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

Zhaocun Zhou, Dengguo Feng and Bin Zhang

Trusted Computing and Information Assurance Laboratory, Institute of Software
Chinese Academy of Sciences

March 18, 2023

Overview

1.1 Backgrouds

- Linear feedback shift register (LFSR) based stream ciphers form an important class of stream cipher system: LILI-128 [CDF02], the SNOW family [EJ00] and the Grain family [AHJM11], etc
- Cryptanalysis based on correlation plays an important role in their evaluations, e.g., (fast) correlation attacks (FCA), linear distinguishing attacks (LDA), etc
- According to decoding strategies, FCA can be divided into two classes
  - One-pass: information set decoding [TIM18], convolution codes [JJ99], etc
  - Probabilistic iterative: Algorithm B [MS89], LDPC codes [CT00], etc
- Applications of iterative decoding are limited as
  - Its properties are hard to describe by mathematical language
  - Lacks of a convenient iterative decoding algorithm to work with the multidimensional linear approximation

1.2 The binary iterative decoding algorithm [MS89]

- A binary iterative decoding algorithm to improve the time complexity of FCA that thought to be exponential to the length of the LFSR [MS89]

Algorithm 1 Meier and Staffelbach's binary iterative decoding 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 $r \in \{1, 2, \ldots\}$ **do**
3.   **For** iteration i from 1 to a small integer **do**
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 $N_w = |\{n | p_n > p_{thr}\}|$ **then**, break; **EndIf**
6.   **EndFor**
7.   Complement the bits of $\boldsymbol{z}$ with $p_n > p_{thr}$.
8.   Reset all positions to initial probability p.
9.   **If** $\boldsymbol{z}$ satisfies all parity-checks **then**, break; **EndIf**
10. **EndFor**
11. Terminate with $\boldsymbol{x} = \boldsymbol{z}$.

1.2 The binary iterative decoding algorithm [MS89]

- The critical part of the decoding phase is calculating a posterior probability (APP) p^* from prior distribution p symbol by symbol through Bayes' formula, instead of directly determine $0/1$

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

$$s(p_{l_1}, \ldots, p_{l_\tau}) = p_{l_\tau} s(p_{l_1}, \ldots, p_{l_{\tau-1}}) + (1 - p_{l_\tau})(1 - s(p_{l_1}, \ldots, p_{l_{\tau-1}}))$$

- The more parity-checks holds (check value $= 0$), the lower value p^* (suppose that $p < 1 - p$)
- When the number of positions with large p^* is greater than a threshold value, perform a complement

1.3 Our Work

- We propose a vectorial iterative decoding algorithm for FCA that
  - Generalizes the binary algorithm in [MS89] naturally
  - May benefit from a multidimensional linear approximation
  - Equips with two novel criteria to improve the iterative decoding process
- We present some cryptographic properties on the vectorial algorithm such as
  - the relationship between the decoding efficiency and the noise distribution by analyzing the first iteration
  - two propositions involving the relationship between the number of parity-checks, the noise distribution and the data complexity
- We apply those results to stream cipher Grain-128a and show its security margin from the perspective of vectorial iterative decoding

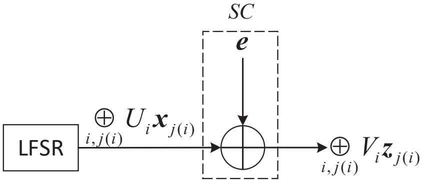
2.1 Channel model: from BSC to SC

- Suppose a linear approximation with dimension m

$$\bigoplus_{\substack{i\in\{1,\ldots,\#\mathcal{T}_x\}\\ j(i)\in\mathcal{T}_x}} U_i \boldsymbol{x}_{j(i)} \oplus \bigoplus_{\substack{i\in\{1,\ldots,\#\mathcal{T}_z\}\\ j(i)\in\mathcal{T}_z}} V_i \boldsymbol{z}_{j(i)} = \boldsymbol{e}.$$

where all U_i and V_i are $m \times w$ matrices over $\mathbb{F}_2$, $\mathcal{T}_x$ and $\mathcal{T}_z$ are sets of indexes related to the linear approximation

- Similarly as BSC, the channel noise vector $\boldsymbol{e}$ is XORed to the code word



2.2 Checking parity with vectorial noises

- Suppose a parity-check over the matrix ring $M_w(\mathbb{F}_2)$

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

- Require that for each G_k, there is a $m \times m$ matrix G'_k satisfies that $U_iG_k = G'_kU_i, \forall i \in \{1, \ldots, \#\mathcal{T}_x\}$. Multiplying with these U_is

$$\bigoplus_{i=0}^{d} G'_i \left(\bigoplus_{j=1}^{\#\mathcal{T}_x} U_j\boldsymbol{x}_{n-i+k(j)} \right) = \bigoplus_{i=0}^{d} G'_i \left(\bigoplus_{j=1}^{\#\mathcal{T}_z} V_j\boldsymbol{z}_{n-i+k'(j)} \right) \oplus \bigoplus_{i=0}^{d} G'_i\boldsymbol{e}_{n-i}$$

- The target is to determine $\boldsymbol{e}_{n-i}$ of each position when observing $\bigoplus_{j=1}^{\#\mathcal{T}_z} V_j\boldsymbol{z}_{n-i+k'(j)}$, which can be accomplished by a vectorial iterative decoding algorithm

2.3 Vectorial iterative algorithm

- Similarly as the binary case, calculate APP from the priori distribution according to check values by Bayes' formula (suppose $\boldsymbol{e}_{n-l_i}$ are independent and all parity-checks are orthogonal)

$$
\begin{aligned}
p_{\zeta}^{*(n)} &= \Pr\left[\boldsymbol{e}_n = \zeta \middle| \text{when observed check values } (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_h)\right] \\
&= \frac{p_{\zeta}^{(n)} \prod_{l \in \mathcal{H}^{(n)}} \Pr[\bigoplus_{i=1}^{\tau} G'_{l_i} \boldsymbol{e}_{n-l_i} = \boldsymbol{c}_l \oplus E\zeta]}{\bigoplus_{\eta} p_{\eta}^{(n)} \prod_{l \in \mathcal{H}^{(n)}} \Pr[\bigoplus_{i=1}^{\tau} G'_{l_i} \boldsymbol{e}_{n-l_i} = \boldsymbol{c}_l \oplus E\eta]}
\end{aligned}
$$

- For each symbol, we compute APP and increase an empirical vector $\boldsymbol{E}^{itr}$. If $\boldsymbol{E}^{itr}$ is still increasing, then we assign PRI with APP, and continue iterating

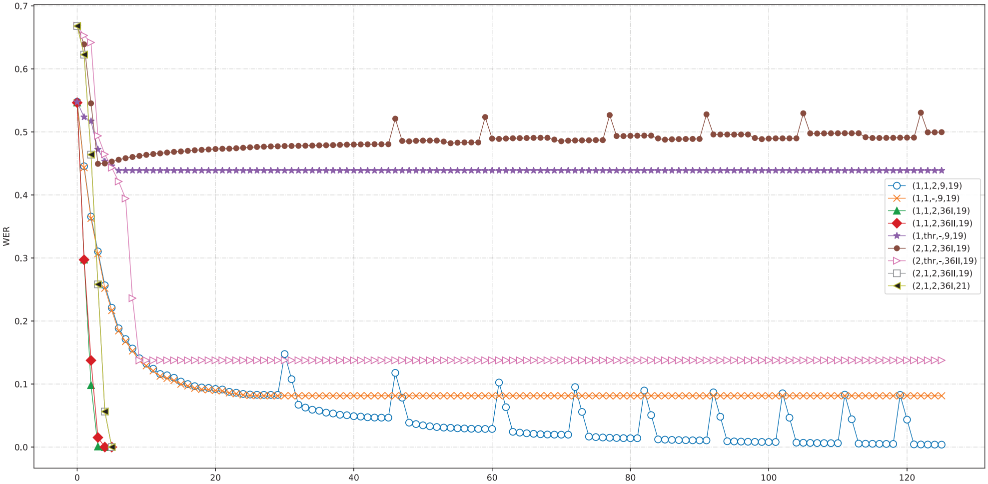
2.3 Vectorial iterative algorithm

Input: The sequence $\boldsymbol{z}'$ of length N derived from key stream,
The sequence of noises $\boldsymbol{e}$ with initial p.d. $\boldsymbol{p}$,
The parity-checks set $\mathcal{H}$ with $\tau + 1$ taps.
parameters: Maximal rounds R, maximal iterations T and minimal gap G to infuse new noises.

$\boldsymbol{pri} \leftarrow \boldsymbol{p}$, $\boldsymbol{E}^{glb} = (E_1^{glb}, \dots, E_{2^m-1}^{glb}) \leftarrow \boldsymbol{0}$.
For $r = 1, 2, \dots, R$ **do**
  $\boldsymbol{E}^{rnd} = (E_1^{rnd}, \dots, E_{2^m-1}^{rnd}) \leftarrow \boldsymbol{0}$, $\zeta \leftarrow \boldsymbol{0}$.
  For $i = 1, 2, \dots, T$ **do**
    $\boldsymbol{E}^{itr} = (E_1^{iter}, \dots, E_{2^m-1}^{iter}) \leftarrow \boldsymbol{0}$.
    For $n = 1, 2, \dots, N$ **do**
      Compute $\boldsymbol{app}$ from $\boldsymbol{pri}$ by equation (6).
      If $p_j^{(n)} > p_0^{(n)}$ **then** $E_j^{itr} \quad E_j^{itr} + 1/N, j \in \{1, 2, \dots, 2^m - 1\}$. **End If**.
    End For.
    If $\boldsymbol{E}^{itr} \succ \boldsymbol{E}^{rnd}$ **then** $\boldsymbol{E}^{rnd} \leftarrow \boldsymbol{E}^{itr}$, $\boldsymbol{pri} \leftarrow \boldsymbol{app}$. **End If**.
    If $\boldsymbol{E}^{itr} \preceq \boldsymbol{E}^{rnd}$ **or** $i = T$ **then**
      If $\boldsymbol{E}^{itr} = \boldsymbol{0}$ **then** return failed.
      else if $||\boldsymbol{E}^{rnd} - \boldsymbol{E}^{glb}|| < G$ **then** reset $\boldsymbol{z}' \leftarrow \boldsymbol{z}' \oplus \boldsymbol{n}$, break.
      else $\boldsymbol{E}^{glb} \leftarrow \boldsymbol{E}^{rnd}$, select ζ that maximizes $E_{int(\zeta)}^{rnd} + E_{int(\zeta)}^{itr}$, break. **End If**.
    End If.
  End For.
  If $\zeta \neq \boldsymbol{0}$ **then** complement all positions of $\boldsymbol{z}'$ such that $p_\zeta > p_{\boldsymbol{0}}$ with ζ. **End If**.
  If $\boldsymbol{z}'$ satisfies all parity-checks **then** return success. **End If**.
  Reset $\boldsymbol{pri} \leftarrow \boldsymbol{p}$.
End For.
Terminate.

2.4 Scaled experiments for the vectorial algorithm

Choose LFSR to be $x^{16} + x^{15} + x + \alpha \in \mathbb{F}_{2^2}[x]$. Tweak channel capacity, the number of parity-checks and the infused noises to verify the word-error ratio (WER).

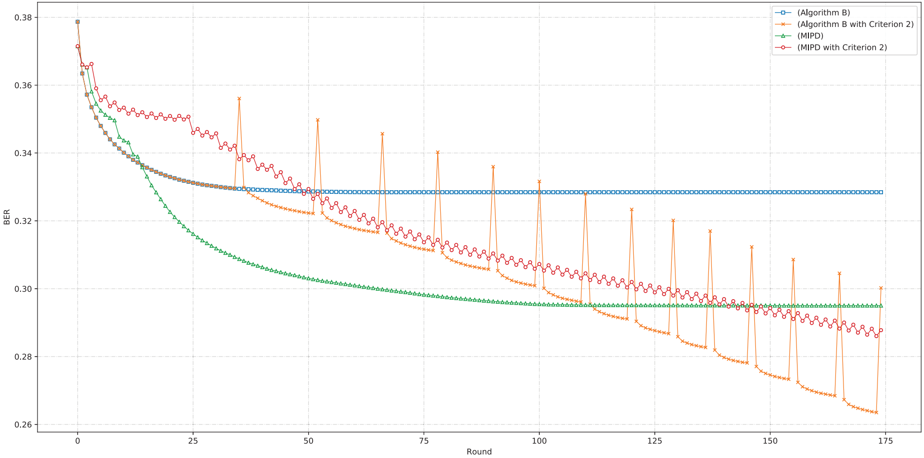


2.5 Two novel iterative criteria

- Criterion 1. Passing through sufficient iterations before breaking up and resetting
  - If new APP strengthens the complement effect, continue iterating
  - Otherwise, select the complement coin with the potential largest complement effect
- Criterion 2. When the empirical complement effect is weak, a sequence of very biased noises is infused in order to break the tie
  - The noises' SEI is required to be appropriate, neither very large to counteract the previous decoding work nor very small to break the tie
  - May help to improve some other binary algorithms, e.g., Algorithm B [MS89], MIPD [CGD96]

2.6 Scaled experiments for Criterion 2

- Algorithm B [MS89], MIPD algorithm [CGD96] versus their modified versions by Criterion 2



1. Introduction
2. Vectorial iterative algorithm and FCA
3. **Some properties of the vectorial iterative algorithm**
4. Applications to Grain-128a
5. Summary

3.1 Statistical properties of the first iteration

(1) Convergence property

- Suppose decoding is feasible, it is expected that APP $p_{\zeta}^{*(n)}$ increases when noise variable $\boldsymbol{e}_n = \zeta$ and decreases when $\boldsymbol{e}_n \neq \zeta$. Similarly as the binary case, we have $E[p^{*(n)}] = p_{\zeta} E[p_{\zeta}^{*(n)} | \boldsymbol{e}_n = \zeta] + (1 - p_{\zeta}) E[p_{\zeta}^{*(n)} | \boldsymbol{e}_n \neq \zeta] = p_{\zeta}$

Examples (1)

Let LFSR be the same as the previous, and the number of parity-checks $h = 3$ with $\tau = 3$ taps.

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

3.1 Statistical properties of the first iteration

(2) Estimating decoding efficiency

- In binary case [MS89], a threshold N_{thr} is introduced to measure the decoding efficiency, which is determined by the intersection point of two shrunk normal distributions
- In vectorial case, the intersection point becomes an intersection curve (surface)
- Our idea is classification and approximation
    - Classification: the parity-checks are divided into two classes, i.e., those whose coefficients are all identity matrices (the set $\mathcal{H}_I$) and the others (the set $\mathcal{H}_{II}$)
    - Approximation: multinomial distribution is approximated by multivariate normal distribution

3.1 Statistical properties of the first iteration

- Suppose $p_0 \geq p_1 \geq \ldots \geq p_{2^m-1} > 0$. Let $q_{\boldsymbol{c}}$ denote the probability that the τ taps sum to be $\boldsymbol{c}$
- The probability that noise $\boldsymbol{e} = \zeta$ and x_i check values equal i follows multinomial distribution

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

- For $\mathcal{H}_I$, using distribution p_i and q_i. For $\mathcal{H}_{II}$, using distribution p_i and symmetric distribution q'_i

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

3.1 Statistical properties of the first iteration

Example (2)

Let parameters be the same as the previous. Calculate the theoretical and approximate value of N_{ζ}^{thr}/N via classifying parity-checks.

No. of parity-checks (h_I, h_{II})	ζ	theoretical	approximate $N = 2^{19}$	approximate $N = 2^{20}$	approximate $N = 2^{21}$
(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

3.1 Statistical properties of the first iteration

Approximating the threshold by multivariate normal distribution

When multivariate normal approximation is feasible, the threshold can also be

$$N \sum_{\zeta \in \mathbb{F}_2^m} \int_{\mathcal{A}(\zeta)} \mathcal{N}(\boldsymbol{\mu}_\zeta, \boldsymbol{\Sigma}_\zeta) d\boldsymbol{x}.$$

where $\mathcal{A}(\zeta)$ is part of a hypercube restricted by $2^m - 1$ coordinate planes and two surfaces

$$\sum_{i}^{2^m-2} x_i = h_I, \frac{1}{2}\left((\boldsymbol{x} - \boldsymbol{\mu}_0)^T \boldsymbol{\Sigma}_0^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_0)\right) - \frac{1}{2}\Big((\boldsymbol{x} - \boldsymbol{\mu}_\zeta)^T \boldsymbol{\Sigma}_\zeta^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_\zeta)\Big) - \ln \frac{p_0}{p_\zeta} = 0,$$

and maximizes the multiple integral

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

3.1 Statistical properties of the first iteration

Example (3)

Let parameters be the same as the previous. In order to simplify the integral, we could even slightly adequate the boundary of $\mathcal{A}$ without much fluctuation.

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

h_I	40	80	200	400
direct computation	0.0686	0.1138	0.1835	0.2266
normal approximation	0.0707	0.1148	0.1841	0.2267

3.2 Two bounds related to complexities

(1) An iterative bound

- In order to perform iterative decoding, the lower bound of h should satisfy that there exists at least a ζ such that $p_\zeta^* > p_0^*$

Proposition 1

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

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

3.2 Two bounds related to complexities

- Potential advantages of vectorial iterative decoding

Examples (4)

When SEI $\Delta(\boldsymbol{e}) = 2^{-\gamma}$, it is expected that there are probability values around $2^{-m} \pm 2^{-\frac{2m+\gamma}{2}}$ in practice [YJM20]. According to Prop. 1, we need at least $2^{\gamma/2}(2^m - 1)$ parity-checks with 3 taps. Thus the length N of data needed satisfies $(2^m - 1)^2 2^{-l}\binom{N}{2} \approx 2^{\gamma/2}(2^m - 1)$ by a birthday collision, which means $N \approx 2^{(\gamma+2l+2)/4}/\sqrt{2^m - 1}$. While $m = 1$, $N \approx 2^{(\gamma+2l+2)/4}$. For the vectorial case, N seems to be smaller than the binary case, because that $m > 1$ and γ is expected to be smaller than the binary case.

3.2 Two bounds related to complexities

(2) A bound related to the expected number of corrected errors

- Let $\mathcal{A}'(i) = \mathcal{A}(i) - \mathcal{A}(i) \cap (\bigcup_{j=1}^{i-1} \mathcal{A}(i))$, $M'_\zeta = p_\zeta \sum_{x \in \mathcal{A}'(\zeta)} q(\boldsymbol{x}, \zeta)$. 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 $p^*_\zeta > p^*_0$
- Summing the probability values in multinomial distributions is inconvenient. Meanwhile, since the integral area $\mathcal{A}'(\zeta)$ is very complicated, multivariate normal approximation is not practical when h is large
- However, since $\boldsymbol{q}'$ simulates the iterative process very well, we could deduce a bound using multinomial distribution $\text{Multi}(h, \boldsymbol{q}')$

3.2 Two bounds related to complexities

Proposition 2

For multinomial distribution Multi$(h, \boldsymbol{q}')$, we have

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

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 those small l.

- When $\zeta = 1$, M'_1 can be estimated by Skellam distribution

1. Introduction

2. Vectorial iterative algorithm and FCA

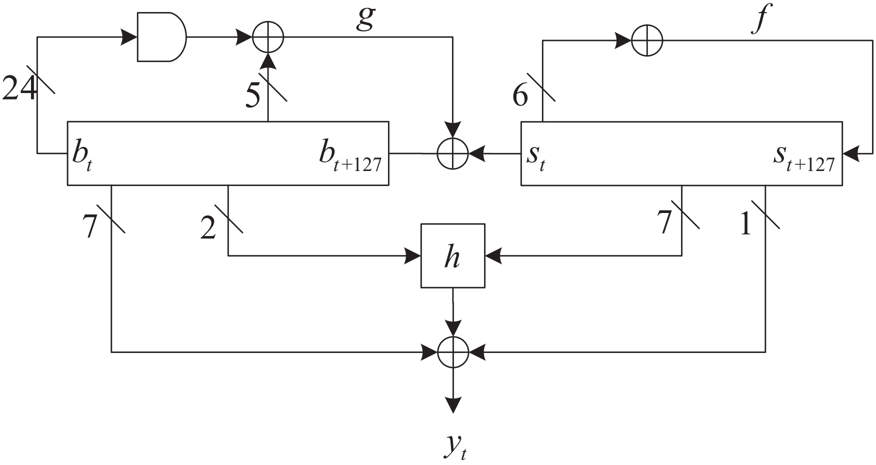
3. Some properties of the vectorial iterative algorithm

4. Applications to Grain-128a

5. Summary

4.1 Grain-128a

Grain-128a includes a 128-bit LFSR cascaded with a 128-bit NFSR.



4.2 Constructing a multidimensional linear approximation

- There are binary linear approximations with correlation $\pm 2^{-57.0454}$ [TIM18]
- Bundling up them will derive a linear approximation with dimension $9 < m \leq 42$, SEI $2^{m-121.0908}$, and the form

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

$$\boldsymbol{x}_t = (\ldots, s_{t+i+8}, s_{t+i+13}, s_{t+i+20}, s_{t+i+42}, s_{t+i+60}, s_{t+i+79}, s_{t+i+94}, \ldots),$$

$$\boldsymbol{u}_t = \left(\sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}, \sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i}, \ldots, \sum_{i \in \mathbb{A} \bigcup \mathbb{T}_z} s_{t+i} \right),$$

$$\boldsymbol{y}_t = \left(\sum_{i \in \mathbb{T}_z} y_{t+i}, \sum_{i \in \mathbb{T}_z} y_{t+i}, \ldots, \sum_{i \in \mathbb{T}_z} y_{t+i} \right), \boldsymbol{e}_t = (e_t, e_{t+1}, \ldots, e_{t+m-1}).$$

- When $m = 42$, the standard basis of linear masks is

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

4.3 Estimating the data complexity

- Suppose the SEI is $2^{-\gamma}$, $p_0 = 2^{-m} + 2^{-\frac{2m+\gamma}{2}}$ is maximal probability point
- Hypothesis: suppose there are at least 2 parity-checks with two taps, or there are more special parity-checks with form

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

- According the two bounds when $m = 42$
  - E.g., $h = 2$, the 1-st bound: $N > 2^{48+42+1} = 2^{91}$, and the 2-nd bound: $N > 2^{86.54+42+1} = 2^{129.54}$

$\log_2(h)$	$\log_2(D_1)$	$\log_2(M'_1)$		$\log_2(\sum_{i=1}^{2^{36}} M'_i)$	$\log_2(\sum_{i=1}^{2^{36}} D'_i)$
		summation	Skellam		
1	-122.5454	-84.0004	-83.0000	-47.9999	-86.5435
2	-119.9605	-81.4150	-81.0000	-45.4151	-83.9722
3	-117.7381	-79.1926	-79.0000	-43.1943	-81.7714
4	-115.6385	-77.0931	-77.0000	-41.1209	-79.7206

Discussion and open problems

- We cannot directly compare the vectorial decoding algorithm with a binary algorithm, and theoretical advantage in the general case is an open problem
- The other theoretical properties of the vectorial algorithm are still not clear
- the main difficulties are figuring out the existence of the special parity-checks and proposing an efficient algorithm to generate suitable parity-checks in matrix rings instead of finite fields

Concluding remarks

- We propose a vectorial iterative decoding algorithm for FCA. The original binary FCA [MS89] is a special case of our FCA with dimension 1
- We describe some cryptographic properties and estimate the quantity of needed parity-checks and keystream
- We apply it to stream cipher Grain-128a and estimate its potential security margin from the point view of vectorial probabilistic iterative decoding

Thank you for your attention!