important improvement is accelerating the parity check evaluations by fast Walsh-Hadamard transform[10]. This technique is applied in cryptanalyzing stream cipher E0[26]. It was then generalized to extension fields and applied to stream cipher SNOW 2.0[39]. An recent improvement of FCA is based on commutative property and applied to Grain family[33].

The other class adopts several pass decoding algorithm. After Meier and Staffelbach's original FCA, low-density parity-check code(LDPC) is introduced into FCA to improve the iterative decoding algorithm[7]. There are many related works in this area, such as [3, 7, **?**,12, 18, 29, 30]. Intuitively, iterative decoding algorithm seems to be more powerful, as their decoding abilities are closer to Shannon's bound. However, comparing with the FCA decoding by information set, it is usually very hard to describe its cryptographic properties in mathematical language, and also lack of a convenient approach to work on extension fields. Thereby, its direct application to modern stream ciphers is very limited.

Our Contributions. In this paper, we propose a vectorial iterative decoding algorithm for fast correlation attack, which generalizes Meier and Staffelbach's original FCA very naturally. Our approach benefits from the contributions of all correlations in a subspace and thereby more powerful than the binary version. We propose two novel criterions to improve the iterative decoding algorithm and

perform a scaled experiments to verify its validity. We also give some cryptographic properties for the first iteration, which allows us to estimate the efficiency and complexity bound via probability distribution approximations.

Furthermore, we apply it to the well-analyzed stream cipher Grain-128a. Based on a hypothesis that the initial probability distribution of noise is close to symmetric probability distribution, and there exist parity checks with two taps or with special form, we give a data complexity bound estimation in the sense of correcting errors of the noisy sequence. The result shows its potential security bound may be lower than we thought from the perspective of iterative decoding. Our analysis reveals the potential vulnerability for LFSRs over generic linear group and also for nonlinear functions with high SEI multidimensional linear approximations such as Grain-128a.

Outline. The rest of the paper is organized as follows. Section 2 is preliminary. The details of vectorial decoding algorithm are described in section 3. In section 4, we propose some cryptographic properties and perform an scaled experiment. How to apply the new FCA to Grain-128a is explained in section 5. Section 6 consists of some further problems. Finally, we conclude the paper.

2 Preliminary

2.1 Notations and Definitions

Linear Approximation. Let $F : \mathbb{F}_2^m \to \mathbb{F}_2^n$ be a vectorial Boolean function. A linear approximation of F with m-bit input mask $\boldsymbol{u} = (u_1, u_2, \cdots, u_m)$ and n-bit output mask pair $\boldsymbol{v} = (v_1, v_2, \cdots, v_n)$ can be represented by

$$\boldsymbol{u}\boldsymbol{x}^T \oplus \boldsymbol{v}F^T(\boldsymbol{x}),$$

The input mask $\boldsymbol{u}$ and $\boldsymbol{v}$ are regarded as a $1 \times m$ matrix and a $1 \times n$ matrix respectively. The multiplication denotes matrix multiplication, and T represents matrix transposition.

Linear correlation is used to measure the bias of a linear approximation, which can be efficiently calculated by FWHT.

$$\begin{aligned} c(\boldsymbol{u}, \boldsymbol{v}) &= 2^{-m} \left(|\{\boldsymbol{x} : \boldsymbol{u}\boldsymbol{x}^T \oplus \boldsymbol{v}F^T(\boldsymbol{x}) = 0\}| - |\{\boldsymbol{x} : \boldsymbol{u}\boldsymbol{x}^T \oplus \boldsymbol{v}F^T(\boldsymbol{x}) = 1\}) \right| \\ &= 2^{-m} \sum_{\boldsymbol{x}} (-1)^{\boldsymbol{u}\boldsymbol{x}^T \oplus \boldsymbol{v}F^T(\boldsymbol{x})}. \end{aligned}$$

With m linear independent mask pairs $\boldsymbol{u}_i$, we could construct a linear approximations with dimension m by matrix mask pair (U, V)[22].

$$U\boldsymbol{x}^T \oplus V\boldsymbol{z}^T = \boldsymbol{e}, \tag{1}$$

where the i-th row of U and V are $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ respectively, $F^T(\boldsymbol{x}) = \boldsymbol{z}$, $\boldsymbol{e}$ could be treated as a noise vector.

Walsh-Hadamard Transform. Walsh-Hadamard transform is a spectral tool widely used in cryptanalysis of linear type. Let $p_x = Pr[X = x], x \in \mathbb{F}_{2^m}$ be the probability density function of discrete probability distribution P. Let $X \sim P$ denote a discrete random variable, the Walsh-Hadamard transform of X is defined by

$$\mathcal{W}(X)_w = 2^{-m} \sum_{x \in \mathbb{F}_{2^m}} p_x (-1)^{\boldsymbol{w}\boldsymbol{x}^T}.$$

Remark 1. Notice that finite field $\mathbb{F}_{2^m}$ is a vector space over $\mathbb{F}_2$ with dimension m, then there is a natural bijection from an element $w \in \mathbb{F}_{2^m}$ to vector $\boldsymbol{w} \in \mathbb{F}_2^m$. For convenience, we alternatively use them if there is no ambiguity in the context.

Another type of spectral analysis tool in dealing with Boolean functions is circulant Walsh-Hadamard transform

$$\tilde{\mathcal{W}}(f)_w = 2^{-m} \sum_{\boldsymbol{x} \in \mathbb{F}_2^m} (-1)^{f(\boldsymbol{x}) \oplus \boldsymbol{w}\boldsymbol{x}^T},$$

where f is a Boolean function. Obviously, $c(\boldsymbol{u}, \boldsymbol{v}) = \tilde{\mathcal{W}}(\boldsymbol{v}F^T)_w$.

Since Walsh-Hadamard transform is a linear operator for addition in $\mathbb{F}_{2^n}$, let random variable $X = X_1 \oplus X_2 \oplus \cdots \oplus X_k$, we can efficiently compute probability distribution of X with the help of the convolution property

$$p_x = \mathcal{W}^{-1}(\mathcal{W}(X_1) \times \cdots \times \mathcal{W}(X_k))_x$$

Square Euclid Imbalance. Relative entropy(also called Kullback–Leibler divergence) is used to measure the difference between two probability distributions.

Definition 1. *Let p_x and q_x be probability density functions of two discrete probability distributions P and Q, their relative entropy is defined by*

$$D(p \parallel q) = \sum_x p_x \log \frac{p_x}{q_x}.$$

If p is close to q, i.e. $p_x = q_x + \epsilon(x)$, the relative entropy could be approximated by

$$D(p \parallel q) \approx \frac{1}{2} \sum_x \frac{(p_x - q_x)^2}{q(x)} + O(\epsilon^3(x)).$$

The summation term $\sum_x \frac{(p_x - q_x)^2}{q(x)}$ is usually called capacity, and denoted by $C(p \parallel q)$. Square Euclid imbalance(SEI) is the capacity between a probability distribution and uniform distribution, i.e.

Definition 2. *Let p_x be the probability density function of probability distribution P, Its SEI is defined by*

$$\Delta(p) = 2^m \sum_x (p_x - \frac{1}{2^m})^2 \tag{2}$$

Particularly, let $e = \boldsymbol{u}\boldsymbol{x}^T \oplus \boldsymbol{v}F^T(\boldsymbol{x}) \in \mathbb{F}_2$ denote a binary random noise variable with density function $p_x = 2^{-m}|\{\boldsymbol{x} : e = x\}|$, then obviously the square of correlation $c(e)^2 = \Delta(p)$. For a non-binary random variable $X \in \mathbb{F}_{2^m}$, we have similar result. Let correlation

$$c(w) = \sum_{x:\boldsymbol{w}\boldsymbol{x}^T=0} Pr[X = x] - \sum_{x:\boldsymbol{w}\boldsymbol{x}^T=1} Pr[X = x],$$

The following theorem reveals the relationship between SEI and linear correlation.

Theorem 1 ([6]). *Let $p(x)$ is probability density function as previous, then its SEI*

$$\Delta(p) = \sum_w \hat{\epsilon}^2(w) = \sum_{w\neq 0} c^2(w),$$

where $\epsilon(x) = p(x) - 2^{-m}$, $\hat{\epsilon}(w)$ denotes the Walsh-Hadamard transform of $\epsilon(x)$.

Parity Check and Characteristic Polynomial. Let $M_m(\mathbb{F}_2)$ denote the $m \times m$ matrix ring over $\mathbb{F}_2$. The generator polynomial of LFSR is denoted by

$$L(x) = E + C_1x + C_2x^2 + \cdots + C_dx^d \in M_m(\mathbb{F}_2)[x],$$

where C_d is nonsingular and E is the identity matrix. The number of information bits of $L(x)$ are denoted by k. If $L(x) \in \mathbb{F}_{2^m}[x]$, it can also be mapped into $GL_m(\mathbb{F}_2)[x]$.

A parity check corresponds to an equation which fulfills the LFSR output sequence $\boldsymbol{x}_t$. For example, it is well known that any multiples of $L(x) \in \mathbb{F}_{2^m}[x]$ is a parity check. Usually, only those very sparse parity checks with low degree are exploited in FCA.

Let set $\mathcal{H}_{\tau+1,d}$ denote all parity checks with $\tau + 1$ taps and degree d, abbreviated by $\mathcal{H}$ without ambiguity. Its cardinality is denoted by $|\mathcal{H}_{\tau+1,d}|$. The available parity checks at position n denoted by $H^{(n)} \subseteq \mathcal{H}$. Suppose a parity check for sequence $\boldsymbol{x}_t$ is denoted by

$$G_n\boldsymbol{x}_t + \cdots + G_1\boldsymbol{x}_{t+n-1} + E\boldsymbol{x}_{t+n} = 0, \tag{3}$$

where G_n is nonsingular. Its characteristic polynomial is denoted by

$$F_n(x) = \det(Ex + A) = \det(\sum_{i=0}^{n} G_{n-i}x^i),$$

where T denotes the companion matrix

$$A = \begin{pmatrix} 0 & E & 0 & 0 & \cdots & 0 \\ 0 & 0 & E & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & E \\ G_n & G_{n-1} & G_{n-2} & \cdots & G_2 & G_1 \end{pmatrix}.$$

Cyclotomic Cosets. Let a be a positive integer relatively prime to b. For i and integer, the b-cyclotomic coset modulo a containing i is $CS_i = \{i, ib, \cdots, ib^{r-1}\}$ mod a, where r is the smallest positive integer such that $ib^r \cong i \mod a$. The minimal integer in CS_i is called coset header and denoted by $\bar{i}$. All coset headers form a set $\mathcal{R}_{b,a}$.

2.2 A Brief Description of Original Algorithm

Meier and Staffelbach's original FCA includes a precomputation phase and a decoding phase.

Precomputation Phase. Let LFSR's generator polynomial $L(x) \in \mathbb{F}_2[x]$. The purpose of precomputation phase is finding sufficient very sparse parity checks with low degree, which is a hard open problem. One way recommended by Zeng[36] is evaluating logarithms in finite fields of characteristic 2. It is rather efficient to find low weight multiples, but the degree is not promised to be low. Another way is by extended K-tree algorithm based on general birthday collision[32]. The extended k-tree algorithm can be used to find low weight multiples of polynomial with not so large degree with flexible parameters.

Decoding Phase. The decoding phase targets to recover the initial state of LFSR from key stream. Suppose we have found sufficient suitable parity checks

$$x_n \oplus a_n^{(i)} = 0 \in \mathcal{H},$$

where $a_n^{(i)}$ is the sum of τ different taps $a_n^{(i)} = \sum_{k=1}^{\tau} x_{n-i_k}$. The corresponding check value is $z_n \oplus b_n^{(i)}$, where $b_n^{(i)}$ is the sum of t different z_{n-i_k} corresponding to x_{n-i_k}. The nonlinear part of a stream cipher is modeled as a binary symmetric channel(BSC), the crossover probability is $p_1 = \Pr[x_n \oplus z_n = 1]$. The critical part of decoding phase is calculating a posteriori probability(APP) with priori distribution symbol by symbol. Suppose that the check values are all 0 for a subset $\mathcal{H}_0 \subseteq \mathcal{H}$, then by Bayes' formula,

$$p^* = \frac{p \prod_{i \in \mathcal{H}_0}(1 - s_i) \prod_{i \in \mathcal{H} \setminus \mathcal{H}_0} s_i}{p \prod_{i \in \mathcal{H}_0}(1 - s_i) \prod_{i \in \mathcal{H} \setminus \mathcal{H}_0} s_i + (1 - p) \prod_{i \in \mathcal{H} \setminus \mathcal{H}_0}(1 - s_i) \prod_{i \in \mathcal{H}_0} s_i}$$

where each $s_i = s(p_{i_1}, \cdots, p_{i_t}) = \Pr[a_n^{(i)} = b_n^{(i)}]$ depends on the probability of t symbols involved in parity check i. Moreover, s_i can be calculated recursively in the BSC Model

$$s(p_{i_1}, \cdots, p_{i_t}) = p_{i_t} s(p_{i_1}, \cdots, p_{i_{t-1}}) + (1 - p_{i_t})(1 - s(p_{i_1}, \cdots, p_{i_{t-1}}))$$

The specific process is depicted in Algorithm 1. For more details we refer to the original paper[27].

Algorithm 1 Meier and Staffelbach's Algorithm B

Input: A key stream sequence $\boldsymbol{z}$ of length N, and $\mathcal{H}$.
1. Calculate the probability threshold p_{thr} and quantity threshold N_{thr}.
2. For round $r \in \{1, 2, \cdots\}$
3. For iteration i from 1 to a small integer
4. Calculate APP p^* from priori probability p, assign $p_n^* = p_n$ for all position n.
5. If $N_w \geq N_{thr}$ where $N_w = |\{n|p_n > p_{thr}\}|$, break;
6. Complement the bits of $\boldsymbol{z}$ with $p_n > p_{thr}$.
7. Reset all positions to initial probability p.
8. If $\boldsymbol{z}$ satisfies all parity checks, break.
9. Terminate with $\boldsymbol{x} = \boldsymbol{z}$.

3 Fast Correlation Attack Based on Vectorial Iterative Decoding Algorithm

3.1 Channel Model

Our channel model is symmetric channel(SC) instead of discrete memoryless channel(DMC). The received word is the transmitted word XOR noise, i.e., $\boldsymbol{z} = \boldsymbol{x} \oplus \boldsymbol{e}$. A symmetric channel model has a transition matrix

$$M = \begin{pmatrix} p(z_1|x_1) & p(z_2|x_1) & \cdots & p(z_{2^m}|x_1) \\ p(z_1|x_2) & p(z_2|x_2) & \cdots & p(z_{2^m}|x_2) \\ \vdots & \vdots & \vdots & \vdots \\ p(z_1|x_{2^m}) & p(z_2|x_{2^m}) & \cdots & p(z_{2^m}|x_{2^m}) \end{pmatrix}.$$

Each row is a permutation of another row, and so as to columns. Moreover, the sum of each row equals 1 as the definition of SC. Particularly, the sum of each column equals 1. Therefore, Our symmetric channel can be treated as an extended BSC. Its channel capacity is certainly $C = m - H(\boldsymbol{r})$, where $\boldsymbol{r}$ denotes a row of M.

Suppose we have a linear approximation with dimension m, i.e.,

$$\bigoplus_{i=1}^{r} U_i \boldsymbol{x}_i^T \oplus \bigoplus_{i=1}^{s} V_i \boldsymbol{z}_i^T = \boldsymbol{e}^T. \tag{4}$$

Similarly as FCA based on BSC, the channel noise vector $\boldsymbol{e}^T$ is XORed to $\bigoplus_{i=1}^{r} U_i \cdot \boldsymbol{x}_i^T$, and the output is $\bigoplus_{i=1}^{s} V_i \cdot \boldsymbol{z}_i^T$, see Fig. 3.1.

3.2 Checking Parity with Vectorial Noise

Since the binary linear approximation is generalized to multidimensional linear approximation, the check methods also need to be adjusted for FCA. Suppose that we have $|\mathcal{H}^{(n)}| = r$ check equations in $M_m(\mathbb{F}_2)[x]$ for position n, and $l \in \mathcal{H}^{(n)}$ denote a specific check equation

$$l : E\boldsymbol{x}_n \oplus G_1 \boldsymbol{x}_{n-1} \oplus \cdots \oplus G_n \boldsymbol{x}_{n-d} = \boldsymbol{0}.$$

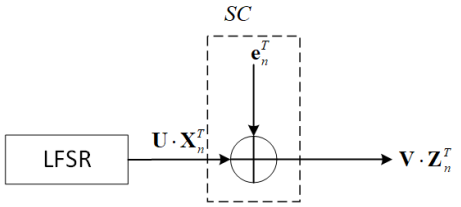

Fig. 1. Channel Model for VFCA

In order to check parity over matrix ring, these $G_k, 1 \leq k \leq n$ are restricted by those coefficients matrices U_j linear approximation (4). More specifically, we require that all $U_i, 1 \leq i \leq r$ are nonsingular. For each G_k, all U_i satisfy that $U_i G_k U_i^{-1} = G'_k$, which implies that if U_i, U_j satisfies $U_i G_k U_i^{-1} = G'_k$ and $U_j G_k U_j^{-1} = G'_k$, then $(U_j^{-1}U_i)G_k(U_j^{-1}U_i)^{-1} = G_k$, i.e., $U_j^{-1}U_i \in C(G_k)$, where $C(G_k)$ denotes the centralizer of G_k in $GL_m(\mathbb{F}_2)$.

For a parity check l we could multiply it with $U_1, U_2, \cdots, U_r$ respectively,

$$U_i(E\boldsymbol{x}_{n+i} \oplus G_1\boldsymbol{x}_{n-1+i} \oplus \cdots \oplus G_d\boldsymbol{x}_{n-d+i}) = \mathbf{0}, i \in \{1, 2, \cdots, r\}.$$

Thus we have

$$E(U_i\boldsymbol{x}_{n+i}) \oplus G'_1(U_i\boldsymbol{x}_{n-1+i}) \oplus \cdots \oplus G'_d(U_i\boldsymbol{x}_{n-d+i}) = \mathbf{0}, i \in \{1, 2, \cdots, r\}.$$

Summing them up, and we have

$$\bigoplus_{i=0}^{d} G'_i(\bigoplus_{j=1}^{r} U_j\boldsymbol{x}_{n-i+j}) = \bigoplus_{i=0}^{d} G'_i(\bigoplus_{j=1}^{s} V_j\boldsymbol{z}_{n-i+j}^T) \oplus \bigoplus_{i=0}^{d} G'_i(\boldsymbol{e}_{n-i}^T)), \tag{5}$$

where $G'_0 = E$. This process can be done for all parity checks in $\mathcal{H}^{(n)}$. The purpose is to determine $\boldsymbol{e}_{n-i}^T$ of each position, when observing $\bigoplus_{j=1}^{s} V_j\boldsymbol{z}_{n-i+j}^T$. Notice that the approach here is generic. When the parity checks and linear approximations have special form, more efficient checking approach is feasible, see section 5.2.

Our FCA doesn't require all $G_i = E, 1 \leq i \leq n$ as in linear distinguishing attack in large alphabets[35], which is expected to have very high degree. For example, the degree of these special parity checks with weight 4 of SNOW 3G is expected to be $O(2^{172})$.

To describe the effect of these parity checks, we divide them into two sets. Let H_I include those parity checks with coefficients all E, while H_{II} includes the rest. We call them type I and type II parities respectively, and we will see that they play different roles in the iterative decoding phase.

3.3 Vectorial Iterative Decoding Algorithm

In this subsection, we consider how to extract information from a noise sequence by vectorial iterative decoding algorithm. Firstly, we try to generalize Meier's original Algorithm B, then improve the iterative criterions.

Let $\boldsymbol{e}_1\boldsymbol{e}_2\cdots\boldsymbol{e}_N$ denote the noise sequence, and $\bigoplus_{j=1}^{s} V_j \boldsymbol{z}_{n-i+j}^T$ denotes the derived sequence from key stream $\boldsymbol{z}_1\boldsymbol{z}_2\cdots\boldsymbol{z}_N$, The initial priori distribution P is the same for each $\boldsymbol{e}_n$, which is derived by linear approximation. Let $p_\zeta^{(n)} = \Pr[\boldsymbol{e}_n = \zeta]$ denote its density function, then the APP $p_\zeta^{*(n)}$ could be computed by Bayes's formula.

$$\begin{aligned} p_\zeta^{*(n)} =& \Pr\left[\boldsymbol{e}_n = \zeta | \text{when observed check values } (c_1, c_2, \cdots, c_h)\right] \\ =& \frac{p_\zeta^{(n)} \prod_{l=1}^{h} \Pr[\bigoplus_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{n-l_i}^T = c_l \oplus I \cdot \zeta]}{\bigoplus_\zeta p_\zeta^{(n)} \prod_{l=1}^{h} \Pr[\bigoplus_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{n-l_i}^T = c_l \oplus I \cdot \zeta]}, \end{aligned} \tag{6}$$

where observed values c_l corresponds to $\bigoplus_{j=1}^{s} V_j \boldsymbol{z}_{n-i+j}^T$.

As ζ run over the alphabet, $\Pr[\sum_{i=1}^{t} G'_{n-l_i} \cdot \boldsymbol{e}_{l_i}^T = c_l + I \cdot \zeta], \zeta \in \mathcal{E}$ can be calculate by convolution property and Fast Walsh-Hadamard Transform. Thus the nominator and denominator can be computed by Algorithm 2.

Algorithm 2 Calculate the nominator

Input: priori p.d $p_\zeta^{(n)}$

1. Let priori probability distribution $\boldsymbol{p}^{(n)} = (p_0, p_1, \cdots, p_{2^m-1})$.
2. For each parity check $l \in \mathcal{L}_n$
3. Calculate p.d $\boldsymbol{p}^l$ of $\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T$ by FWHT and convolution property.
4. Permute $\boldsymbol{p}_c^l \leftarrow \boldsymbol{p}_{c\oplus\zeta}^l$.
5. Multiply corresponding coordinate together of all these vectors.

We always assume $\boldsymbol{e}_{n-l_i}$ are independent, and moreover, all parity checks in $\mathcal{L}_n$ are required to be orthogonal. Now we describe the iterative decoding algorithm as Algorithm 3. The notation $\boldsymbol{E}^{next} \succ \boldsymbol{E}$ means that there is at least one $j \in \{1, 2, \cdots, 2^m - 1\}$ satisfying $E_j^{next} > E_j$, while $\preceq$ has reverse meaning.

The criterions which are used to break up the iterative loop and trigger the reset process are main factors influencing the convergence speed. For more details we refer to [12, 29]. Therefore, we also optimize the criterions by experiments.

Some phenomena are observed in scaled experiments when parity checks are not so many. Firstly, if a threshold is raised to break up loop and reset as Algorithm B, it is easier to be triggered in the earlier rounds than in the later rounds. Secondly, if a complement is performed very early without passing through enough iterations, it will pull the algorithm into a self-combination state too early and weaken the decoding efficiency. To improve this, two main criterions are proposed to break the iterate loop and trigger the reset process.

Criterion 1. Passing through sufficient iterations before breaking up and re-setting, which corresponding to line 7-11 and 14. More specifically, if new ***app*** strengthen the empirical complement effect and iterations is less than maximal, then continue iteration by Bayes's rule. Otherwise, select the complement coin which has potential largest empirical complement effect.

Algorithm 3 Vectorial iterative decoding

Input: A sequence $\boldsymbol{z}'$ of length N derived from key stream,
noise sequence $\boldsymbol{e}$ with initial p.d. $\boldsymbol{p}$,
h different parity checks with weight $t+1$.

parameters: Maximal rounds R, maximal iterations T and minimal gap G to infuse new noise.

1. Initialize a priori p.d. sequence $\boldsymbol{pri}$ of length N all with the same initial p.d. $\boldsymbol{p}$.
2. Initialize a global empirical vector $\boldsymbol{E}^{glb} = (E_1^{glb}, \cdots, E_{2^m-1}^{glb}) \leftarrow \boldsymbol{0}$.
3. For round $r = 1, 2, \cdots, R$ do
4. Initialize a round empirical vector $\boldsymbol{E}^{rnd} = (E_1^{rnd}, \cdots, E_{2^m-1}^{rnd}) \leftarrow \boldsymbol{0}$.
5. Initialize complement coin $c \leftarrow 0$.
6. For iteration $i = 1, 2, \cdots, T$ do
7. Initialize a iteration empirical vector $\boldsymbol{E}^{itr} = (E_1^{iter}, \cdots, E_{2^m-1}^{iter}) \leftarrow \boldsymbol{0}$.
8. For position $n = 1, 2, \cdots, N$ do
9. Compute $\boldsymbol{app}$ from $\boldsymbol{pri}$ by equation (6).
10. If $p_j^{(n)} > p_0^{(n)}$, then $E_j^{itr} \leftarrow E_j^{itr} + 1/N, j \in \{1, 2, \cdots, 2^m - 1\}$.
11. If $\boldsymbol{E}^{itr} \succ \boldsymbol{E}^{rnd}$, then $\boldsymbol{E}^{rnd} \leftarrow \boldsymbol{E}^{itr}$, $\boldsymbol{pri} \leftarrow \boldsymbol{app}$.
12. If $\boldsymbol{E}^{itr} \preceq \boldsymbol{E}^{rnd}$ or $i = T$, then
13. If $\boldsymbol{E}^{itr} = \boldsymbol{0}$, then return failed.
14. Else if $||\boldsymbol{E}^{rnd} - \boldsymbol{E}^{glb}|| < G$, then choose an very biased noise sequence $\boldsymbol{n}$ of length N, reset $\boldsymbol{z}' \leftarrow \boldsymbol{z}' \oplus \boldsymbol{n}$, break up current loop.
15. Else then $\boldsymbol{E}^{glb} \leftarrow \boldsymbol{E}^{rnd}$, select c s.t $E_c^{rnd} + E_c^{itr}, c \in 1, 2, \cdots, 2^m - 1$ is maximal, break up current loop.
16. If $c \neq 0$, then complement all positions of $\boldsymbol{z}'$ such that $p_c > p_0$ with c.
17. If $\boldsymbol{z}'$ satisfies all parity checks, then return success.
18. Reset a priori p.d. sequence $\boldsymbol{pri}$ initial p.d. $\boldsymbol{p}$.
19. Terminate.

Criterion 2. When the empirical complement effect is weak from the previous round to current round, a very biased noise sequence is infused hoping to break the tie caused by self-combination property of LFSR. the noise's SEI is required to be appropriate, neither too large to counteract the previous decoding work, nor too small to break the tie.

criterion 1 is easy to understand. Regardless of the front and back rounds, sufficiently iteration are needed to correct more positions before complements to avoid converging to self-combination state too early. The idea behind criterion 2 is simple but novel. After many rounds, the complement would correct very few positions because of the self-composition property of LFSR. Therefore, a slightly noise are XORed to the indeterminate middle sequence Z' to get out of the trap.

The complement in algorithm 3 operates on derived sequence $\boldsymbol{z}' = \bigoplus_{i=1}^{s} V_i \boldsymbol{z}_i$, The n-th position$\boldsymbol{z}'_n$ is changed to $\boldsymbol{z}'_n \oplus \boldsymbol{e}_n$ when the noise is determined to be $\boldsymbol{e}_n$ and the complement is implemented. If $\boldsymbol{z}'$ satisfies all parity checks at the end, we just deduce that all $\boldsymbol{e}_i^T = \boldsymbol{0}$. Then with the help of LFSR's feedback polynomial, the initial state of LFSR can be recovered.

4 Cryptographic Properties and Experimental Results

4.1 Statistical Model

Convergence Property. It is necessary to figure out the convergence property when iteratively computing APP. Intuitively, we hope that APP $p_\zeta^{*(n)}$ increases when noise variable $e_n = \zeta$ and decreases when $e_n \neq \zeta$. Its expected value is computed as follows.

$$\begin{aligned} E_0[p_\zeta^{*(n)}] =& E[p_\zeta^{*(n)} | e_n = \zeta] \\ =& \sum_{(c_1,c_2,\cdots,c_h)} \frac{p_\zeta^{(n)} \left(\prod_{l=1}^{h} \Pr[\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T = c_l + I \cdot \zeta] \right)^2}{\sum_\zeta p_\zeta^{(n)} \prod_{l=1}^{h} \Pr[\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T = c_l + I \cdot \zeta]}, \\ E_1[p_\zeta^{*(n)}] =& E[p_\zeta^{*(n)} | e_n \neq \zeta] \\ =& \sum_{\zeta' \neq \zeta} \sum_{(c_1,c_2,\cdots,c_h)} \frac{p_\zeta^{(r)} \prod_{l=1}^{h} \Pr[\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T = c_l + I \cdot \zeta]}{\sum_\zeta p_\zeta^{(n)} \prod_{l=1}^{h} \Pr[\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T = c_l + I \cdot \zeta]} \\ & \frac{p_{\zeta'}^{(r)} \prod_{l=1}^{h} \Pr[\sum_{i=1}^{t} G'_{l_i} \cdot \boldsymbol{e}_{l_i}^T = \epsilon_l + I \cdot \zeta']}{1 - p_\zeta^{(r)}}. \end{aligned}$$

And we conclude that $E[p^{*(n)}] = p_\zeta E_0[p^{*(n)}] + (1 - p_\zeta) E_1[p^{*(n)}] = p_\zeta$.

Example 1. Exploiting 3 type I parity checks with 3 taps, we get the increasing and decreasing ratios in Table 1. The second row is priori probability P. $E_0[p^*]/[p^*]$ and $E_1[p^*]/[p^*]$ denote the increasing and decreasing ratio. Particularly, E'_0/p^* and E'_1/p^* denote the case only considering the number of holding parity checks. Both cases meet our expectation.

Table 1. An example of increasing and decreasing ratio

x	0	1	2	3
p_x	0.4500	0.2500	0.2000	0.1000
E'_0/p^*	1.02618712	1.00117564	1.02744428	1.10462318
E'_1/p^*	0.97857418	0.99960812	0.99313893	0.98837520
E_0/p^*	1.03907892	1.06836181	1.16004050	1.19334394
E_1/p^*	0.96802634	0.97721273	0.95998988	0.97851734

Decoding Efficiency. In algorithm 1, a threshold N_{thr} is computed to promote the efficiency of complement. It is determined by the intersection point of two shrunk normal distributions, and it reflects the correcting ability of the first iteration. In the multidimensional case, the intersection point becomes a intersection curve(surface). Now we discuss how to estimate the correcting ability by measuring the volume of the intersection area.

Let N_ζ^{thr} denote this threshold corresponding to $\zeta \in \{1, 2, \cdots, 2^m - 1\}$. Without loss of generality, we assume that the priori probability distribution P of noise sequence $\boldsymbol{e}_1 \cdots \boldsymbol{e}_N$ s.t.

$$p_0 \geq p_1 \geq \cdots \geq p_{2^m-1} > 0. \tag{7}$$

For two independent random variables $X \sim P$ and $Y \sim P$, the probability distribution $Q_{\beta_1 X \oplus \beta_2 Y}$ of their linear combination $\beta_1 X \oplus \beta_2 Y, \beta_1, \beta_2 \in \mathbb{F}_{2^m}^*$ still has 0 as maximal value point, which could be deduced from the convolution property and Walsh-Hadamard transform. Particularly, if $\beta_1 = \beta_2 = 1$, $Q_{X \oplus Y}$ preserves the order

$$q_0 \geq q_1 \geq \cdots \geq q_{2^m-1} > 0.$$

The approach to calculate N_ζ^{thr} is inspired by the fact p_ζ^* is large when more check values appear to be ζ. Let $q_c = \Pr[\sum_{i=1}^t G'_{l_i} \cdot \boldsymbol{e}_{n-l_i}^T = c]$ denotes the probability that the t taps sum to be c for check equation l. Obviously, q_c depends on the individual parity check since their coefficients G'_{l_i} may be different. This phenomenon makes it very complicated to calculate the threshold N_ζ^{thr}. To simplify the calculation, we divide all parity checks into two sets $\mathcal{H}_I$ and $\mathcal{H}_{II}$ according to its coefficients, then deal with them separately.

The set $\mathcal{H}_I$ includes all parity checks with all identity coefficients. For this class, q_c is obviously independent of parity checks. Let $|\mathcal{H}_I| = h_I$, the probability the current noise $\boldsymbol{e} = \zeta$ and x_i check values equal $i, i \in \{0, \cdots, 2^m - 1\}$ is as follows.

$$p_\zeta q(x_0, x_1, \cdots, x_{2^m-1}, \zeta) = p_\zeta \frac{h_I!}{x_0! x_1! \cdots x_{2^m-1}!} \prod_{i=0}^{2^m-1} q_{i \oplus \zeta}^{x_i}, \tag{8}$$

where $x_{2^m-1} = h_I - \sum_{i=0}^{2^m-2} x_i$.

Obviously, random vector $\boldsymbol{x} = (x_0, x_1, \cdots, x_{2^m-1})$ follows multinomial distribution Multi$(h_I, \boldsymbol{q}_\zeta)$ with parameter $\boldsymbol{q}_\zeta = (q_\zeta, q_{1 \oplus \zeta}, \cdots, q_{2^m-1 \oplus \zeta})$. Its density

function are denoted by $q(\boldsymbol{x}, \zeta)$. For convenience, we introduce notations

$$\boldsymbol{q}_{\zeta}^{\boldsymbol{x}} = \prod_{i=0}^{2^m-1} q_{i\oplus\zeta}^{x_i}, \binom{h_I}{\boldsymbol{x}} = \frac{h_I!}{x_0!x_1!\cdots x_{2^m-1}!}.$$

Let $\mathcal{A}$ be a subset of all possible random vector $\boldsymbol{x}$. Once we complement those noises corresponding to $\mathcal{A}$ in the iterative process, the number of correctly changed noises and erroneously changed noises are respectively

$$NW(p, \mathcal{A}, \zeta) = N \sum_{\boldsymbol{x}\in\mathcal{A}} p_{\zeta} q(\boldsymbol{x}, \zeta), NW(p, \mathcal{A}, 0) = N \sum_{\boldsymbol{x}\in\mathcal{A}} p_0 q(\boldsymbol{x}, 0), \tag{9}$$

where N denote the length of data. All the other cases of changing are neutral. Thereby, the number of actual corrected positions is the difference

$$NI(p, \mathcal{A}, \zeta, 0) = NW(p, \mathcal{A}, \zeta) - NW(p\mathcal{A}, 0). \tag{10}$$

Given P and $\mathcal{H}_I$, if we can find a set $\mathcal{A}$ maximizing $I(p, \mathcal{A}, \zeta, 0)$, then the expected number of actual corrected positions of each complement should be maximized. Firstly, we observe that the means of the two multinomial distributions are respectively

$$E_{p(\boldsymbol{x},\zeta)}(\boldsymbol{x}) = h_I \boldsymbol{q}_{\zeta}, E_{p(\boldsymbol{x},0)}(\boldsymbol{x}) = h_I \boldsymbol{q}_0.$$

Therefore, similar as the binomial case, there is a set A of $\boldsymbol{x}$ in which $I(p, \mathcal{A}, \zeta, 0)$ takes non-negative value.

Since given $\boldsymbol{x}$, $I(p, \mathcal{A}, \zeta, 0)$ and $p_{\zeta}^* - p_0^*$ have the same sign, it is equivalent to find $\mathcal{A}$ such that $p_{\zeta}^* - p_0^* > 0$ for each $\boldsymbol{x} \in A$, that is to determine the region $\mathcal{A}$ such that

$$\delta(\zeta, 0) = p_{\zeta} q(\boldsymbol{x}, \zeta) - p_0 q(\boldsymbol{x}, 0) > 0, \boldsymbol{x} \in \mathcal{A}. \tag{11}$$

Example 2. Let initial probability distribution P be the same as in Example 1, and $h_I = 15$. The difference $\delta(\zeta, 0)$ is illustrated in Fig. 2. The non-negative and the negative area are separated. The size of circle represents the relative absolute value of the difference $\delta(\zeta, 0)$.

When h is small, it is feasible to evaluate N_{ζ}^{thr} by exhaustively searching. The threshold N_{ζ}^{thr} can be determined by

$$N_{\zeta}^{thr} = N(\sum_{\boldsymbol{x}\in A} \sum_{c\in\mathcal{E}} p_c q(\boldsymbol{x}, c)). \tag{12}$$

The time complexity is about $O(2^m \binom{h+2^m}{2^m})$.

When h is large and $\boldsymbol{q}$ is not near the boundary of the parameter space, multivariate normal distribution approximation is suitable. Multi$(h_I, \boldsymbol{q})$ could be approximated by $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with density function

$$\frac{1}{\sqrt{(2\pi)^{2^m-1}|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})^T\right),$$

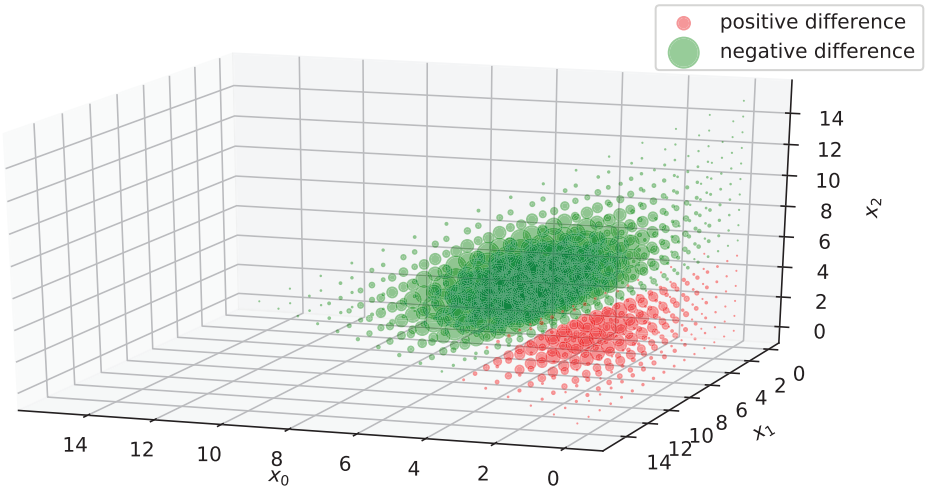

Fig. 2. Example for the difference distribution

where mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ are determined by Multi$(h_I, \boldsymbol{q})$. Therefore, the area $\mathcal{A}$ maximizing the multiple integral

$$I(p, \mathcal{A}, \zeta, 0) \approx \int_{\mathcal{A}} (p_\zeta \mathcal{N}(\boldsymbol{\mu}_\zeta, \boldsymbol{\Sigma}_\zeta) - p_0 \mathcal{N}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)) d\boldsymbol{x} \tag{13}$$

should be part of a hypercube with dimension $2^m - 2$ that restricted by the $2^m - 1$ coordinate plane and two surfaces

$$\begin{aligned} \Omega_1 &: \sum_{i}^{2^m-2} x_i = h, \\ \Omega_2 &: \frac{1}{2}((\boldsymbol{x} - \boldsymbol{\mu}_0)\boldsymbol{\Sigma}_0^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_0)^T) - \frac{1}{2}((\boldsymbol{x} - \boldsymbol{\mu}_\zeta)\boldsymbol{\Sigma}_\zeta^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_\zeta)^T) - \ln\frac{p_0}{p_\zeta} = 0. \end{aligned} \tag{14}$$

Notice that Ω_2 is a quadratic form in the real field, the multiple integral (13) can be computed by repeated integral. Once $\mathcal{A}$ is determined, the threshold can be calculated by volume integral

$$N \sum_{c \in \mathcal{E}} \int_{\mathcal{A}} \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c) d\boldsymbol{x}. \tag{15}$$

Example 3. Let the probability distribution P be the same as in Example 1. To illustrate this multivariate normal approximation, The results of $I(p, \mathcal{A}, 1, 0)$ computed by two methods id depicted in Table 2. In order to simplify the integral, we could even slightly adequate the boundary of A without fluctuating the result much.

Table 2. Direct computation and normal approximation for $I(p, \mathcal{A}, 1, 0)$

number of equations h	40	80	200	400
direct computation	0.0686	0.1138	0.1835	0.2266
normal approximation	0.0707	0.1148	0.1841	0.2267

When the parity checks stem from $H_{II} = \mathcal{H} \backslash \mathcal{H}_I$, q_c depends on individual parity check. To avoid this drawback, when the probability value peak is q_0, we introduce a symmetric multinomial probability distribution Q' to simulate the influences of type II parity checks, which parameter is

$$q'_0 = q_0, q'_1 = \cdots = q'_{2^m-1} = \frac{1 - q'_0}{2^m - 1}. \tag{16}$$

Then the calculation is similar as for $\mathcal{H}_I$. According to the size of $\mathcal{H}_I$ and $\mathcal{H}_{II}$, we could estimate N_ζ^{thr} by combine $\mathcal{H}_I$ and $\mathcal{H}_{II}$ together. The multinomial distribution is replaced by $\text{Multi}(h_I, \boldsymbol{q}_\zeta)\text{Multi}(h_{II}, \boldsymbol{q}'_\zeta)$ in this case.

Example 4. To verify the validity of these approximations, under the same P as Example 1, we compute the theoretical ratio of N_ζ^{thr}/N and the empirical ratio by the ratio where $p_\zeta^* > p_0^*$. Table 3 depicts that our estimations are rather appropriate.

Table 3. Theoretical and empirical value of N_ζ^{thr}/N

No. of parities		theoretical	empirical		
(h_I, h_{II})	ζ		$N = 2^{19}$	$N = 2^{20}$	$N = 2^{21}$
(36,0)	1	0.277133	0.227242	0.250517	0.264012
	2	0.253926	0.242359	0.246835	0.249339
	3	0.200412	0.164480	0.181245	0.190250
(18,18)	1	0.297959	0.251286	0.270056	0.279394
	2	0.260769	0.220915	0.238914	0.248543
	3	0.167968	0.125576	0.144096	0.154273
(0,138)	1	0.376058	0.360392	0.364783	0.368026
	2	0.325561	0.321800	0.332389	0.338674
	3	0.221771	0.198662	0.213513	0.221388

4.2 Information Theory Properties

In this subsection, we discuss some properties from the point view of information theory. Suppose the noises are independent and the parity checks are linear

independent, the relative entropy between Multi$(h, \boldsymbol{q}_0)$ with density function $q(\boldsymbol{x})$ and Multi$(h, (2^{-m}, \cdots, 2^{-m}))$ with density function $u(\boldsymbol{x})$ is

$$D(q \parallel u) = H(q, u) - H(q) = h \sum_{i=0}^{2^m-1} q_i \log \frac{q_i}{2^{-m}} = h(m - H(\boldsymbol{q}_0)).$$

That is the relative entropy is the number of parity checks times the SEI of probability distribution Q.

Secondly, we hope that the right corrected positions are as many as possible in the complement process. Now we think about the sum of relative entropy between Multi$(h, \boldsymbol{q}_c)$ and Multi$(h, \boldsymbol{q}_0)$ for all $c \neq 0$, and we have

Proposition 1. *Let $q_c(\boldsymbol{x})$ and $q_0(\boldsymbol{x})$ be density functions of Multi$(h, \boldsymbol{q}_c)$ and Multi$(h, \boldsymbol{q}_c)$ respectively, then*

$$\sum_{c \neq 0} D(q_c(\boldsymbol{x}) \parallel q_0(\boldsymbol{x})) = -h \log \prod_{i=0}^{2^m-1} q_i - h2^m H(\boldsymbol{q}_0).$$

Proof. By equation (4.2),

$$\begin{aligned}
\sum_{c \neq 0} D(q_c(\boldsymbol{x}) \parallel q_0(\boldsymbol{x})) &= \sum_{c \neq 0} h(\sum_{i=0}^{2^m-1} q_{i \oplus c} \log q_{i \oplus c} - \sum_{i=0}^{2^m-1} q_{i \oplus c} \log q_i) \\
&= - h(2^m - 1)H(\boldsymbol{q}_0) - h \sum_{i=0}^{2^m-1} \sum_{c \neq 0} q_{i \oplus c} \log q_i \\
&= - h(2^m - 1)H(\boldsymbol{q}_0) - h \sum_{i=0}^{2^m-1} (1 - q_i) \log q_i \\
&= - h \log \prod_{i=0}^{2^m-1} q_i - h2^m H(\boldsymbol{q}_0).
\end{aligned}$$

This tells us when the probability distribution of noise approaches uniform distribution, the total relative entropy converges to 0.

4.3 Complexity Analysis

Firstly, given the SEI $\Delta(p)$, To transmit k bits information through an SC channel, the code rate $k/N < \Delta(p)/(2\ln(2))$ by Shannon's Theorem. On the other hand, the number of parity checks h influences the decoding complexity. We focus on the property of the first iteration in the first round, which seems to be the critical part by previous section, and discuss how to deduce some theoretical bounds for h as well as key stream length N.

A Bound from Decoding Codes. Similarly as Proposition 1 in [9], In order to perform an error corrected iterative decoding, the lower bounds of h should satisfy that there exists at least a ζ such that $p_\zeta^* > p_0^*$. It is summarized as follows.

Proposition 2. *If iterative decoding is feasible, then there is at least one* $\zeta \in \{1, 2, \cdots, 2^m - 1\}$ *such that* $p_\zeta q(\boldsymbol{x}, \zeta)/(p_0 q(\boldsymbol{x}, 0)) > 1$. *Particularly, when* P, Q *and* Q' *are multinomial probability distributions as before, then* $\zeta = 2^m - 1$ *and*

$$\frac{p_\zeta}{p_0} > \left(\frac{q_\zeta}{q_0}\right)^{h_I} \left(\frac{q'_\zeta}{q'_0}\right)^{h_{II}}. \tag{17}$$

Proof. Since if $p_\zeta q(\boldsymbol{x}, \zeta)/(p_0 q(\boldsymbol{x}, 0)) \leq 1$ holds for all ζ, then p_i^* converges to 0 or becomes ambiguous during the iterations, i.e. $p_0^* = p_i^*$ is one of the largest. The decoding algorithm won't work.

Particularly, when the probability values of P and Q(or Q') are in order as stated before, and all values of parity checks are ζ, obviously we have

$$\frac{p_\zeta \boldsymbol{q}_\zeta^{\boldsymbol{x}}}{p_0 \boldsymbol{q}_0^{\boldsymbol{x}}} \leq \frac{p_\zeta q_\zeta^{h_I} q_\zeta^{h_{II}}}{p_0 q_0^{h_I} q_0^{h_{II}}}.$$

Therefore, the result follows.

Remark 2. Though the ratio $\eta(\zeta, 0)$ has large value when all check values are ζ, The lower bound for h given in Proposition 2 may be loose, as the probability that all check values are ζ is small.

A lower bound for N could be derived through Proposition 2. For example, when generator polynomial $L(x) \in \mathbb{F}_{2^m}[x]$, the number of parity checks h and the key stream length N shall satisfy that

$$\binom{N}{\tau}(2^m - 1)^\tau \approx h 2^k.$$

As an application of Proposition 2, we give two formulas of h for two important probability distributions. Since when $\Delta(\boldsymbol{e}) = 2^{-\gamma}$, it is expected that there is a probability value around $2^{-m} \pm 2^{-\frac{m+\gamma}{2}}$ in practice[35], the distributions P and P' in Table 4 is very likely to appear, where ϵ denotes $(1 - 2^{-m} - 2^{-\frac{m+\gamma}{2}})/(2^m - 1) - 2^{-m}$.

Table 4. Two probability distributions P and P'

x	0	$1 \cdots i-1$	i	$i+1 \cdots 2^m - 1$
$p_x - 2^{-m}$	$2^{-\frac{m+\gamma+1}{2}}$	$0 \cdots 0$	$-2^{-\frac{m+\gamma+1}{2}}$	$0 \cdots 0$
$p'_x - 2^{-m}$	$2^{-\frac{m+\gamma}{2}}$	$\epsilon \cdots \epsilon$	ϵ	$\epsilon \cdots \epsilon$

By Taylor's formula, we have

$$\frac{p_i}{p_0} = \frac{1-2^{\frac{m-\gamma-1}{2}}}{1+2^{\frac{m-\gamma-1}{2}}} \approx 1-2^{\frac{m-\gamma+1}{2}}, \frac{p'_i}{p'_0} = \frac{1+\epsilon 2^m}{1+2^{\frac{m-\gamma}{2}}} \approx 1-\frac{2^m+1}{2^m-1}2^{\frac{m-\gamma}{2}}.$$

Furthermore, by the convolution property, when each parity check has $\tau+1, t \geq 2$ taps, we have

$$\frac{q_i}{q_0} = \frac{1-2^{-\frac{(t-2)m+t(\gamma-1)+2}{2}}}{1+2^{-\frac{(t-2)m+t(\gamma-1)+2}{2}}} \approx 1-2^{-\frac{(t-2)m+t(\gamma-1)}{2}}.$$

Hence, by Proposition 2, the number of type I and II parity checks for P are

$$\begin{aligned} &1-2^{\frac{m-\gamma+1}{2}} \geq (1-2^{-\frac{(t-2)m+t(\gamma-1)}{2}})^{h_I} \Rightarrow h_I \geq 2^{\frac{(t-1)(m+\gamma-1)}{2}} \\ &1-2^{\frac{m-\gamma+1}{2}} \geq (1-(\frac{2^m}{2^m-1})2^{-\frac{(t-2)m+t(\gamma-1)+2}{2}})^{h_{II}} \Rightarrow h_{II} \geq 2^{\frac{(t-1)(m+\gamma-1)+2}{2}}, \end{aligned} \tag{18}$$

where $\frac{2^m}{2^m-1} \approx 1$.

For the case of P', the general term formula of distributions convolution could be deduced by its recursion formula, i.e.

$$p'_0 = 2^{-m} + \frac{2^{m(t-1)}}{(2^m-1)^{t-1}}2^{-\frac{m+\gamma}{2}t}, q'_i = 2^{-m} - \frac{2^{m(t-1)}}{(2^m-1)^t}2^{-\frac{m+\gamma}{2}t}.$$

Thus we have

$$\frac{q'_i}{q'_0} \approx 1-\frac{2^{m(t+1)}}{(2^m-1)^t}2^{-\frac{m+\gamma}{2}t}, \frac{p'_i}{p'_0} \approx 1-\frac{2^{2m}}{2^m-1}2^{-\frac{m+\gamma}{2}},$$

which means

$$\begin{aligned} &1-\frac{2^{2m}}{2^m-1}2^{-\frac{m+\gamma}{2}} \geq \left(1-\frac{2^{m(t+1)}}{(2^m-1)^t}2^{-\frac{m+\gamma}{2}t}\right)^h \\ &\Rightarrow h \geq \left(\frac{2^m-1}{2^m}\right)^{t-1} 2^{\frac{m+\gamma}{2}(t-1)}. \end{aligned} \tag{19}$$

Notice that type I and II parities are not distinguished in the case of P'.

We expected that the practical bound is between those deduced by P and P'. The FCA mainly benefits from the increased SEI. More specifically, according to Theorem 1, there are 2^m-1 binary linear approximations contributing to the SEI of linear approximation with dimension m.

A Bound from the Practical Corrected Errors. In this part, we discuss how to deduce a bound from the number of expected positions with $p^*_\zeta > p^*_0, \zeta \neq 0$.

Let us consider the sets $\mathcal{A}(i), i \in \{1, 2, \cdots, 2^m - 1\}$ for multinomial distributions. Since $\mathcal{A}(i)$ may intersect with each other, the way of computing threshold in section 4.1 can't be directly applied. Thereby, we introduce some new sets

$$\mathcal{A}'(i) = \mathcal{A}(i) - \mathcal{A}(i) \cap \bigcup_{j=1}^{i-1} \mathcal{A}(i),$$

That is $\mathcal{A}(i)$ excluding all elements that are included in previous sets $\mathcal{A}(i), i \in \{1, 2, \cdots, i\}$. Let M'_i denote the summation of probability values over set $\mathcal{A}'(i)$, more specifically,

$$\sum_{\zeta=1}^{2^m-1} M'_\zeta = \sum_{\zeta=1}^{2^m-1} p_\zeta \sum_{\boldsymbol{x} \in \mathcal{A}'(\zeta)} q(\boldsymbol{x}, \zeta). \tag{20}$$

It is reasonable to require that $\sum_{\zeta=1}^{2^m-1} M'_\zeta > 1$ after the first iteration. Then the succeeding iterations may trigger more positions with $p^*_\zeta > p^*_0$. This phenomenon may be the main advantage that soft decision decoding algorithms have.

Summing up the probability values in multinomial distributions is inconvenient. Though multivariate normal distribution approximation could also be used as before when h is large, the integral may not be easy to evaluate in practice, as the integral area $\mathcal{A}'(\zeta)$ is very complicated. Since symmetric distribution Q' simulates the iterative process very well, we could deduce boundaries for $\mathcal{A}'_\zeta$ using Multi$(h, \boldsymbol{q}')$. The following results shows how to estimate M'_ζ in this case.

Proposition 3. *For multinomial probability distribution Multi$(h, \boldsymbol{q}')$, we have*

$$M'_\zeta = \sum_{l=h_b}^{h} \binom{h}{l} (1 - \sum_{i=0}^{\zeta} q'_{i\oplus\zeta})^{h-l} \sum_{(x_0,\cdots,x_\zeta)\in\mathcal{B}(\zeta)} \binom{l}{x_0, \cdots, x_\zeta} \prod_{i=0}^{\zeta} q'^{x_i}_{i\oplus\zeta}, 1 \le \zeta < 2^m,$$

where $\mathcal{B}_\zeta$ is constrained by $\sum_{i=1}^{\zeta} x_i = l$, $x_\zeta - x_0 \ge h_b$ and $x_i - x_0 \le h_b, 1 \le i < \zeta$.

Particularly, when $\sum_{i=0}^{\zeta} q'_{i\oplus\zeta}$ is small and $hq'_i \le h_b$, the expected number of positions with $p^*_\zeta > p^*_0$ in the first iteration are dominated by when l is small.

Proof. Since $q'_1 = \cdots = q'_{2^m-1}$, we have

$$\begin{aligned} M'_\zeta &= \sum_{\boldsymbol{x}\in\mathcal{A}'_\zeta} \binom{h}{\boldsymbol{x}} \boldsymbol{q}'^{\boldsymbol{x}}_\zeta \\ &= \sum_{l=h_b}^{h} \binom{h}{l} (1 - \sum_{i=0}^{\zeta} q'_{i\oplus\zeta})^{h-l} \sum_{(x_0,\cdots,x_\zeta)\in\mathcal{B}(\zeta)} \binom{l}{x_0, \cdots, x_\zeta} \prod_{i=1}^{\zeta} q'^{x_i}_{i\oplus\zeta}. \end{aligned}$$

By Proposition 2, we deduce that there is a minimal positive integer h_b such that $\delta(\zeta, 0) > 0$ when $x_\zeta - x_0 \le h_b$. Furthermore, $x_i - x_0 < h_b$ should holds to exclude the points in $\mathcal{A}'(i)$ for all $0 < i < \zeta$. Therefore, when $p^*_\zeta > p^*_0$,

$(x_0, \cdots, x_\zeta) \in \mathcal{A}'(\zeta)$ must satisfy that

$$\begin{cases} x_i \geq 0, & 0 \leq i \leq \zeta, \\ x_i - x_0 < h_b, & 0 < i < \zeta, \\ x_\zeta - x_0 \geq h_b, \\ x_0 + \cdots + x_\zeta < h\ . \end{cases}$$

When h is not small and $\sum_{i=0}^{\zeta} q'_{i\oplus\zeta}$ is not high, multidimensional distribution $\mathrm{Multi}(h, \boldsymbol{q}'_\zeta)$ could be approximated by $\zeta+1$ independent Poisson distributions with means $\lambda_{i\oplus\zeta} = hq'_{i\oplus\zeta}$, i.e.

$$\Pr(X = \boldsymbol{x}) \approx \sum_{\mathcal{A}'(\zeta)} \prod_{i=0}^{\zeta} \frac{\lambda_{i\oplus\zeta}^{x_i}}{x_i!} e^{-\lambda_{i\oplus\zeta}} = \frac{\lambda_0^{x_\zeta}}{x_\zeta!} e^{-\lambda_0} \frac{\lambda_\zeta^{x_0+\cdots+x_{\zeta-1}}}{x_0! \cdots x_{\zeta-1}!} e^{-\zeta\lambda_\zeta}. \tag{21}$$

As $\lambda_i \leq h_b$, the maximal value of $\Pr(X = \boldsymbol{x})$ is when $\sum_{i=0}^{\zeta} x_i$ is small, i.e. when l is small. Therefore, the result follows.

Proposition 3 gives us a hint that the value corresponding small l dominate M'_i. When ζ is not very large, M'_ζ could be approximated by partial summation for small l close to the boundary. Obviously, $M'_i, 0 < i$ are monotone non-increasing sequence.

When $\zeta = 1$, there is another elegant way to estimate M'_1 by Skellam distribution. Let $Y_0 \sim \mathrm{Pois}(\lambda_1)$ and $Y_1 \sim \mathrm{Pois}(\lambda_0)$, we know that their difference $K = Y_1 - Y_0$ follows Skellam distribution with following probability density function.

$$p(k, \lambda_\zeta, \lambda_0) = e^{-\lambda_\zeta - \lambda_0} \left(\frac{\lambda_\zeta}{\lambda_0}\right)^{k/2} I_{|k|}(2\sqrt{\lambda_1\lambda_0}),$$

where $I_{|k|}$ is the modified Bessel function of the first kind. Obviously, $M'_1 = Np_\zeta \Pr(K > h_b)$ since a boundary line is $x_1 \geq x_0 + h_b$ by Proposition 3.

On Sparse Check Equations. Since sparse parity checks have large advantages while checking parity, we are interested in these parity checks with $\tau = 1$ or 2. In this section, we give some miscellaneous observations about them.

Let $\boldsymbol{x}_t = (x_{t+c_1}, x_{t+c_2}, \cdots, x_{t+c_m}), c_1 < \cdots < c_m$ denotes the output at time t of LFSR with generator polynomial $L(x) \in M_m(\mathbb{F}_2)[x]$. Each coordinate sequence is a m-sequence $(x_1 x_2 \cdots)$ left shifting c_i times, and its minimal polynomial $f(x) \in \mathbb{F}_2[x]$ has degree k. Particularly, when $(c_1, c_2, \cdots, c_m)$ satisfies special condition, it becomes an LFSR over extension field $\mathbb{F}_{2^m}$[19].

Though parity checks with two taps have very large advantages, unfortunately, the existence of them is a problem by the following direct observations.

Proposition 4. *Let $\boldsymbol{x}_t = (x_{t+c_1}, x_{t+c_2}, \cdots, x_{t+c_m})$ be as stated above, we have*

- *If $c_m - c_1 + m - 1 < k$, then there is no parity check with $\tau = 1$.*

– *Given two parity checks with* $\tau = 1$,

$$G\boldsymbol{x}_t + E\boldsymbol{x}_{t+d_1} = 0, G'\boldsymbol{x}_t + E\boldsymbol{x}_{t+d_2} = 0,$$

if $d_1 = d_2$ *and* $\boldsymbol{x_t}$ *run over all values in* $\{1, \cdots, 2^m - 1\}$, *then* $G = G'$. *If* $d_1 \neq d_2$, *then* $\gcd(d_1, d_2) > k - m$.

Proof. 1. Let $G\boldsymbol{x}_t + E\boldsymbol{x}_{t+d} = 0$ be a parity check. Since i-th row of A and E forms an check polynomial f_i with nonzero constant for x_t, then $f|f_i$. As G is nonsingular, there must be two different check polynomials $f_i(x)$ and $f_j(x)$. That means $f_i + f_j$ also forms a check polynomial, but $c_m - c_1 + m - 1 < k$ means a polynomial with degree less than k could be deduced, which is impossible.

2. When $d_1 = d_2$, it is deduced that $(G + G')\boldsymbol{x}_t = 0$ for all $\boldsymbol{x}_t$, When $\boldsymbol{x_t}$ run over all values in $\{1, \cdots, 2^m - 1\}$, then we have $G = G'$.

When $d_i < d_j$, we could deduce a linear dependent parity check,

$$G_{n,i}(E\boldsymbol{x}_t + G_{n,i}^{-1}G_{n,j}\boldsymbol{x}_{t+i-j}) = 0.$$

Therefore, according to Euclid long division algorithm, we finally have

$$E\boldsymbol{x}_t + G\boldsymbol{x}_{t+gcd(i,j)} = 0.$$

Since there are k information bit of LFSR, then $gcd(i, j) \geq k - m$. Therefore, the result follows.

This observation means parity checks with $\tau = 1$ may be rare, but it doesn't mean none, even though the key stream length needed may be large. For example, $(x_{t+c_1}, x_{t+c_2}, \cdots, x_{t+c_m})$ may be only in a subspace of $\mathbb{F}_2^m$, and $c_m - c_1 + m - 1$ may be large. Once a parity check is found, more could be constructed by sliding and adding together. For example,

$$G\boldsymbol{x}_t + E\boldsymbol{x}_{t+d} = 0 \Rightarrow G^2\boldsymbol{x}_{t-d} + E\boldsymbol{x}_{t+d} = 0.$$

Moreover, if a parity check satisfies sequence $\boldsymbol{x}_t$, then its characteristic polynomial $F_n(x) \in \mathbb{F}_2[x]$ has $f(x)$ as a factor. Since $G_n = G, G_1 = \cdots = G_{n-1} = 0$, then $F_n(x) = \det(Ex^n + G)$, the choices for matrix G and n are $(N/m - 1)|GL_m(\mathbb{F}_2)|$. Let $\mathcal{S} = \{F_n(x) : 1 \leq nm \leq N\}$ denote all possible characteristic polynomials. For convenience, we introduce a map sending $F_n(x) \in \mathcal{S}_G$ to $\mathbb{F}_2[x]$.

$$\phi : \mathcal{S}_G \to \mathbb{F}_2[x]$$
$$F_n(x) = \det(Ex^n + G) \to F(x) = \det(Ex + G).$$

Since $F(x)$ is the characteristic polynomial of invertible matrix G, the number of different $F(x)$ is 2^{m-1}. Suppose that $F(x) = f_1^{n_1} \cdots f_v^{n_v}$, where the f_i are distinct irreducible polynomials of degree d_i, it has been proved that the number of G with given $F(x)$ is $\eta(F(x))$[17], i.e.

$$\eta(F(x)) = \frac{2^{m^2-m}\prod_{i=1}^{m}(1 - 2^{-i})}{\prod_{i=1}^{v}\prod_{j=1}^{n_i}(1 - 2^{-jd_i})}.$$

We also know that $F_{n_1}(x) = F_{n_2}(x)^{2^i}$ for some $i > 0$ when n_1 and n_2 are in the same 2-cyclotomic coset $CS_{\bar{n}}$ modulo $ord(f) = 2^k - 1$. And the size of set $\mathcal{F} = \{F_{\bar{n}}(x) : 1 < nm < N\}$ is bounded by

$$N/(km) < |\mathcal{F}| \leq \sum_{d|k} \mu(d) \sum_{i=1}^{k/d} 2^i,$$

where $\mu(\cdot)$ is Möbius function.

For the case $\tau \geq 2$, there are $(N/m - 1)|GL_m(\mathbb{F}_2)|(2^{m^2} - 1)$ choices for the two coefficients and n. An upper bound of $|\mathcal{S}|$ is the number of conjugacy classes of T in $GL_{nm}(\mathbb{F}_2)$, which is roughly about $2^{nm} - \sum_{i=\lfloor nm/3 \rfloor}^{\lfloor (nm-1)/2 \rfloor} 2^i$. We believe it is much more than $(2^m - 1)^2$ when $L(x) \in \mathbb{F}_{2^m}$.

The Case of $m = 1$. Regardless of the differences in criterions, the original FCA proposed by Meier et. al. can be treated as our FCA with dimension $m = 1$. The coefficient matrices of LFSR degenerates to scalar elements in $\mathbb{F}_2$. Therefore, the commutative condition for check parity is no need to be considered. The multidimensional linear approximations degenerates to binary linear approximation. Since the multinomial distribution degenerates to binomial distribution, ζ must be 1. The bound derived from Proposition 2 is the same as in [9]. estimating M_1' is also simple.

Small Scale Experiments In this section we perform a scaled experiment to verify the vectorial iterative decoding algorithm. The experiment settings are as follows. The generator polynomial of LFSR is

$$f(x) = x^{16} + x^{15} + x + a \in \mathbb{F}_{2^2}[x].$$

The output of LFSR at time t is the rank cell x_t. The noise stems from a SC channel instead of nonlinear part of a stream cipher. The target is recovering LFSR output sequence $x_1 x_2 \cdots x_N$ from noisy sequence $z_1 z_2 \cdots z_N = (x_1 x_2 \cdots x_N) \oplus (e_1 e_2 \cdots e_N)$.

We tweak the parameters such as channel capacity, the number of parity checks and the noise introduced to verify the word-error ratio(WER) after iterating a number of rounds. More specifically, priori probability distributions are $P_1 = \{0.45, 0.25, 0.2, 0.1\}$ or $P_2 = \{0.33, 0.25, 0.22, 0.20\}$. The key stream length is $N = 2^{19}$ or 2^{21} words. The number of parity checks with $\tau = 2$ are $h = 9$, $h_I = 36$ or $h_{II} = 36$. For example, curve $(1, 1, 2, 9, 19)$ denotes the experiment result derived by parameters P_1, $h = 9$, $N = 2^{19}$ with criterions 1 and 2. Curve $(2, \text{thr}, -, 36II, 19)$ denotes the experiment result derived by parameters P_2, $h_{II} = 36$, $N = 2^{19}$ with threshold criterion. The experiment result is depicted in the following picture.

Some observations could be induced from Figure 3. Firstly, comparing the curve $(1, 1, 2, 9, 19)$ with $(1, 1, 2, 36I, 19)$, we know that the convergence speed

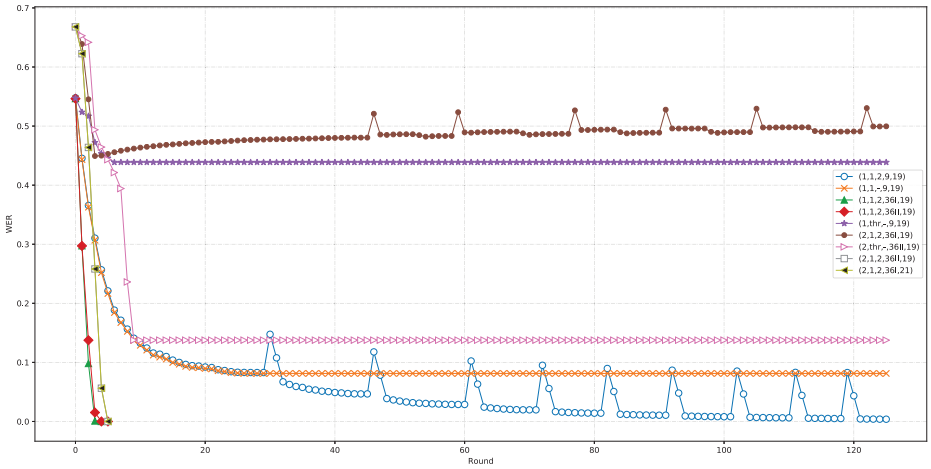

Fig. 3. Several vectorial iterative decoding curves of scaled experiment

increases with the number of parity checks and fixed channel capacity. Secondly, infusing new noise indeed increases the convergence speed. Thirdly, criterion 1 increases the convergence speed. Noticing that curve $(2, 1, 2, 36I, 19)$ seems bad than $(2, 1, 2, 36II, 19)$. The reason is that key stream length $N = 2^{19}$ is not large enough comparing with the degrees. Therefore, the average feasible parity checks for both the head and tail segments of the key stream words in $(2, 1, 2, 36I, 19)$ are less than $(2, 1, 2, 36II, 19)$.

5 Application to Grain-128a

In this section, we apply our new techniques to stream cipher Grain-128a. We assume the cryptanalysis is under the known-plaintext scenario. Since the output is directly used as key stream and the plaintext never participates in updating internal states, this assumption is reasonable for Grain-128a.

5.1 A Brief Description of Grain-128a

Grain-128a includes a 128-bit LFSR cascaded with a 128-bit NFSR. Let $s^{(t)} = (s_t, s_{t+1}, \cdots, s_{t+127})$ and $b^{(t)} = (b_t, b_{t+1}, \cdots, b_{t+127})$ denote their internal states at time t. The output y_t of the pre-output function at time t is represented by

$$y_t = h(s^{(t)}, b^{(t)}) \oplus s_{t+93} \oplus b_{t+2} \oplus b_{t+15} \oplus b_{t+36} \oplus b_{t+45} \oplus b_{t+64} \oplus b_{t+73} \oplus b_{t+89},$$

where $h(s^{(t)}, b^{(t)})$ is defined as

$$\begin{aligned} h(s^{(t)}, b^{(t)}) =& h(b_{t+12}, s_{t+8}, s_{t+13}, s_{t+20}, b_{t+95}, s_{t+42}, s_{t+60}, s_{t+79}, s_{t+94}) \\ =& b_{t+12}s_{t+8} \oplus s_{t+13}s_{t+20} \oplus b_{t+95}s_{t+42} \oplus s_{t+40}s_{t+79} \oplus b_{t+12}b_{t+95}s_{t+94}. \end{aligned}$$

The feedback bits of LFSR and NFSR are computed by

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

Key stream bit $z_t = y_t$ in the stream cipher mode, while $z_t = y_{2w+2t}$ in the authenticated mode, where w is the tag size. The overall structure of Grain-128a is depicted in Fig. 4.

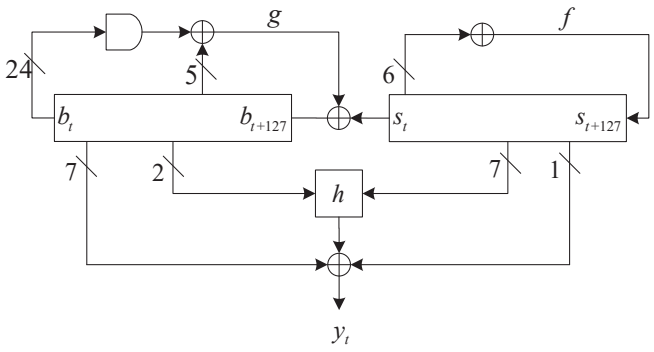

Fig. 4. Overall schematic of Grain-128a

5.2 Constructing Multidimensional Linear Approximations and Checking Parity

In [33], the authors proposed a family of linear approximations of Grain-128a by pilling up different clocks to eliminate the linear terms of the NFSR, which forms are

$$\begin{aligned} \oplus_{i\in\mathbb{T}_z} y_{t+i} \approx & \oplus_{i\in\mathbb{T}_z} s_{t+i+93} \oplus \oplus_{j\in\mathbb{A}} s_{t+j} \oplus_{i\in\mathbb{T}_z} \langle \Lambda_i[1-3], (s_{t+i+8}, s_{t+i+13}, s_{t+i+20})\rangle \\ & \oplus \langle \Lambda_i[5-8], (s_{t+i+42}, s_{t+i+60}, s_{t+i+79}, s_{t+i+94})\rangle, \end{aligned} \tag{22}$$

where $\mathbb{A} = \{2, 15, 36, 45, 64, 73, 89\}, \mathbb{T}_z = \{0, 26, 56, 91, 96, 128\}$, Λ_i is a 9-bit binary linear mask, $\Lambda_i[0, 4]$ is fixed.

According to [33], an assignment of $\Lambda_i[1-3]$ and $\Lambda_i[5-8]$ will completely determine the correlation of h function, when $\Lambda_i[0, 4]$ is fixed. For a specific $i \in \mathbb{T}_z$, there are only 64 possible $\Lambda_i[0, 4], i \in \mathbb{A}$ such that the correlation of

Eq. (22) is nonzero. Hence, the linear correlation value of (22) can be deduced by summing up all these 64 $\Lambda_i[0,4], i \in \mathbb{T}_z$. Meanwhile, there are 2^6 values of $\Lambda_i[1-3,5-8]$ of a specific $i \in \mathbb{T}_z$ with the correlation of h function is nonzero. For example, when $\Lambda_i[1-3,5-8] = 0000000, \forall i \in \mathbb{T}_z$, the correlation of (22) is about $\pm 2^{-57.0454}$. For more details of these linear approximations, we refer to [33].

In this paper, we reuse these linear approximations but in a new way by bundling them up. Firstly, we choose 42 linear approximations which $\Lambda_i[1-3,5-8], i \in \mathbb{T}_z$ has form

$$(\Lambda_0[1-3,5-8], \Lambda_{26}[1-3,5-8], \cdots, \Lambda_{128}[1-3,5-8]) = (0, \cdots, 0, 1, 0, \cdots, 0),$$

i.e., $\Lambda_i[1-3,5-8], i \in \mathbb{T}_z$ as a group of standard basis. Then a linear approximation with dimension $9 \le m \le 42$ can be established as follows.

$$E(\boldsymbol{x}_t + \boldsymbol{u}_t) + E\boldsymbol{y}_t = \boldsymbol{e}_t, \tag{23}$$

where E is an $m \times m$ identity matrix in $\mathbb{F}_2$. $\boldsymbol{e}_t$ is noise vector, and

$$\begin{aligned}
\boldsymbol{x}_t &= (\cdots, s_{t+i+8}, s_{t+i+13}, s_{t+i+20}, s_{t+i+42}, s_{t+i+60}, s_{t+i+79}, s_{t+i+94}, \cdots), \\
\boldsymbol{u}_t &= \left(\sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}, \sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}, \cdots, \sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i} \right), \\
\boldsymbol{y}_t &= \left(\sum_{i \in \mathbb{T}_z} y_{t+i}, \sum_{i \in \mathbb{T}_z} y_{t+i}, \cdots, \sum_{i \in \mathbb{T}_z} y_{t+i} \right), \\
\boldsymbol{e}_t &= (e_t, e_{t+1}, \cdots, e_{t+m-1}).
\end{aligned}$$

Any even Hamming weight linear combination of Eq. (23) will generate a linear approximation without $\sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}$ and $\sum_{i \in \mathbb{T}_z} y_{t+i}$, which correlation would be treated as 0. As for odd linear combinations, it is still required that any of $\Lambda_i[1-3,5-8], i \in \mathbb{T}_z$ will not deduce a zero correlation for h function. Therefore, we can construct a multidimensional linear approximation with dimension $9 \le m \le 42$, which consisting of $2^{m-1-6} = 2^{m-7}$ linear approximations with correlation $\pm 2^{-57.0454}$. By Theorem 1, its SEI $\Delta(e_t) = 2^{m-121.0908}$.

As s_t is a m-sequence, shifting and summation sequence $s'_{t+c'_j} = s_{t+c_j} + \sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}$ is also a a m-sequence with same generator polynomial as s_t. Let vectorial sequence $\boldsymbol{x}'_t = (s'_{t+c'_1}, \cdots, s'_{t+c'_m})$, since shift offsets $c'_j, 1 \le j \le m$ have large difference, the parity checks with $\tau = 1$ are not all ruled out.

Since $\boldsymbol{x}'_t$ runs over $\{1, \cdots, 2^m - 1\}$, there is at most one parity check with $\tau = 1$ for each $0 < n \le N/m$. In order to increase the occurrence possibility for parity check with $t = 1$, several redundant binary linear approximations with nonzero correlation could be added into the subspace. The dimension increases but SEI is almost unchanged. Therefore, the maximal probability value should decrease.

Another way is exploiting a kind of special parity checks with $\tau > 1$. In order to avoid the great loss of SEI while implementing convolution, we play a trade-off trick when special parity checks are feasible.

For example, suppose we have h special parity checks as follows.

$$G_{n,1}\boldsymbol{x}'_{t-d_{n,1}} + \sum_{i=1}^{a} G_{n-i,1}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = 0,$$

$$\cdots,$$

$$G_{n,h}\boldsymbol{x}'_{t-d_{n,h}} + \sum_{i=1}^{a} G_{n-i,h}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = 0.$$

Notice that all of them involve vector variables $\boldsymbol{x}'_t, \boldsymbol{x}'_{t-d_1}, \cdots, \boldsymbol{x}'_{t-d_a}$ except for the last variable $\boldsymbol{x}'_{t-d_{n,j}}$[1]. Let $D_{n-i,j} = G_{n-i,j} + G_{n-i,1}$, $1 \le i \le a$, denote the coefficient difference between the j-th and the 1-st equation. Let $\sum_{i=1}^{a} D_{n-i,j}\boldsymbol{x}'_{t-d_i} = \boldsymbol{\delta}_j$ denote the difference value. Moreover, we require that $\boldsymbol{\delta}_j$ satisfies some restrictions.

Since we have $h-1$ linear equation groups with coefficients $(D_{n-1,j}, \cdots, D_{n-a,j})$, we require that those linear equation groups have the same solution subspace S with large dimension, for example, $am-1$ or $am-2$, which implies that the rank of $(D_{n-1,j}, \cdots, D_{n-a,j})$ may be 1 or 2. Thus when $(\boldsymbol{x}'_t, \boldsymbol{x}'_{t-d_1}, \cdots, \boldsymbol{x}'_{t-d_a}) \in S$, all $\boldsymbol{\delta}_j = \mathbf{0}$. Otherwise, $\boldsymbol{\delta}_j \neq \mathbf{0}$ are likely to different. Thus we have

$$G_{n,1}\boldsymbol{x}'_{t-d_{n,1}} + \mathbf{0} + \sum_{i=1}^{a} G_{n-i,1}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = 0,$$

$$G_{n,1}\boldsymbol{x}'_{t-d_{n,2}} + \boldsymbol{\delta}_2 + \sum_{i=1}^{a} G_{n-i,2}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = 0,$$

$$\cdots,$$

$$G_{n,r}\boldsymbol{x}'_{t-d_{n,h}} + \boldsymbol{\delta}_h + \sum_{i=1}^{a} G_{n-i,h}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = 0.$$

Then the APP for $\sum_{i=1}^{a} G_{n-i,j}\boldsymbol{e}_{t-d_i} + E\boldsymbol{e}_t$ could be evaluated by total probability theorem according to whether all of $\boldsymbol{\delta}_j$ are $\mathbf{0}$. The initial state is recovered from observed values $\boldsymbol{z}_t$ of the error-corrected positions, i.e.

$$\sum_{i=1}^{a} G_{n-i,1}(\boldsymbol{x}'_{t-d_i} + \boldsymbol{e}_{t-d_i}) + E(\boldsymbol{x}'_t + \boldsymbol{e}_t) = \sum_{i=1}^{a} G_{n-i,1}\boldsymbol{x}'_{t-d_i} + E\boldsymbol{x}'_t = \boldsymbol{z}_t.$$

The dimension of linear approximation is not changed but the APP converges slower. Thus the decoding ability decreases when dimension of S decreasing. However, the constraints for parity checks is relaxed.

With these techniques, the fast correlation attack could be performed with these special parit checks and and multidimensional linear approximations in (23).

[1] Some of $G_{n-i,j}$ may be zero matrix.

5.3 Complexity Estimation

In this section, we estimate some theoretical bounds for Grain-128a, which would bring us a new perspective for its security margin.

Let the SEI $\Delta(\boldsymbol{e}_t) = 2^{-\gamma}$, dimension $m = 42$, and $p_0 = 2^{-m} + 2^{-\frac{\gamma+m}{2}}$ be the maximal probability value. To simplify the process of estimating the expected number of positions with $p_\zeta^* > p_0^*$, we need the following hypothesis.

Hypothesis 1 – *The probability distribution P stemming from SEI is close to symmetric distribution.*
– *There are at least 2 parity checks with two taps, or there are more special parity checks as stated in previous section.*

Suppose we have h special parity checks corresponding to a solution subspace of dimension $am - 1$ as stated above. Let $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_h$ denote the check values, $\boldsymbol{\gamma} = (\gamma_0, \cdots, \gamma_{2^m-1})$ and $\boldsymbol{\gamma}' = (\gamma'_0, \cdots, \gamma'_{2^m-1})$ denote the frequency of values in $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_h$ and $\boldsymbol{v}_1, \boldsymbol{v}_2 \oplus \boldsymbol{\delta}_2, \cdots, \boldsymbol{v}_h \oplus \boldsymbol{\delta}_h$ respectively. There exist two events that may deduce $p_\zeta^* > p_0^*$: event A denotes that $\boldsymbol{\gamma} \in \mathcal{A}'_\zeta$, while event B denotes that $\boldsymbol{\gamma}' \in \mathcal{A}'_\zeta$. For simplicity, we only consider that when A occurs, then we have

$$M'_\zeta = 2^{-1} p_\zeta \left(\sum_{\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)} \binom{h}{\boldsymbol{\gamma}} \prod_i p_{i\oplus\zeta}^{\gamma_i} + \sum_{\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)} \binom{h}{\boldsymbol{\gamma}'} \prod_i p_{i\oplus\zeta}^{\gamma'_i} \right),$$

$$M'_0 = 2^{-1} p_0 \left(\sum_{\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)} \binom{h}{\boldsymbol{\gamma}} \prod_i p_i^{\gamma_i} + \sum_{\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)} \binom{h}{\boldsymbol{\gamma}'} \prod_i p_i^{\gamma'_i} \right).$$

The first term denotes the probability that current noise symbol is ζ or 0, when the frequency vector $\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)$ and all $\boldsymbol{\delta}_j = \mathbf{0}$. The second term corresponds to when the frequency vector $\boldsymbol{\gamma} \in \mathcal{A}'(\zeta)$ but many $\boldsymbol{\delta}_j \neq \mathbf{0}$, $2 \leq j \leq h$. Thus the observed vector is γ'. Since $\boldsymbol{\gamma}'_i$ are likely different, It is reasonable to assume that the second terms of M'_ζ and M'_0 are close. To simplify the evaluation, we only consider the first term.

Table 5 in Appendix A depicts the approximation of $M'_\zeta (2^{-1}$ is neglected). M'_1 is estimated by two methods: Skellam distribution and summation for small l. The two estimations are very close to each other. Let $D_i = M'_i - M'_0$ denote the difference. We also compute the summation $\sum_{i=1}^{2^{36}} M'_i$ and the difference summation $\sum_{i=1}^{2^{36}} D'_i$. For example, when $h = h_b = 2$, the expected key stream length $N > 2^{48+42+1} = 2^{91}$. As P is symmetric, it seems no need to evaluate every probability value of APP distribution. Therefore, we use the key stream length N multiplying with the number of parity checks h as time complexity.

For the other case when there are at least 2 parity checks with two taps, there is no probability loss caused by trade-off. The complexity estimation is similar.

6 Further Problems

The analysis of vectorial iterative decoding algorithm is very complicated, there are several problems needed further study.

Firstly, the time complexity is estimated by the key stream length multiplying with the number of parity checks. There are lots of redundant computations. However, we have no idea whether FWHT acceleration technique could be applied in this case. Secondly, we don't know the number of suitable parity checks. Thus the estimation for M_i' and D_i' of Grain-128a is based on a hypothesis. Thirdly, in this paper, we didn't study whether there is also K-tree like method to generate these parity checks in matrix ring. Therefore, the complexity of the precomputation phase is skipped over.

7 Conclusion

In this paper, a vectorial iterative decoding algorithm for FCA is proposed. Two novel criterions are given to break tie and improve the decoding efficiency. The original binary FCA proposed by Meier and Staffelbach is a special case of our FCA with dimension 1. We describe some cryptographic properties about its statistical model, decoding efficiency etc. Based on the statistical property of the first iteration, we estimate the bound of expected key stream length from the perspective of iterative decoding. We also perform a scaled experiment to verify the validity of the vectorial iterative decoding algorithm.

Moreover, we apply it to stream cipher Grain-128a. We construct a multi-dimensional linear approximation with large SEI by bundling up those binary linear approximations proposed in CRYPTO 18. We also give an trade-off approach to use special parity checks with $t > 1$. Consequently, we give an estimation of data complexity for Grain-128a from the point view of vectorial iterative decoding, which is a novel result to evaluate the potential security margin of a real world cipher.

References

1. 29167-13, I.J.I.: Information technology — Automatic identification and data capture techniques — Part 13: Crypto suite Grain-128A security services for air interface communications (2015)
2. Ågren, M., Hell, M., Johansson, T., Meier, W.: Grain-128a: a new version of grain-128 with optional authentication. Int. J. Wirel. Mob. Comput. **5**(1), 48–59 (2011)
3. Ågren, M., Löndahl, C., Hell, M., Johansson, T.: A survey on fast correlation attacks. Cryptogr. Commun. **4**(3-4), 173–202 (2012)
4. Armknecht, F., Mikhalev, V.: On lightweight stream ciphers with shorter internal states. In: Leander, G. (ed.) Fast Software Encryption - 22nd International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers. Lecture Notes in Computer Science, vol. 9054, pp. 451–470. Springer (2015)
5. Aumasson, J., Henzen, L., Meier, W., Naya-Plasencia, M.: Quark: A lightweight hash. J. Cryptol. **26**(2), 313–339 (2013)
6. Baignères, T., Junod, P., Vaudenay, S.: How far can we go beyond linear cryptanalysis? In: Lee, P.J. (ed.) Advances in Cryptology - ASIACRYPT 2004. pp. 432–450. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)

7. Canteaut, A., Trabbia, M.: Improved fast correlation attacks using parity-check equations of weight 4 and 5. In: Preneel, B. (ed.) Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding. Lecture Notes in Computer Science, vol. 1807, pp. 573–588. Springer (2000)
8. Chepyzhov, V.V., Johansson, T., Smeets, B.J.M.: A simple algorithm for fast correlation attacks on stream ciphers. In: Schneier, B. (ed.) Fast Software Encryption, 7th International Workshop, FSE 2000, New York, NY, USA, April 10-12, 2000, Proceedings. Lecture Notes in Computer Science, vol. 1978, pp. 181–195. Springer (2000)
9. Chepyzhov, V.V., Smeets, B.J.M.: On A fast correlation attack on certain stream ciphers. In: Davies, D.W. (ed.) Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings. Lecture Notes in Computer Science, vol. 547, pp. 176–185. Springer (1991)
10. Chose, P., Joux, A., Mitton, M.: Fast correlation attacks: An algorithmic point of view. In: Knudsen, L.R. (ed.) Advances in Cryptology — EUROCRYPT 2002. pp. 209–221. Springer Berlin Heidelberg, Berlin, Heidelberg (2002)
11. Clark, A., Dawson, E., Fuller, J., Golić, J., Lee, H.J., Millan, W., Moon, S.J., Simpson, L.: The lili-ii keystream generator. In: Batten, L., Seberry, J. (eds.) Information Security and Privacy. pp. 25–39. Springer Berlin Heidelberg, Berlin, Heidelberg (2002)
12. Clark, A.J., Golic, J.D., Dawson, E.: A comparison of fast correlation attacks. In: Gollmann, D. (ed.) Fast Software Encryption, Third International Workshop, Cambridge, UK, February 21-23, 1996, Proceedings. Lecture Notes in Computer Science, vol. 1039, pp. 145–157. Springer (1996)
13. Dinur, I., Shamir, A.: Breaking grain-128 with dynamic cube attacks. In: Joux, A. (ed.) Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers. Lecture Notes in Computer Science, vol. 6733, pp. 167–187. Springer (2011)
14. Ekdahl, P., Johansson, T.: Snow-a new stream cipher. In: Proceedings of first open NESSIE workshop, KU-Leuven. pp. 167–168 (2000)
15. Ekdahl, P., Johansson, T.: A new version of the stream cipher snow. In: Nyberg, K., Heys, H. (eds.) Selected Areas in Cryptography. pp. 47–61. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
16. Ekdahl, P., Johansson, T., Maximov, A., Yang, J.: A new SNOW stream cipher called SNOW-V. IACR Trans. Symmetric Cryptol. **2019**(3), 1–42 (2019)
17. Gerstenhaber, M.: On the number of nilpotent matrices with coefficients in a finite field. Illinois Journal of Mathematics **5**(2), 330 – 333 (1961)
18. Golic, J.D., Hawkes, P.: Vectorial approach to fast correlation attacks. Des. Codes Cryptogr. **35**(1), 5–19 (2005)
19. Gong, G., Xiao, G.Z.: Synthesis and uniqueness of m-sequences over gf(q^n) as n-phase sequences over gf(q). IEEE Trans. Commun. **42**(8), 2501–2505 (1994)
20. Hell, M., Johansson, T., Maximov, A., Meier, W.: A stream cipher proposal: Grain-128. In: Proceedings 2006 IEEE International Symposium on Information Theory, ISIT 2006, The Westin Seattle, Seattle, Washington, USA, July 9-14, 2006. pp. 1614–1618. IEEE (2006)
21. Hell, M., Johansson, T., Meier, W.: Grain: a stream cipher for constrained environments. Int. J. Wirel. Mob. Comput. **2**(1), 86–93 (2007)
22. Hermelin, M., Cho, J.Y., Nyberg, K.: Multidimensional linear cryptanalysis. J. Cryptol. **32**(1), 1–34 (2019)

23. Johansson, T., Jönsson, F.: Fast correlation attacks based on turbo code techniques. In: Wiener, M.J. (ed.) Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings. Lecture Notes in Computer Science, vol. 1666, pp. 181–197. Springer (1999)
24. Johansson, T., Jönsson, F.: Improved fast correlation attacks on stream ciphers via convolutional codes. In: Stern, J. (ed.) Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding. Lecture Notes in Computer Science, vol. 1592, pp. 347–362. Springer (1999)
25. Johansson, T., Jönsson, F.: Fast correlation attacks through reconstruction of linear polynomials. In: Bellare, M. (ed.) Advances in Cryptology — CRYPTO 2000. pp. 300–315. Springer Berlin Heidelberg, Berlin, Heidelberg (2000)
26. Lu, Y., Vaudenay, S.: Faster correlation attack on bluetooth keystream generator E0. In: Franklin, M.K. (ed.) Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3152, pp. 407–425. Springer (2004)
27. Meier, W., Staffelbach, O.: Fast correlation attacks on certain stream ciphers. J. Cryptol. **1**(3), 159–176 (1989)
28. Mihaljevi, M.J., Fossorier, M.P.C., Imai, H.: Fast correlation attack algorithm with list decoding and an application. In: Matsui, M. (ed.) Fast Software Encryption. pp. 196–210. Springer Berlin Heidelberg, Berlin, Heidelberg (2002)
29. Mihaljevic, M.J., Golic, J.D.: A comparison of cryptanalytic principles based on iterative error-correction. In: Davies, D.W. (ed.) Advances in Cryptology - EUROCRYPT '91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings. Lecture Notes in Computer Science, vol. 547, pp. 527–531. Springer (1991)
30. Mihaljević, M.J., Golić, J.D.: Convergence of a bayesian iterative error-correction procedure on a noisy shift register sequence. In: Rueppel, R.A. (ed.) Advances in Cryptology — EUROCRYPT' 92. pp. 124–137. Springer Berlin Heidelberg, Berlin, Heidelberg (1993)
31. Mikhalev, V., Armknecht, F., Müller, C.: On ciphers that continuously access the non-volatile key. IACR Trans. Symmetric Cryptol. **2016**(2), 52–79 (2016)
32. Nikolić, I., Sasaki, Y.: Refinements of the k-tree algorithm for the generalized birthday problem. In: Iwata, T., Cheon, J.H. (eds.) Advances in Cryptology – ASIACRYPT 2015. pp. 683–703. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)
33. Todo, Y., Isobe, T., Meier, W., Aoki, K., Zhang, B.: Fast correlation attack revisited. In: Shacham, H., Boldyreva, A. (eds.) Advances in Cryptology – CRYPTO 2018. pp. 129–159. Springer International Publishing, Cham (2018)
34. UEA2&UIA, I.: Specification of the 3gpp confidentiality and integrity algorithms uea2& uia2. document 2: Snow 3g specifications. version: 1.1. etsi (2006)
35. Yang, J., Johansson, T., Maximov, A.: Spectral analysis of ZUC-256. IACR Trans. Symmetric Cryptol. **2020**(1), 266–288 (2020)
36. Zeng, K., Yang, C.H., Rao, T.R.N.: An improved linear syndrome algorithm in cryptanalysis with applications. In: Menezes, A.J., Vanstone, S.A. (eds.) Advances in Cryptology-CRYPTO' 90. pp. 34–47. Springer Berlin Heidelberg, Berlin, Heidelberg (1991)

37. Zhang, B., Li, Z., Feng, D., Lin, D.: Near collision attack on the grain v1 stream cipher. In: Moriai, S. (ed.) Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers. Lecture Notes in Computer Science, vol. 8424, pp. 518–538. Springer (2013)
38. Zhang, B., Xu, C., Feng, D.: Practical cryptanalysis of bluetooth encryption with condition masking. J. Cryptol. **31**(2), 394–433 (2018)
39. Zhang, B., Xu, C., Meier, W.: Fast correlation attacks over extension fields, large-unit linear approximation and cryptanalysis of snow 2.0. In: Gennaro, R., Robshaw, M. (eds.) Advances in Cryptology – CRYPTO 2015. pp. 643–662. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)

A Estimation of M_i'

Table 5. Estimation of some M_i' with $m = 42$

$\log_2(h)$	$\log_2(D_1)$	$\log_2(M_1')$		$\log_2(\sum_{i=1}^{2^{36}} M_i')$	$\log_2(\sum_{i=1}^{2^{36}} D_i')$
		summation	Skellam		
1	-101.5454	-84.0004	-83.0000	-47.9999	-65.5417
2	-98.9604	-81.4150	-81.0000	-45.4151	-62.9717
3	-96.7380	-79.1926	-79.0000	-43.1943	-60.7722
4	-94.6385	-77.0931	-77.0000	-41.1209	-58.7914
5	-92.5912	-75.0458	-75.0000	-39.0876	-56.7683
6	-90.5681	-73.0227	-73.0000	-37.1719	-54.9443
7	-88.5567	-71.0113	-71.0000	-35.4574	-53.3305
8	-86.5510	-69.0056	-69.0000	-34.0809	-52.0229
9	-84.5482	-67.0028	-67.0000	-33.0023	-50.9621
10	-82.5468	-65.0014	-65.0000	-32.0000	-49.9604
11	-80.5461	-63.0007	-63.0000	-31.0000	-48.9604
12	-78.5458	-61.0003	-61.0000	-30.0000	-47.9604
13	-76.5456	-59.0002	-59.0000	-29.0000	-46.9604
14	-74.5455	-57.0001	-57.0000	-28.0000	-45.9604