

Peeling Nonlinear Layers: Algebraic Cryptanalysis of Full-Round **Iasta**

Chandan Dey^{*} Abul Kalam[†] Santanu Sarkar[‡]

Abstract

Iasta is a stream cipher designed for hybrid homomorphic encryption (HHE), with claimed 128-bit security for its **Iasta-3** and **Iasta-4** instances. In this work, we present the first third-party cryptanalysis of the full-round **Iasta-3** and **Iasta-4** instances and further extend our approach to **Iasta-5**. Our cryptanalysis exploits the restricted randomness and structured construction of the nonce-dependent affine-layer matrices. We show that the matrix space contains only $2^{31.30}$ and $2^{19.62}$ distinct matrices for **Iasta-3** and **Iasta-4/5**, respectively, compared with the 2^{29} and 2^{22} matrix randomness claimed by the designers. This restricted matrix space enables us to construct *weak nonces* that induce an identical matrix in the final affine layer. For such nonces, we peel off the final Cube transformation, yielding polynomial equations of degree at most 2^{d-1} in the secret-key coefficients, which we solve using linearization. We further consider the more restrictive class of nonces that induce identical matrices in both the initial and final affine layers. This allows us to additionally peel off the first non-linear layer, reducing the degree of the resulting equations to at most 2^{d-2} at the cost of introducing additional linearization variables. This extended attack substantially improves the attack complexity for **Iasta-4** and **Iasta-5**.

For **Iasta-3** and **Iasta-4**, our best estimated attack complexities are 2^{59} and 2^{67} operations, respectively, under $\omega = 2$, reducing the claimed 128-bit security level to an almost square-root security level. Even under the conservative setting $\omega = 3$, the attack requires approximately 2^{80} and 2^{82} operations against **Iasta-3** and **Iasta-4**, respectively, both below 2^{128} . For **Iasta-5**, the extended attack achieves an estimated complexity of 2^{99} operations under $\omega = 2$. Although no overall security level is explicitly specified for **Iasta-5**, this result demonstrates that our attack can also reach a complexity below 2^{128} for this instance. In all three instances, our attacks reveal a structural weakness in **Iasta** arising from the restricted space of nonce-dependent affine-layer matrices.

Keywords: HE-friendly cipher, stream cipher, **Iasta**, linearization attack, weak nonce

^{*}Indian Statistical Institute, Kolkata, India. Email: c.dey.math95@gmail.com

[†]Indian Institute of Technology Madras, India. Email: abulkalam.sunny@gmail.com

[‡]Indian Institute of Technology Madras, India. Email: sarkar.santanu.bir1@gmail.com

1 Introduction

Hybrid homomorphic encryption (HHE) has become a practical way to address the large ciphertext size problem of fully homomorphic encryption (FHE), especially in bandwidth- and storage-limited applications such as cloud-assisted Internet of Things (IoT) systems. In a typical HHE system, an edge device first encrypts its data using a lightweight symmetric-key cipher. Then sends the compact symmetric ciphertext along with an FHE-encrypted version of the symmetric key to the cloud server. The server uses FHE to evaluate the decryption circuit of the symmetric cipher and converts the data to FHE ciphertexts. These ciphertexts can then be further processed using homomorphic operations.

Since symmetric decryption is performed within the FHE framework, the number and depth of its nonlinear operations, particularly AND or multiplication operations, have a direct impact on the computational cost and noise growth at the server. This has led to the development of efficient HE-friendly or FHE-friendly ciphers. Early examples include LowMC [ARS⁺15], followed by the Rasta family (Rasta [DEG⁺18], Dasta [HL20], Fasta [CIR22], Masta [HKC⁺20], Pasta [DGH⁺23], Pastav2 [GLR⁺25]), as well as HERA [CHK⁺21], Rubato [HKL⁺22], Chaghri [AMT22], Elisabeth-4 [CHMS22], FRAST [CCH⁺24], and Transistor [BBB⁺25]. These ciphers are designed to reduce the number and depth of nonlinear operations. However, this efficiency often comes with trade-offs, such as unusual algebraic structures, fewer rounds, and, in some cases, reduced randomness in their linear layers. As a result, recent cryptanalytic studies have demonstrated that such design choices can introduce subtle vulnerabilities [DLR16, RST18, ACG⁺19, LIM21, Din21, LSMI21, LSW⁺22, AMT22, GBJR23, LKSM24, LWN⁺25, GAH⁺23].

Iasta [YCY⁺26] is a recent design in this family. It is an $\mathbb{F}_p$ -based stream cipher designed for BGV/BFV-style Hybrid Homomorphic Encryption (HHE) in cloud-assisted IoT applications. Similarly to Rasta, **Iasta** follows an ASASA-like structure in which affine and nonlinear layers are applied alternately. However, **Iasta** differs from previous designs based on $\mathbb{F}_p$, such as Masta [HKC⁺20] and Pasta [DGH⁺23], in two main aspects. First, **Iasta** uses a different nonlinear layer, and second, it uses a highly structured affine layer instead of a fully random matrix.

Instead of using the traditional chi transformation, **Iasta** combines two types of S-boxes. The first is a Feistel-style S-box applied to most of the state components, while the second is a higher-degree Cube S-box applied only in the final round. According to the designers, this approach is intended to prevent the final-round chi-stripping attack proposed by Liu et al. [LSMI21]. It also increases the algebraic degree of the cipher and allows **Iasta** to use fewer rounds than Rasta, Fasta, and Pasta while targeting a similar security level.

Iasta constructs its affine matrices from a small number of structured, rotation-based components. These components are generated using a specially designed sparse polynomial together with rotation matrices controlled by only a few shift parameters. As a result, the amount of randomness required to generate the affine layer is greatly reduced. For the **Iasta-4** parameter set targeting

128-bit security, only about 22 bits of XOF output are required for the affine layer of each round. This is significantly smaller than the corresponding randomness requirement of *Pasta*. The designers argue that this reduction does not significantly affect security because the structured matrices still provide sufficient diffusion. These structures motivate a more detailed algebraic analysis of *Iasta*.

Linearization Technique. The linearization method is one of the simplest approaches for solving an overdetermined system of polynomial equations. Suppose that, over a field $\mathbb{F}$, we have m equations of degree at most d in n variables. The number of non-constant monomials of degree at most d is

$$N_M = \binom{n+d}{n} - 1.$$

The linearization technique treats each monomial as an independent variable, thereby transforming the original system into a linear system of m equations in N_M variables. If $m \geq N_M$ and the resulting $m \times N_M$ coefficient matrix has full column rank, the linearized system can be solved by Gaussian elimination. In the usual case, we choose $m = N_M$, obtaining a square linear system. The resulting linear system can then be solved in

$$\mathcal{O}(N_M^\omega) = \mathcal{O}\left(\binom{n+d}{n}^\omega\right),$$

field operations, where ω denotes the algebraic constant, with $2 \leq \omega \leq 3$.

The Algebraic Constant (ω). Matrix multiplication, matrix inversion, and Gaussian elimination have asymptotically equivalent arithmetic complexity [CLRS09, Str69]. The complexity of Gaussian elimination on an $n \times n$ matrix can be expressed as $\mathcal{O}(n^\omega)$ field operations, where $2 \leq \omega \leq 3$. The conventional Gaussian elimination algorithm corresponds to $\omega = 3$. Faster matrix multiplication techniques, such as Strassen’s algorithm [Str69], give $\omega \approx 2.8$. However, security evaluations based on linearization attacks often adopt [DEG⁺18, HKC⁺20, HL20, CHK⁺21, CIR22, HKL⁺22, YCY⁺26] the optimistic value $\omega = 2$ when determining the required number of rounds. To account for these different assumptions, we evaluate our attack for $\omega \in \{2, 2.8, 3\}$.

1.1 Our Contribution

We present the first third-party cryptanalysis of the full-round *Iasta*-3 and *Iasta*-4 instances, and further extend our approach to *Iasta*-5. Our cryptanalysis is based on a structural property of the nonce-dependent affine layers of *Iasta*. Although the matrices in these layers appear to be generated from a large random space, their actual number of possible values is significantly restricted. This restriction allows us to identify classes of nonces that induce the same affine-layer matrices. By fixing the matrix in the final affine layer, we can peel off the final non-linear layer. We further consider the more restrictive

class of nonces for which both the initial and final affine-layer matrices are fixed, allowing us to peel off the first non-linear layer as well.

The attack naturally divides into an offline phase and an online phase. The offline phase identifies a sufficiently large collection of such weak nonces, while the online phase uses the corresponding keystreams to construct and solve the resulting low-degree polynomial systems.

Offline Phase. The offline phase exploits the limited randomness of the affine-layer matrices. In particular, we show that there are $2^{31.30}$ possible matrices for **Iasta-3** and $2^{19.62}$ possible matrices for **Iasta-4/5**, whereas [YCY⁺26] claims 29 bits and 22 bits of matrix randomness, respectively. We call a nonce *weak* if it induces a prescribed matrix in the final affine layer. To obtain the required number of weak nonces, we first fix a target matrix $\mathbf{M}_d$ and then sample nonces until sufficiently many of them induce this matrix. Since the final-round matrix is determined solely by the public nonce through the XOF and is independent of the secret key, weak nonces can be identified entirely offline.

Online Phase. In the online phase, we use the weak nonces to obtain the corresponding keystreams. Since all selected nonces induce the same final affine matrix, the contribution of the secret key to the first coefficient of the input to the final non-linear layer is captured by a common quantity σ . Guessing σ allows us to invert the final cube transformation and obtain a system of polynomial equations of degree at most 2^{d-1} in the $2n$ secret-key coefficients. We solve the resulting system using linearization.

Extended Attack. We further extend the attack by fixing the matrices in both the initial and final affine layers. This more restrictive choice of nonces allows us to peel off the first non-linear layer in addition to the final one, reducing the degree of the resulting equations from at most 2^{d-1} to at most 2^{d-2} .

In offline phase, we search for nonces that simultaneously induce prescribed target matrices in the initial and final affine layers. This increases the offline search cost, but provides a substantial reduction in the size of the polynomial system used in the online phase.

In online phase, by introducing the quadratic terms arising from the first Feistel-type layer as additional variables, we express the state after the first non-linear layer using $4n$ linearized variables. The remaining rounds before the final non-linear layer then yield equations of degree at most 2^{d-2} in these $4n$ variables.

Our Results. The **Iasta-3**, **Iasta-4**, and **Iasta-5** instances are defined over the finite-field polynomial ring

$$\mathcal{R}_p = \mathbb{F}_p[X]/(X^n + 1),$$

with $p = 2^{16} + 1$, $n = 128$ for **Iasta-3**, and $n = 64$ for **Iasta-4** and **Iasta-5**. The designers' security analysis considers the optimistic setting $\omega = 2$ and claims a security level of 128 bits for **Iasta-3** and **Iasta-4**. In this work, we show that this claimed security level does not withstand our algebraic attack.

The main results of our attacks are summarized in [Table 1](#) and described below.

- For **Iasta-3** and **Iasta-4**, we obtain best attack complexities of 2^{59} (see [subsection 3.3](#)) and 2^{67} (see [subsection 3.4](#)) operations, respectively, under $\omega = 2$, reducing the claimed 128-bit security level by 69 and 61 bits, respectively. Even under the conservative setting $\omega = 3$, our attack achieves complexities of approximately 2^{80} and 2^{82} operations for **Iasta-3** and **Iasta-4**, respectively, both of which remain below the claimed 128-bit security level.
- For **Iasta-5**, the extended attack achieves a best complexity of 2^{99} operations under $\omega = 2$ (see [subsection 3.4](#)). Although no overall security level is explicitly specified for **Iasta-5** in [\[YCY⁺26\]](#), if a 128-bit security level is intended, our attack would invalidate it.

Table 1: Best estimated time complexities of our attacks on **Iasta-3**, **Iasta-4**, and **Iasta-5**, obtained for $\omega = 2$.

Instance	Attack	Time complexity
Iasta-3	Basic (subsection 3.3)	2^{59}
Iasta-4	Extended (subsection 3.4)	2^{67}
Iasta-5	Extended (subsection 3.4)	2^{99}

Organization: The remainder of this paper is organized as follows. [Section 2](#) provides a detailed description of the **Iasta** stream cipher. [Section 3](#) presents the proposed algebraic attack and its complexity analysis. [Section 4](#) demonstrates the effectiveness of the attack on a toy version of the cipher. Finally, [Section 5](#) concludes the paper.

2 Description of Iasta

Iasta is a stream cipher defined over the finite-field polynomial ring

$$\mathcal{R}_p = \mathbb{F}_p[X]/(X^n + 1),$$

where p is a prime satisfying $\gcd(p - 1, 3) = 1$. An **Iasta** state consists of two elements of $\mathcal{R}_p$. For a d -round instance, it takes the secret key $\mathbf{k} = (k_0, k_1) \in \mathcal{R}_p^2$, and a public nonce $\mathbf{nc}$. The output of **Iasta** is used as the keystream and is denoted by

$$\mathbf{z} = \text{Iasta}_d(\mathbf{k}, \mathbf{nc}) \in \mathcal{R}_p^2.$$

The overall structure of d -round **Iasta** is illustrated in [Figure 1](#) and the detailed procedure is described in the following.

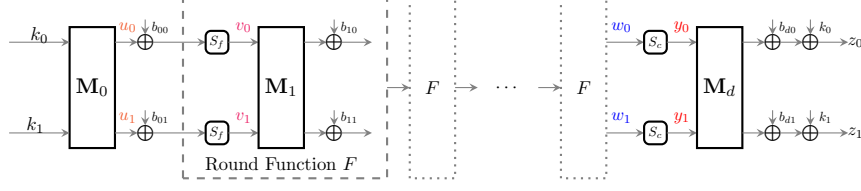


Figure 1: Illustration of d -round **Iasta**

Generation of Affine Layers. The affine transformation in **Iasta** is of the form

$$\mathcal{A}_{i,\text{nc}}(\mathbf{k}) = \mathbf{M}_{i,\text{nc}}\mathbf{k} + \mathbf{b}_{i,\text{nc}},$$

where $\mathbf{M}_{i,\text{nc}} \in \mathcal{R}_p^{2 \times 2}$ is an invertible linear transformation and $\mathbf{b}_{i,\text{nc}} \in \mathcal{R}_p^2$ is a round constant. Both $\mathbf{M}_{i,\text{nc}}$ and $\mathbf{b}_{i,\text{nc}}$ are derived from the output of an XOOF (SHAKE-256) instantiated with the public nonce nc . The matrix $\mathbf{M}_{i,\text{nc}}$ is obtained by alternating Θ -transformations with word rotations. In general, it is given by

$$\mathbf{M}_{i,\text{nc}} = \mathbf{P}_{\Theta_r} \mathbf{R}_{v_{r-1}} \mathbf{P}_{\Theta_{r-1}} \cdots \mathbf{R}_{v_2} \mathbf{P}_{\Theta_2} \mathbf{R}_{v_1} \mathbf{P}_{\Theta_1},$$

where r denotes the number of Θ -transformations. We have $r = 4$ for the 3-round instance and $r = 3$ for the 4- and 5-round instances of **Iasta**, respectively. For $j \in \{1, \dots, r\}$, the Θ -transformation is determined by $p_{\Theta_j}(X) = X^{\alpha_{3,j}} + X^{\alpha_{2,j}} + X^{\alpha_{1,j}} + 1$, where $1 \leq \alpha_{1,j} \leq 3$, $4 \leq \alpha_{2,j} \leq 6$, and $\alpha_{3,j} = 7 + (2\alpha_{1,j} + \alpha_{2,j} + 1 \bmod 3)$. The corresponding matrix is

$$\mathbf{P}_{\Theta_j} = \begin{pmatrix} p_{\Theta_j} + 1 & p_{\Theta_j} \\ p_{\Theta_j} & p_{\Theta_j} + 1 \end{pmatrix}.$$

The word-rotation matrices are of the form

$$\mathbf{R}_{v_j} = \begin{pmatrix} X^{\mu_0^{(j)}} + 1 & X^{\mu_1^{(j)}} \\ X^{\mu_1^{(j)}} & X^{\mu_0^{(j)}} + 1 \end{pmatrix}.$$

The rotation parameters are chosen as

$$\begin{aligned} (\mu_0^{(1)}, \mu_1^{(1)}) &= (5 + i_1, 10 + i_2), & 0 \leq i_1, i_2 \leq 4, \\ (\mu_0^{(2)}, \mu_1^{(2)}) &= (14 + j_1, 28 + j_2), & 0 \leq j_1, j_2 \leq 13, \\ (\mu_0^{(3)}, \mu_1^{(3)}) &= (33 + \epsilon_1, 66 + \epsilon_2), & 0 \leq \epsilon_1, \epsilon_2 \leq 32, \end{aligned}$$

where the third pair is used only in **Iasta-3**.

Non-linear Layers. **Iasta** consists of two types of non-linear layers: a Feistel-type transformation $S_f : \mathcal{R}_p \rightarrow \mathcal{R}_p$ and a Cube transformation $S_c : \mathcal{R}_p \rightarrow \mathcal{R}_p$, defined respectively as

$$S_f : x = (x_0, \dots, x_{n-1}) \mapsto \begin{cases} x_0, & i = 0, \\ x_i + x_{i-1}^2, & 1 \leq i < n, \end{cases}$$

and

$$S_c : x = (x_0, \dots, x_{n-1}) \mapsto (x_0^3, \dots, x_{n-1}^3).$$

The transformation S_f is applied in the first $d - 1$ rounds, whereas S_c is applied in the final round.

Parameter Sets. The recommended parameter sets for **Iasta-3**, **Iasta-4**, and **Iasta-5** given in [YCY+26] are summarized in Table 2. The overall security level of **Iasta-5** is not explicitly specified in [YCY+26]. However, in Section V.B, the designers claim that **Iasta-5** provides 290-bit security against algebraic attacks.

Table 2: Parameter sets of **Iasta-3**, **Iasta-4** and **Iasta-5**.

Parameter	Iasta-3	Iasta-4	Iasta-5
Field size (p)	65537	65537	65537
Ring dimension (n)	128	64	64
Number of rounds (d)	3	4	5
Θ -iterations (r)	4	3	3
Key size	4096 bits	2048 bits	2048 bits
State size	4096 bits	2048 bits	2048 bits
Matrix randomness	29 bits	22 bits	22 bits
Claimed security	128 bits	128 bits	Not specified

3 Algebraic Cryptanalysis of Iasta

In this section, we first examine the limited randomness of the matrices used in the affine layers of **Iasta** and show how this property can be exploited to obtain lower-degree equations.

3.1 A Discussion on the Randomness of the Iasta Matrices

We first observe that the Θ -transformations and word rotations commute with each other. Indeed, for any j and ℓ , we have

$$\mathbf{P}_{\Theta_j} = \begin{pmatrix} p_{\Theta_j} + 1 & p_{\Theta_j} \\ p_{\Theta_j} & p_{\Theta_j} + 1 \end{pmatrix} = (p_{\Theta_j} + 1)\mathbf{I} + p_{\Theta_j}\mathbf{J},$$

and

$$\mathbf{R}_{v_\ell} = \begin{pmatrix} X^{\mu_0^{(\ell)}} + 1 & X^{\mu_1^{(\ell)}} \\ X^{\mu_1^{(\ell)}} & X^{\mu_0^{(\ell)}} + 1 \end{pmatrix} = (X^{\mu_0^{(\ell)}} + 1)\mathbf{I} + X^{\mu_1^{(\ell)}}\mathbf{J},$$

where $\mathbf{I}$ is the identity matrix and

$$\mathbf{J} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Since any two matrices of the form $a_1\mathbf{I} + a_2\mathbf{J}$ and $b_1\mathbf{I} + b_2\mathbf{J}$, with $a_1, a_2, b_1, b_2 \in \mathcal{R}_p$, commute, we can rearrange the alternating product defining $\mathbf{M}_{i,\text{nc}}$ as

$$\mathbf{M}_{i,\text{nc}} = \underbrace{\mathbf{P}_{\Theta_r} \mathbf{P}_{\Theta_{r-1}} \cdots \mathbf{P}_{\Theta_1}}_{\mathbf{P}_{\text{nc}}} \underbrace{\mathbf{R}_{v_{r-1}} \mathbf{R}_{v_{r-2}} \cdots \mathbf{R}_{v_1}}_{\mathbf{R}_{\text{nc}}}.$$

We refer to $\mathbf{P}_{\text{nc}}$ and $\mathbf{R}_{\text{nc}}$ as the Θ -part and rotation part of $\mathbf{M}_{i,\text{nc}}$, respectively. We will consider the number of distinct Θ -parts and rotation parts separately.

There are $3 \times 3 = 9$ possible Θ -transformations, since $(\alpha_1, \alpha_2) \in \{1, 2, 3\} \times \{4, 5, 6\}$ uniquely determines α_3 . Moreover, the Θ -matrices commute. Hence, an r -fold product of Θ -matrices is determined by the multiplicity of each of the 9 possible matrices. Consequently, by the stars-and-bars argument, the number of possible Θ -parts is

$$|\mathcal{P}^{(r)}| = \binom{9 + r - 1}{r} = \binom{r + 8}{r},$$

provided that distinct multisets yield distinct products. In particular, $|\mathcal{P}^{(4)}| = \binom{12}{4} = 495$ for **Iasta-3** and $|\mathcal{P}^{(3)}| = \binom{11}{3} = 165$ for **Iasta-4** and **-5**.

We next consider the rotation part. Let

$$\mathcal{R}_j = \left\{ \mathbf{R}_{v_j} : (\mu_0^{(j)}, \mu_1^{(j)}) \text{ satisfy the prescribed parameter ranges} \right\}.$$

From the given parameter ranges, we have

$$|\mathcal{R}_1| = 5^2, \quad |\mathcal{R}_2| = 14^2, \quad |\mathcal{R}_3| = 33^2.$$

Since the rotation matrices commute, the rotation part can be represented as a product of matrices from these sets. Thus, the number of possible rotation parts is

$$|\mathcal{R}^{(r-1)}| = \prod_{j=1}^{r-1} |\mathcal{R}_j|.$$

Consequently, the total number of possible matrices $\mathbf{M}_{i,\text{nc}}$ is

$$|\mathcal{P}^{(4)}| |\mathcal{R}^{(3)}| = 2^{31.30}$$

for **Iasta-3**, and

$$|\mathcal{P}^{(3)}| |\mathcal{R}^{(2)}| = 2^{19.62}$$

for **Iasta-4** and **Iasta-5**. We have also experimentally verified these matrix-space sizes for all **Iasta** instances.

[[YCY+26](#)] claims 29 bits and 22 bits of matrix randomness for **Iasta-3** and **Iasta-4/5**, respectively. Our enumeration, however, reveals a discrepancy with these claims: the matrix space contains $2^{31.30}$ possibilities for **Iasta-3**, exceeding the claimed 2^{29} , whereas it contains only $2^{19.62}$ possibilities for **Iasta-4/5**, falling short of the claimed 2^{22} .

3.2 Peeling off the Last Non-Linear Layer when $\mathbf{M}_{d,\text{nc}}$ is fixed

Consider a keystream

$$\mathbf{z} = (z_0, z_1) = \text{Iasta}_d(\mathbf{k}, \text{nc}).$$

Let $\mathbf{w} = (w_0, w_1)$ denote the output of the $(d-1)$ -th round, and let $\mathbf{y} = (y_0, y_1)$ denote the input to the last affine layer. For an illustration of $\mathbf{w}$ and $\mathbf{y}$ see [Figure 1](#). Then

$$\mathbf{y} = \mathbf{M}_{d,\text{nc}}^{-1}(\mathbf{z} - \mathbf{b}_{d,\text{nc}} - \mathbf{k}) = \mathbf{M}_{d,\text{nc}}^{-1}(\mathbf{z} - \mathbf{b}_{d,\text{nc}}) - \mathbf{M}_{d,\text{nc}}^{-1}\mathbf{k}.$$

Thus, taking the constant coefficient of the first component, we obtain

$$y_{00} = \underbrace{\left(\mathbf{M}_{d,\text{nc}}^{-1}(\mathbf{z} - \mathbf{b}_{d,\text{nc}})\right)[0][0]}_c - \underbrace{\left(\mathbf{M}_{d,\text{nc}}^{-1}\mathbf{k}\right)[0][0]}_\sigma,$$

where $c \in \mathbb{F}_p$ is determined by the known quantities $\mathbf{z}$, $\mathbf{M}_{d,\text{nc}}$, and $\mathbf{b}_{d,\text{nc}}$, while $\sigma \in \mathbb{F}_p$ is a linear expression in the coefficients of the secret key. Hence,

$$y_{00} = c - \sigma.$$

Since the final non-linear layer is the cube transformation S_c , we have

$$w_{00} = y_{00}^{1/3} = (c - \sigma)^{1/3}. \quad (1)$$

Observe that w_{00} can be expressed as a polynomial of degree 2^{d-1} in the $2n$ key coefficients

$$k_{00}, \dots, k_{0(n-1)}, k_{10}, \dots, k_{1(n-1)}.$$

Thus, if we guess $\sigma \in \mathbb{F}_p$, the above relation yields a degree- 2^{d-1} equation in these $2n$ key unknowns.

Now, suppose that we have multiple keystreams

$$\mathbf{z}^{(i)} = \text{Iasta}_d(\mathbf{k}, \text{nc}^{(i)}), \quad i = 1, 2, \dots, m,$$

such that the matrix used in the last affine layer is the same for all these keystreams, i.e.,

$$\mathbf{M}_{d,\text{nc}^{(i)}} = \mathbf{M}_{d,\text{nc}^{(j)}} = \mathbf{M}_d \quad \text{for all } 1 \leq i, j \leq m.$$

We then guess

$$\sigma = (\mathbf{M}_d^{-1}\mathbf{k})[0][0] \in \mathbb{F}_p.$$

For each keystream $\mathbf{z}^{(i)}$, we obtain a corresponding value $c^{(i)}$ and hence a degree- 2^{d-1} equation

$$w_{00}^{(i)} = (c^{(i)} - \sigma)^{1/3}.$$

Therefore, m keystreams with the same last-round matrix yield m equations of degree 2^{d-1} in the same $2n$ key unknowns.

3.3 Attack on Iasta

In our attack, we assume that the attacker has full control over the public nonce $\mathbf{nc}$. Thus, our attack is conducted in the *chosen-nonce attack* model, which is commonly used in the analysis of FHE-friendly ciphers [LKSM24, LWN⁺25]. Recall that, in order to generate multiple lower-degree equations, we require multiple keystreams whose last-round matrices are identical. This condition can be satisfied during the offline phase, since the last-round matrices are independent of the secret key $\mathbf{k}$ and are determined solely by the public XOF output, whose input is the controllable nonce $\mathbf{nc}$.

However, generating the corresponding keystreams requires knowledge of the secret key $\mathbf{k}$ and therefore can only be performed during the online phase. Accordingly, our attack naturally consists of two phases: an offline phase, in which suitable nonces yielding the same last-round matrix are identified, and an online phase, in which the corresponding keystreams are generated and used to construct the algebraic equations. We describe these two phases in detail below.

3.3.1 Offline Phase

Our attack uses linearization to solve the resulting polynomial equations. Thus, the number of equations N_E must be at least the number of non-constant monomials, denoted by N_M . Since each equation has degree at most 2^{d-1} in the $2n$ key unknowns, we have

$$N_M = \binom{2n + 2^{d-1}}{2n} - 1.$$

Therefore, we require

$$N_E \geq \binom{2n + 2^{d-1}}{2n} - 1.$$

To obtain N_E nonces $\mathbf{nc}$ yielding the same last-round matrix, we randomly sample nonces and compute the corresponding last-round matrices using the XOF. Specifically, we first select a target matrix $\mathbf{M}_d$ from the matrix space of **Iasta**. We then sample random nonces $\mathbf{nc}$ and retain those for which the XOF generates the target matrix. We refer to such nonces as *weak nonces*.

Cost of the Offline Phase. As discussed in [subsection 3.1](#), the matrix space of **Iasta** contains 2^τ possible matrices, where

$$\tau = \begin{cases} 31.30, & \text{for Iasta-3,} \\ 19.62, & \text{for Iasta-4 and Iasta-5.} \end{cases}$$

Assuming that the matrices generated from the XOF are uniformly distributed over this space, a randomly chosen nonce produces the target matrix with probability $2^{-\tau}$. Hence, the expected number of nonce trials required to obtain one

weak nonce is 2^τ , and obtaining N_E weak nonces requires, on average, $\mathcal{O}(N_E 2^\tau)$ XOF evaluations. Thus, the time complexity of the offline phase is

$$T_{\text{offline}} = \mathcal{O}(N_E 2^\tau).$$

3.3.2 Online Phase

After the offline phase, we have N_E weak nonces for which the corresponding last-round matrices are identical to the predefined matrix $\mathbf{M}_d$. In the online phase, we use these nonces to generate the required keystreams and construct the resulting system of polynomial equations.

Generating Keystreams. For each weak nonce nc , we perform the following steps:

Step 1: Compute the matrices and round constants $\mathbf{M}_{i,\text{nc}}$ and $\mathbf{b}_{i,\text{nc}}$ for $i = 0, 1, \dots, d$ from $\text{XOF}(\text{nc})$.

Step 2: Compute the keystream

$$\mathbf{z} = \text{Iasta}_d(\mathbf{k}, \text{nc}) \quad \text{and} \quad c = \left(\mathbf{M}_{d,\text{nc}}^{-1} (\mathbf{z} - \mathbf{b}_{d,\text{nc}}) \right) [0][0].$$

Step 3: Store

$$(c, \mathbf{M}_{0,\text{nc}}, \dots, \mathbf{M}_{d-1,\text{nc}}, \mathbf{b}_{0,\text{nc}}, \dots, \mathbf{b}_{d-1,\text{nc}})$$

in a table $\mathcal{L}$.

Constructing and Solving the Equations. Once $\mathcal{L}$ contains N_E entries, we enumerate the possible values of

$$\sigma = (\mathbf{M}_d^{-1} \mathbf{k}) [0][0] \in \mathbb{F}_p.$$

For each guess $\sigma \in \mathbb{F}_p$, we perform the following steps:

Step 1. For each entry of $\mathcal{L}$, reconstruct the input $\mathbf{w} = (w_0, w_1)$ to the final non-linear layer in terms of $2n$ unknowns $(k_{00}, \dots, k_{0(n-1)}, k_{10}, \dots, k_{1(n-1)})$ and impose the equation

$$w_{00} = (c - \sigma)^{1/3}.$$

Here, w_{00} is a polynomial of degree at most 2^{d-1} in the $2n$ key coefficients. Thus, each entry yields a polynomial equation of degree at most 2^{d-1} .

Step 2. Add the consistency equation

$$(\mathbf{M}_d^{-1} \mathbf{k}) [0][0] = \sigma$$

to the constructed system and solve the resulting polynomial system.

Step 3. For each candidate secret key $\mathbf{k}$ obtained under the guess σ , verify its correctness using a fresh keystream.

Solving Equations with the Linearization Technique. Observe that we do not need to retain the table $\mathcal{L}$ after constructing the linearized system. The corresponding polynomials w_{00} and values c can be generated on the fly from the weak nonces and used immediately to construct the corresponding rows of the linearized system. Hence, apart from the memory required to store the linearized system, these data do not incur any additional memory requirement.

Since the equations involve $2n$ unknowns and have degree at most 2^{d-1} , the number of non-constant monomials is

$$N_M = \binom{2n + 2^{d-1}}{2n} - 1.$$

Thus, we set

$$N_E = N_M = \binom{2n + 2^{d-1}}{2n} - 1$$

equations to obtain a square linearized system. After linearization, the resulting system can be written as

$$\mathbf{W}\mathbf{x} = \mathbf{t},$$

where $\mathbf{x}$ contains the linearized monomials in the $2n$ key unknowns. Importantly, for the fixed set of weak nonces, the coefficient matrix $\mathbf{W}$ is independent of the guess $\sigma \in \mathbb{F}_p$, whereas the target vector $\mathbf{t}$ depends on σ . For each possible value $\sigma \in \{0, 1, \dots, p-1\}$, denote the corresponding target vector by $\mathbf{t}_\sigma$, and define

$$\mathbf{T} = [\mathbf{t}_0, \mathbf{t}_1, \dots, \mathbf{t}_{p-1}] \in \mathbb{F}_p^{N_M \times p}.$$

Since $\mathbf{W}$ is fixed and independent of σ , we compute $\mathbf{W}^{-1}$ only once¹, at a cost of

$$\mathcal{O}(N_M^\omega), \quad 2 \leq \omega \leq 3.$$

We then compute $\mathbf{W}^{-1}\mathbf{T}$. Using fast matrix multiplication with exponent ω , the product of an $N_M \times N_M$ matrix and an $N_M \times p$ matrix (where $N_M > p$) can be computed in $\mathcal{O}(N_M^2 p^{\omega-2})$ arithmetic operations. This bound is achieved by partitioning the matrices into square blocks of size $p \times p$, resulting in $(N_M/p)^2$ multiplications of $p \times p$ matrices, each costing $\mathcal{O}(p^\omega)$. Therefore, considering all p possible values of σ simultaneously, the total complexity of the linearized solving phase is

$$\mathcal{O}(N_M^\omega + N_M^2 p^{\omega-2}).$$

The costs of generating the required keystreams and constructing the polynomial systems are negligible compared with the complexity of solving the resulting systems. Consequently, the online complexity is

$$T_{\text{online}} = \mathcal{O}(N_M^\omega + N_M^2 p^{\omega-2}).$$

The memory required to store the linearized coefficient matrix and its inverse is

$$M_{\text{online}} = \mathcal{O}(N_M^2).$$

¹Since $\mathbf{W}$ behaves like a random matrix, it is full rank with high probability. If $\mathbf{W}$ is rank-deficient, we return to the offline phase and collect additional weak nonces.

3.3.3 Impact on the security of Iasta

Based on the above analysis, the overall time and memory complexities of our attack on the d -round **Iasta** are given by

$$T_{\text{attack}} = T_{\text{offline}} + T_{\text{online}} = \mathcal{O}(N_M 2^\tau + N_M^\omega + N_M^2 p^{\omega-2})$$

and

$$M_{\text{attack}} = \mathcal{O}(N_M^2),$$

where

$$2 \leq \omega \leq 3, \quad N_M = \binom{2n + 2^{d-1}}{2n} - 1, \quad \tau = \begin{cases} 31.30, & \text{for Iasta-3,} \\ 19.62, & \text{for Iasta-4 and -5.} \end{cases}$$

The parameters n and d of **Iasta-3**, **Iasta-4** and **Iasta-5** are given in [Table 2](#), with $p = 2^{16} + 1$ for all instances. For **Iasta-3**, we have $n = 128$ and $d = 3$, yielding

$$N_M = \binom{256 + 4}{256} - 1 \approx 2^{27.47}.$$

For **Iasta-4**, we have $n = 64$ and $d = 4$, yielding

$$N_M = \binom{128 + 8}{128} - 1 \approx 2^{41.10}.$$

For **Iasta-5**, we have $n = 64$ and $d = 5$, yielding

$$N_M = \binom{128 + 16}{128} - 1 \approx 2^{69.22}.$$

[Table 3](#) summarizes the estimated time complexity of our attack for different values of the algebraic constant ω . Both **Iasta-3** and **Iasta-4** claim 128-bit security; however, our attack achieves complexities below 2^{128} for all considered values of ω , thus falling below the claimed security level. In contrast, the attack does not achieve a complexity below 2^{128} for **Iasta-5** under any of the considered values of ω . Thus, if a 128-bit security level is assumed for **Iasta-5**, peeling off the last non-linear layer does not break this security level.

3.4 Extending the Attack by Peeling off the First Non-linear Layer

When the initial-round matrix $\mathbf{M}_{0,\text{nc}}$ is fixed in addition to the final-round matrix $\mathbf{M}_{d,\text{nc}}$, we can further peel off the first non-linear layer. Consider a class of nonces for which

$$\mathbf{M}_{0,\text{nc}} = \mathbf{M}_0 \quad \text{and} \quad \mathbf{M}_{d,\text{nc}} = \mathbf{M}_d$$

are fixed. Let

$$\mathbf{u} = \mathbf{M}_0 \mathbf{k}$$

Table 3: Estimated time complexities of our attack on **Iasta-3**, **Iasta-4**, and **Iasta-5** for different values of the algebraic constant ω . The complexities for **Iasta-3** and **Iasta-4** are below their claimed 128-bit security level. For comparison, the complexities obtained for **Iasta-5** remain above 2^{128} for all considered values of ω .

Algebraic constant (ω)	Iasta-3	Iasta-4	Iasta-5
$\omega = 2$	2^{59}	2^{83}	2^{139}
$\omega = 2.8$	2^{77}	2^{115}	2^{194}
$\omega = 3$	2^{82}	2^{123}	2^{208}

denote the output of the initial linear transformation, and let $\mathbf{v} = (v_0, v_1)$ denote the output of the first non-linear layer. For an illustration of $\mathbf{u}$ and $\mathbf{v}$ see [Figure 1](#). For $i \in \{0, 1\}$ and $0 \leq j < n$, we have

$$v_{i,j} = \begin{cases} u_{i,0} + b_{0,i,0}, & j = 0, \\ u_{i,j} + b_{0,i,j} + (u_{i,j-1} + b_{0,i,j-1})^2, & 1 \leq j < n. \end{cases} \quad (2)$$

Thus, $\mathbf{v}$ is quadratic in the coefficients of $\mathbf{u}$, and the only quadratic terms are of the form $u_{i,j}^2$. We can therefore linearize these terms by introducing $u_{i,j}^2$ as additional variables. In other words, $\mathbf{v}$ is linear in the $4n$ variables²

$$\{u_{i,j}, u_{i,j}^2 : i \in \{0, 1\}, 0 \leq j < n\}.$$

Since $\mathbf{M}_0$ is fixed and invertible, the key can be expressed as the linear function

$$\mathbf{k} = \mathbf{M}_0^{-1} \mathbf{u}.$$

We next consider the computation from $\mathbf{v}$ to the input $\mathbf{w}$ of the final non-linear layer. After the first non-linear layer, there remain $d - 2$ non-linear layers before the final affine transformation. Each Feistel-type layer at most doubles the algebraic degree. Consequently, $\mathbf{w}$ has degree at most 2^{d-2} in the coefficients of $\mathbf{v}$. Since each coefficient of $\mathbf{v}$ is linear in the variables $\{u_{i,j}, u_{i,j}^2\}$, the same degree bound holds when $\mathbf{w}$ is expressed in these $4n$ variables.

From the fixed final-round matrix, as described in [subsection 3.2](#), we again obtain

$$w_{00} = (c - \sigma)^{1/3},$$

where $\sigma = (\mathbf{M}_d^{-1} \mathbf{k})[0][0]$. Hence, after peeling off both the first and final non-linear layers, the above equation has degree at most 2^{d-2} in the $4n$ variables $\{u_{i,j}, u_{i,j}^2\}$, where each $u_{i,j}^2$ is treated as an independent variable. The corresponding number of non-constant monomials is therefore at most

$$N_M^{\text{Ext}} = \binom{4n + 2^{d-2}}{4n} - 1.$$

²Observe that $u_{0,0}^2$ and $u_{1,0}^2$ do not occur in $\mathbf{v}$. Thus, the actual number of variables is $4n - 2$. For simplicity, we use $4n$ variables, which does not affect our complexity estimates.

We now modify the offline and online phases accordingly.

Offline Phase. In the offline phase, we first fix two target matrices $\mathbf{M}_0$ and $\mathbf{M}_d$ from the corresponding matrix space of **Iasta**. We then search for a class of weak nonces of size N_M^{Ext} for which the XOF generates both target matrices in the initial and final affine layers, respectively. Assuming that the matrices generated by the XOF are independently and uniformly distributed over the corresponding matrix spaces, a randomly chosen nonce produces both target matrices with probability $2^{-2\tau}$. Hence, the expected number of nonce trials required to obtain N_M^{Ext} such weak nonces is

$$T_{\text{offline}}^{\text{Ext}} = \mathcal{O}(N_M^{\text{Ext}} 2^{2\tau}), \quad \tau = \begin{cases} 31.30, & \text{for Iasta-3,} \\ 19.62, & \text{for Iasta-4 and -5.} \end{cases}$$

Online Phase. The “keystream-generation procedure” remains unchanged from the basic attack. The only modification concerns Step 1 of the “Constructing and Solving the Equations” procedure. In this step, we express the output $\mathbf{v}$ of the first non-linear layer, as defined in Equation 2, in terms of the $4n$ variables

$$\{u_{i,j}, u_{i,j}^2 : i \in \{0, 1\}, 0 \leq j < n\},$$

where $u_{i,j}^2$ is treated as an independent variable. This allows us to reconstruct the input $\mathbf{w}$ to the final non-linear layer as a function of these $4n$ variables. We then impose the equation

$$w_{00} = (c - \sigma)^{1/3}$$

on these variables. The remaining steps of the attack are unchanged. Consequently, the complexity of the modified online phase is

$$T_{\text{online}}^{\text{Ext}} = \mathcal{O}((N_M^{\text{Ext}})^\omega + (N_M^{\text{Ext}})^2 p^{\omega-2}).$$

Impact of the Extended Attack. The overall time complexity of the extended attack on a d -round **Iasta** is

$$T_{\text{attack}}^{\text{Ext}} = T_{\text{offline}}^{\text{Ext}} + T_{\text{online}}^{\text{Ext}} = \mathcal{O}(N_M^{\text{Ext}} 2^{2\tau} + (N_M^{\text{Ext}})^\omega + (N_M^{\text{Ext}})^2 p^{\omega-2}),$$

where

$$2 \leq \omega \leq 3, \quad N_M^{\text{Ext}} = \binom{4n + 2^{d-2}}{4n} - 1, \quad \tau = \begin{cases} 31.30, & \text{for Iasta-3,} \\ 19.62, & \text{for Iasta-4 and -5.} \end{cases}$$

For **Iasta-3**, we have $n = 128$ and $d = 3$, yielding

$$N_M^{\text{Ext}} = \binom{512 + 2}{512} - 1 \approx 2^{17}.$$

For **Iasta-4**, we have $n = 64$ and $d = 4$, yielding

$$N_M^{\text{Ext}} = \binom{256 + 4}{256} - 1 \approx 2^{27.47}.$$

For **Iasta-5**, we have $n = 64$ and $d = 5$, yielding

$$N_M^{\text{Ext}} = \binom{256 + 8}{256} - 1 \approx 2^{48.90}.$$

Table 4 compares the estimated complexities of the basic and extended attacks. For **Iasta-4**, peeling off the first non-linear layer provides a substantial improvement: the estimated complexity decreases from 2^{83} to 2^{67} for $\omega = 2$, from 2^{115} to 2^{77} for $\omega = 2.8$, and from 2^{123} to 2^{82} for $\omega = 3$. For **Iasta-5**, the extended attack also provides a substantial improvement, reducing the estimated complexity from 2^{139} to 2^{99} for $\omega = 2$. Thus, if a 128-bit security level is intended for **Iasta-5**, our attack would invalidate this security level under the assumption $\omega = 2$. In contrast, for **Iasta-3**, the extended attack achieves an estimated complexity of 2^{80} for all three values of ω . This improves upon the basic attack only for $\omega = 3$, where the complexity decreases from 2^{82} to 2^{80} . For $\omega = 2$ and $\omega = 2.8$, the extended attack for **Iasta-3** is more expensive because the cost of finding weak nonces that simultaneously fix the initial- and final-round matrices in offline phase dominates the overall attack complexity.

Table 4: Estimated time complexities of the basic (subsection 3.3) and extended attacks (subsection 3.4) on **Iasta-3**, **Iasta-4** and **Iasta-5** for different values of the algebraic constant ω .

ω	Iasta-3		Iasta-4		Iasta-5	
	Basic	Extended	Basic	Extended	Basic	Extended
2	2^{59}	2^{80}	2^{83}	2^{67}	2^{139}	2^{99}
2.8	2^{77}	2^{80}	2^{115}	2^{77}	2^{194}	2^{137}
3	2^{82}	2^{80}	2^{123}	2^{82}	2^{208}	2^{147}

4 Experimental Verification

To empirically validate our linearization attack, we implemented both the offline and online phases described in subsection 3.3 using C program on a scaled-down “toy” instance of **Iasta**. The cipher parameters were set to $n = 2$, $d = 3$, and $r = 3$, with a 40-bit nonce space and modulus $p = 2^{16} + 1$.

Under these parameters, each resulting equation has a degree bound of $2^{d-1} = 4$. For $2n = 4$ key unknowns, the total number of non-constant monomials is

$$N_M = \binom{2n + 2^{d-1}}{2n} - 1 = \binom{4 + 4}{4} - 1 = 69.$$

Thus, a minimum of $N_E = 69$ linearly independent equations (and hence 69 weak nonces) is required to uniquely determine the key via linearization. We conducted 100 independent experiments and report the average results below.

Offline Phase. We fixed an arbitrary target matrix $\mathbf{M}_d$ generated by the XOF for the final affine layer. By sequentially sampling 40-bit nonces and evaluating their corresponding final-round matrices, we searched for nonces that produced the exact target matrix $\mathbf{M}_d$. As discussed in [subsection 3.1](#), the size of the matrix space for $r = 3$ is $2^{19.62}$. Therefore, the theoretical expected number of XOF evaluations required to find $N_E = 69$ weak nonces is $69 \times 2^{19.62} \approx 2^{25.73}$. In our experiments, successfully locating these 69 weak nonces required, on average, $72,857,239 \approx 2^{26.11}$ XOF evaluations. This experimental result closely matches our theoretical estimate.

Online Phase. For each weak nonce identified in the offline phase, we computed the keystream $\mathbf{z}$ under a fixed secret key $\mathbf{k} = (k_{00}, k_{01}, k_{10}, k_{11}) \in \mathbb{F}_p^4$ and extracted the constant $c = (\mathbf{M}_d^{-1}(\mathbf{z} - \mathbf{b}_d)) [0][0]$. We then symbolically evaluated the cipher state up to the input of the final non-linear layer, expressing w_{00} as a multivariate polynomial of degree at most 4 in the 4 key variables.

By mapping the $N_M = 69$ non-constant monomials to independent linear variables, we constructed the coefficient matrix $\mathbf{W} \in \mathbb{F}_p^{69 \times 69}$. Across all 100 experiments, we observed that $\text{rank}(\mathbf{W}) = 69 = N_M$. Thus, $\mathbf{W}$ has full rank in every experiment, confirming that the resulting linearized system can be uniquely solved and that the attack successfully recovers the secret key.

5 Conclusion

As a strategy for efficiently randomizing affine layers in FHE-friendly ciphers, the use of nonce-dependent matrices relies on the assumption that the resulting matrix space provides sufficient randomness. Our results reveal a structural weakness in **Iasta** arising from the restricted space of these matrices. By exploiting multiple nonces that induce identical affine-layer matrices, we can peel off nonlinear layers and derive low-degree equations in the secret key, leading to substantially improved algebraic attacks. As a consequence, under $\omega = 2$, we obtain attack complexities of 2^{59} , 2^{67} , and 2^{99} operations for **Iasta-3**, **Iasta-4**, and **Iasta-5**, respectively, all below the 128-bit security level. These results highlight the importance of carefully evaluating the effective randomness of nonce-dependent matrices when designing and analyzing FHE-friendly ciphers.

References

- [ACG⁺19] Martin R. Albrecht, Carlos Cid, Lorenzo Grassi, Dmitry Khovratovich, Reinhard Lüftenegger, Christian Rechberger, and Markus Schofnegger. Algebraic cryptanalysis of stark-friendly designs: Application to marvellous and mimc. In Steven D. Galbraith and Shihō Moriai, editors, *Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019*,

Proceedings, Part III, volume 11923 of *Lecture Notes in Computer Science*, pages 371–397. Springer, 2019.

- [AMT22] Tomer Ashur, Mohammad Mahzoun, and Dilara Toprakhisar. Chaghri: A FHE-friendly block cipher. In *CCS '22: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 139–150. Association for Computing Machinery, 2022. Full version: IACR ePrint 2022/592.
- [ARS⁺15] Martin R. Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. Ciphers for MPC and FHE. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 430–454. Springer, 2015.
- [BBB⁺25] Jules Baudrin, Sonia Belaïd, Nicolas Bon, Christina Boura, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain, Yann Rotella, and Samuel Tap. Transistor: a TFHE-friendly stream cipher. In Yael Tauman Kalai and Seny F. Kamara, editors, *Advances in Cryptology - CRYPTO 2025*, volume 16004 of *Lecture Notes in Computer Science*, pages 530–565. Springer, Cham, 2025. Full version: IACR ePrint 2025/282.
- [CCH⁺24] Mingyu Cho, Woohyuk Chung, Jincheol Ha, Jooyoung Lee, Eun-Gyeol Oh, and Mincheol Son. FRAS: TFHE-friendly cipher based on random s-boxes. *IACR Transactions on Symmetric Cryptology*, 2024(3):1–43, 2024. Full version: IACR ePrint 2024/745.
- [CHK⁺21] Jihoon Cho, Jincheol Ha, Seongkwang Kim, ByeongHak Lee, Joohee Lee, Jooyoung Lee, Dukjae Moon, and Hyojin Yoon. Transciphering framework for approximate homomorphic encryption. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part III*, volume 13092 of *Lecture Notes in Computer Science*, pages 640–669. Springer, 2021.
- [CHMS22] Orel Cosserson, Clément Hoffmann, Pierrick Méaux, and François-Xavier Standaert. Towards case-optimized hybrid homomorphic encryption - featuring the elisabeth stream cipher. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology - ASIACRYPT 2022, Proceedings, Part III*, volume 13793 of *Lecture Notes in Computer Science*, pages 32–67. Springer Cham, 2022. Full version: IACR ePrint 2022/180.

- [CIR22] Carlos Cid, John Petter Indrøy, and Håvard Raddum. FASTA - A stream cipher for fast FHE evaluation. In Steven D. Galbraith, editor, *Topics in Cryptology - CT-RSA 2022 - Cryptographers' Track at the RSA Conference 2022, Virtual Event, March 1-2, 2022, Proceedings*, volume 13161 of *Lecture Notes in Computer Science*, pages 451–483. Springer, 2022.
- [CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, Cambridge, MA, 3 edition, 2009.
- [DEG⁺18] Christoph Dobraunig, Maria Eichlseder, Lorenzo Grassi, Virginie Lallemand, Gregor Leander, Eik List, Florian Mendel, and Christian Rechberger. Rasta: A cipher with low anddepth and few ands per bit. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 662–692. Springer, 2018.
- [DGH⁺23] Christoph Dobraunig, Lorenzo Grassi, Lukas Helming, Christian Rechberger, Markus Schofnegger, and Roman Walch. Pasta: A case for hybrid homomorphic encryption. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2023(3):30–73, 2023.
- [Din21] Itai Dinur. Cryptanalytic applications of the polynomial method for solving multivariate equation systems over $\text{GF}(2)$. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021, Proceedings, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 374–403. Springer, 2021. Preprint: IACR ePrint 2021/578.
- [DLR16] Sébastien Duval, Virginie Lallemand, and Yann Rotella. Cryptanalysis of the FLIP family of stream ciphers. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016, Proceedings, Part I*, volume 9814 of *Lecture Notes in Computer Science*, pages 457–475. Springer, Berlin, Heidelberg, 2016. Preprint: IACR ePrint 2016/271.
- [GAH⁺23] Lorenzo Grassi, Irati Manterola Ayala, Martha Norberg Hovd, Morten Øygarden, Håvard Raddum, and Qingju Wang. Cryptanalysis of symmetric primitives over rings and a key recovery attack on rubato. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023, Proceedings, Part III*, volume 14083 of *Lecture Notes in Computer Science*, pages 305–339. Springer, Cham, 2023. Full version: IACR ePrint 2023/822.

- [GBJR23] Henri Gilbert, Rachelle Heim Boissier, J  r  my Jean, and Jean-Ren   Reinhard. Cryptanalysis of Elisabeth-4. In Jian Guo and Ron Steinfeld, editors, *Advances in Cryptology – ASIACRYPT 2023, Proceedings, Part III*, volume 14440 of *Lecture Notes in Computer Science*, pages 256–284. Springer, 2023. Preprint: IACR ePrint 2023/1436.
- [GLR⁺25] Lorenzo Grassi, Fukang Liu, Christian Rechberger, Fabian Schmid, Roman Walch, and Qingju Wang. Minimize the randomness in rasta-like designs: How far can we go? Application to PASTA. In Maria Eichlseder and S  bastien Gambs, editors, *Selected Areas in Cryptography – SAC 2024*, volume 15517 of *Lecture Notes in Computer Science*, pages 207–238. Springer, Cham, 2025. Full version: IACR ePrint 2024/791.
- [HKC⁺20] Jincheol Ha, Seongkwang Kim, Wonseok Choi, Jooyoung Lee, Dukjae Moon, Hyojin Yoon, and Jihoon Cho. Masta: An he-friendly cipher using modular arithmetic. *IEEE Access*, 8:194741–194751, 2020.
- [HKL⁺22] Jincheol Ha, Seongkwang Kim, ByeongHak Lee, Jooyoung Lee, and Mincheol Son. Rubato: Noisy ciphers for approximate homomorphic encryption. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part I*, volume 13275 of *Lecture Notes in Computer Science*, pages 581–610. Springer, 2022.
- [HL20] Phil Hebborn and Gregor Leander. Dasta - alternative linear layer for rasta. *IACR Trans. Symmetric Cryptol.*, 2020(3):46–86, 2020.
- [LIM21] Fukang Liu, Takanori Isobe, and Willi Meier. Cryptanalysis of full LowMC and LowMC-M with algebraic techniques. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology – CRYPTO 2021*, pages 368–401, Cham, 2021. Springer International Publishing.
- [LKSM24] Fukang Liu, Abul Kalam, Santanu Sarkar, and Willi Meier. Algebraic attack on FHE-friendly cipher HERA using multiple collisions. *IACR Transactions on Symmetric Cryptology*, 2024(1):214–233, 2024. Preprint: IACR ePrint 2023/1800.
- [LSMI21] Fukang Liu, Santanu Sarkar, Willi Meier, and Takanori Isobe. Algebraic attacks on rasta and dasta using low-degree equations. In Mehdi Tibouchi and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2021 - 27th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 6-10, 2021, Proceedings, Part I*, volume 13090 of *Lecture Notes in Computer Science*, pages 214–240. Springer, 2021.

- [LSW⁺22] Fukang Liu, Santanu Sarkar, Gaoli Wang, Willi Meier, and Takanori Isobe. Algebraic meet-in-the-middle attack on LowMC. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022, Proceedings, Part I*, volume 13791 of *Lecture Notes in Computer Science*, pages 225–255. Springer, 2022. Preprint: IACR ePrint 2022/019.
- [LWN⁺25] Fukang Liu, Tianning Wang, Jianqiang Ni, Daisuke Yamamoto, Guilherme T. S. Sato, Haoyang Wang, and Willi Meier. Chosen-iv algebraic attack on randomized ciphers FASTA and HERA. *IACR Transactions on Symmetric Cryptology*, 2025(4):357–380, December 2025.
- [RST18] Christian Rechberger, Hadi Soleimany, and Tyge Tiessen. Cryptanalysis of low-data instances of full LowMCv2. *IACR Transactions on Symmetric Cryptology*, 2018(3):163–181, 2018. Preprint: IACR ePrint 2018/859.
- [Str69] Volker Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13(4):354–356, August 1969.
- [YCY⁺26] Bo Yang, Liquan Chen, Chenyue Yin, Ziyang Zhang, and Shang Gao. Iasta: An efficient cipher for hybrid homomorphic encryption in cloud-assisted iot. *IEEE Transactions on Dependable and Secure Computing*, 2026.