Higher-Order Differential Attacks on FHE-Friendly Cipher YuX

No Author Given

No Institute Given

Abstract. Recent applications of advanced cryptographic protocols like fully homomorphic encryption (FHE) and zero-knowledge proofs (ZKP) have led to the development of new symmetric primitives over large finite fields, known as arithmetization-oriented (AO) ciphers. These designs, focused on minimizing field multiplications, are highly susceptible to algebraic attacks, particularly higher-order differential attacks. YuX is an FHE friendly SPN-based block cipher proposed by Liu et al. in IEEE Trans. Inf. Theory. Its internal state is defined over $\mathbb{F}_q^{16}$, where q can be $2^8, 2^{16}$, or a prime number $p = 65537$, corresponding to three variants: $\mathsf{Yu}_2\mathsf{X}$-8, $\mathsf{Yu}_2\mathsf{X}$-16, and $\mathsf{Yu}_p\mathsf{X}$. By using an S-box derived from a Nonlinear Feedback Shift Register (NLFSR), YuX reduces multiplicative complexity and circuit depth.
This paper focuses on the analysis of $\mathsf{Yu}_2\mathsf{X}$-8 and $\mathsf{Yu}_2\mathsf{X}$-16, collectively referred to as $\mathsf{Yu}_2\mathsf{X}$. While the designers claim that all YuX variants have at most 6-round integral distinguishers, in order to exploit higher-order differential properties to find longer distinguishers for $\mathsf{Yu}_2\mathsf{X}$. We propose a new technique based on the concept of *exponent sets* to efficiently estimate the upper bound on the algebraic degree of the $\mathsf{Yu}_2\mathsf{X}$ round function. By tracking the evolution of exponent sets across rounds, we formally prove that the algebraic degree of $\mathsf{Yu}_2\mathsf{X}$ increases linearly. We further validate our theoretical findings using the general monomial prediction technique along with the actual round function zero-sum verification. Based on the upper bounds on algebraic degree, we construct longer-round high-order differential distinguishers, and develop the first third-party key-recovery attacks on $\mathsf{Yu}_2\mathsf{X}$, specifically for 7-round $\mathsf{Yu}_2\mathsf{X}$-8 and 11-round $\mathsf{Yu}_2\mathsf{X}$-16 high-order differential distinguishers and key recovery attacks. These results provide new insights into the algebraic structure and security margin of $\mathsf{Yu}_2\mathsf{X}$.

Keywords: YuX· Algebraic degree · Higher-order differential attack.

1 Introduction

Fully Homomorphic Encryption (FHE) allows computations directly over encrypted data, making it ideal for privacy-preserving applications. However, FHE has traditionally been hindered by performance limitations and significant ciphertext expansion, especially in lattice-based schemes like BFV [20], BGV [7], CKKS [9], and TFHE [10]. To address these challenges, Hybrid Homomorphic

Encryption (HHE) or transciphering has been introduced [32]. In HHE, computational tasks are split between the client and the server: the client handles lightweight operations, while the server performs the more computationally intensive tasks. Instead of encrypting messages directly with lattice-based schemes, the client uses a symmetric-key cipher to encrypt the plaintext and encrypts the symmetric key with FHE. The server then performs homomorphic decryption on the encrypted key and evaluates the encrypted messages, significantly reducing communication overhead and keeping ciphertext size nearly proportional to the original message.

However, while Hybrid Homomorphic Encryption effectively addresses ciphertext expansion, the process still requires efficient homomorphic decryption before any evaluation can be performed. Therefore, optimizing homomorphic decryption becomes critical for improving overall performance. This need has led to the development of so-called FHE-friendly symmetric-key encryption schemes, many of which have emerged in recent years. These include schemes such as LowMC [2], FiLP [8,31], Rasta [17] and its derivatives [14,22,18], Chaghri [3], HERA [11], Rubato [21], Elisabeth-4 [15], FRAST [12] and Transistor [4], among others. These designs often incorporate randomness in various components, such as randomized affine transformations, key schedules, or S-box constructions. Although these ciphers have demonstrated superior performance compared to traditional schemes like AES in the FHE setting, many have failed to maintain security under cryptanalytic scrutiny, especially against algebraic attacks such as higher-order differential attacks [27,28], and have been quickly broken [30,29]

Related Works. Higher-order differential attacks exploit insufficient growth in the algebraic degrees of cryptographic primitives. Lai [25] first discovered this in the context of traditional symmetric cryptography, and Eichlseder et al. [19] extended the concept to the cipher over $\mathbb{F}_{2^n}$, revealing the linear growth of MiMC's algebraic degree. Further analysis of SPN ciphers over $\mathbb{F}_{2^n}$ was presented by Cid et al. [13], while Bouvier et al. [6] studied the algebraic degree growth of MiMC and determined its exact value. In [16], Cui et al. extended the monomial prediction method [23] to $\mathbb{F}_{2^n}$, applying general monomial prediction to evaluate the algebraic degrees of MiMC, Feistel MiMC, and GMiMC, leading to tighter bounds. Beyne et al. [5] extended the integral attack to large finite fields over any characteristic. Liu et al. [27] introduced the coefficient grouping technique at EUROCRYPT 2023 to simplify the evaluation of algebraic degrees, applying it to launch a high-order differential attack on the FHE-friendly block cipher Chaghri. This method successfully broke Chaghri and proposed a new affine layer to exponentially increase its algebraic degree. At CRYPTO 2023, Liu et al. [28] analyzed algebraic degree growth for SPN-based AO ciphers, and proposed a variant of the coefficient grouping technique for efficiently calculating upper bounds of algebraic degrees for arbitrary affine layers, though with relaxed constraints.

Motivations. YuX [26] is a block cipher designed to be friendly for FHE applications. The designers proposed three variants of YuX, each operating over a different finite field depending on the choice of the state representation. Specifically, the cipher variants are denoted as $\mathsf{Yu_2X}$-8, $\mathsf{Yu_2X}$-16, and $\mathsf{Yu}_p\mathsf{X}$, corresponding to the fields $\mathbb{F}_{2^8}$, $\mathbb{F}_{2^{16}}$, and $\mathbb{F}_{65537}$, respectively. In terms of throughput for FHE evaluation, $\mathsf{Yu_2X}$-8 and $\mathsf{Yu_2X}$-16 significantly outperform AES-128, Chaghri, and LowMC-128 by approximately 12×, 17×, and 9×, respectively. The 128-bit security round count for $\mathsf{Yu_2X}$ is 12 rounds, with the specific parameters detailed in Table 2.

In terms of resistance against integral attacks, the designers claim that for $\mathsf{Yu_2X}$-8, at least 2^{127} plaintexts are required to find a 6-round integral distinguisher. However, this conclusion is based solely on conventional search techniques for integral distinguishers, without incorporating algebraic degree-based upper bound analysis over large finite fields—particularly the kind of analysis suited for "Big Fields" like $\mathbb{F}_{2^{16}}$ or beyond. As a result, the claimed security margins require further validation. Similarly, for $\mathsf{Yu_2X}$-16, the designers assert that integral distinguishers do not exist beyond 6 rounds. Yet, this claim also lacks support from rigorous algebraic degree propagation analysis and thus remains open to scrutiny.

Contributions. In this work, we analyze the round function of $\mathsf{Yu_2X}$ and observe that, unlike many FHE-friendly ciphers such as Chaghri [3] and MiMC [1], which use power mappings in their S-box $x \mapsto x^d$, $\mathsf{Yu_2X}$'s S-box does not rely on power mappings. As a result, traditional analysis methods are not directly applicable to $\mathsf{Yu_2X}$. To enable a tailored security analysis, we introduce the concept of an *exponent set* to accurately track the evolution of the algebraic degree. Furthermore, we formalize the propagation rules of exponent sets to characterize how the algebraic degree changes across rounds.

Specifically, we prove that under the univariate setting, the algebraic degree upper bound of the round function grows linearly, increasing by 2 per round. In the multi-variable setting, the algebraic degree of $\mathsf{Yu_2X}$ grows linearly in the worst case, with an increase of $2s$ per round, where s is the number of active input words. We further use generalized monomial prediction techniques and conduct specific experiments to validate our theoretical analysis. Based on these results, we construct longer-round high-order differential distinguishers and develop the first third-party key-recovery attacks on $\mathsf{Yu_2X}$. Specifically, for the 7-round and 10-round high-order differential distinguishers of $\mathsf{Yu_2X}$-8 and $\mathsf{Yu_2X}$-16, the time complexities are 2^{127} and 2^{63}, respectively. For the 8-round and 11-round key recovery attacks, the time complexities are 2^{127} and 2^{96}, respectively. This results in $\mathsf{Yu_2X}$-16 having only one redundant round for security. A summary of the attack results is provided below and tabulated in Table 1.

Outline. In Sect. 2, we introduce the basic notation and relevant concepts, describing the block cipher YuX. In Sect. 3, we present the concept of exponent sets and use their properties to track the polynomials of the $\mathsf{Yu_2X}$ round function, proving that the upper bound of its algebraic degree grows linearly. At the end of Sect. 3, we apply generalized monomial prediction techniques to compute and verify our theoretical analysis. In Sect. 4, we use the algebraic degree

Table 1. Summary of high-order differential distinguisher and key-recover attack for YuX.

Cipher	Attack Type	Round	Time	Data	Reference
Yu$_2$X-8	Integral distinguisher	6/12	2^{127}	2^{127}**CP**	[26]
	HD distinguisher	**6**/12	$\mathbf{2^{31}}$	$\mathbf{2^{31}}$**CC**	**Sect. 4.1**
	HD distinguisher	**7**/12	$\mathbf{2^{127}}$	$\mathbf{2^{127}}$**CC**	**Sect. 4.1**
	Key recovery	**8**/12	$\mathbf{2^{127}}$	$\mathbf{2^{127}}$**CC**	**Sect. 4.2**
Yu$_2$X-16	HD distinguisher	**10**/12	$\mathbf{2^{63}}$	$\mathbf{2^{63}}$**CC**	**Sect. 4.1**
	Key recovery	**11**/12	$\mathbf{2^{96}}$	$\mathbf{2^{96}}$**CC**	**Sect. 4.2**

a/b: a is the number of attack rounds, and b is the total number of rounds.
HD: High-order differential.
CP: Chosen plaintext.
CC: Chosen ciphertext.

upper bound to construct high-order differential distinguishers for Yu$_2$X, and employ these distinguishers to launch key recovery attacks. Finally, in Sect. 5, we conclude the paper.

2 Preliminaries

2.1 Notation

The following notations will be used throughout the paper. $\mathbb{F}q$ denotes a finite field with q elements, where q is a power of a prime p. The notation $\mathbb{F}_{2^n}^t$ represents the product of the binary extension field $\mathbb{F}_{2^n}$ taken t times. To represent word vectors, bold italic lowercase letters are used. For example, $\boldsymbol{u} \in \mathbb{F}_{2^n}^m$ represents the m-word vector $(u_0, \ldots, u_{m-1})$, where the Hamming weight of $\boldsymbol{u}$ is denoted as $\mathrm{H}(\boldsymbol{u}) = \sum_{i=0}^{n-1} \mathrm{H}(u_i)$. Additionally, over the finite field $\mathbb{F}_{2^n}$, the operation $+$ denotes the XOR operation. The function $\mathrm{Mod}_n(x)$ is defined as $2^n - 1$ if $x \geq 2^n - 1$ and $2^n - 1 \mid x$, and as $x \pmod{2^n - 1}$ otherwise.

2.2 Algebraic Degree for Polynomials and Higher-order Differential Attack over Binary Extension Fields

To mount a higher-order differential attack, the first step is to find an upper bound of the algebraic degree of the polynomial representation of the output in terms of the input. In order to do this, in the following we heavily exploit the link between the algebraic degree of the functions over $\mathbb{F}_{2^n}$ and over $\mathbb{F}_2^n$. First, let us recall the two notions of degree that apply to a function over a finite field with characteristic 2.

Definition 1 (Univariate degree, Algebraic Degree, and Exponent Set). *For any univariate function $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ as*

$$F(X) = \sum_{e=0}^{2^n-1} \phi(e) X^e,$$

*where $\phi(e) \in \mathbb{F}_{2^n}$, for $e \in [0, 2^n - 1]$. The **univariate degree** of F denoted by $\mathcal{D}^u(F)$ is defined as:*

$$\mathcal{D}^u(F) = \max\{e : e \in [0, 2^n - 1], \phi_e \neq 0\}.$$

*The **algebraic degree** of F denoted by $\mathcal{D}^a(F)$ is defined as:*

$$\mathcal{D}^a(F) = \max\{H(e) : e \in [0, 2^n - 1], \phi_e \neq 0\}.$$

*We further define the **exponent set** of F, denoted by $\mathcal{E}(F)$, as the set of all exponents $e \in [0, 2^n - 1]$ such that $\phi_e \neq 0$, $\mathcal{E}(F) = \{e \in [0, 2^n - 1] \mid \phi_e \neq 0\}$. We also adopt the shorthand notation $\{\preceq e\}$ to represent the integer set $\{0, 1, \ldots, e\}$ when describing contiguous exponent intervals.*

The multivariate case. Let $\boldsymbol{F}(X_1, \ldots, X_t) : \mathbb{F}_{2^n}^t \to \mathbb{F}_{2^n}$ be a be a multivariate function over the quotient ring $\frac{\mathbb{F}_{2^n}[X_1,\ldots,X_t]}{\langle X_1^{2^n}+X_1,\ldots,X_t^{2^n}+X_t\rangle}$. Then $\boldsymbol{F}$ admits a unique polynomial representation:

$$\boldsymbol{F}(X_1, \ldots, X_t) = \sum_{\boldsymbol{e}=(e_1,\ldots,e_t)\in\mathbb{F}_{2^n}^t} \phi(\boldsymbol{e}) \cdot X_1^{e_1} \cdots X_t^{e_t},$$

where each coefficient $\phi(\boldsymbol{e}) \in \mathbb{F}_{2^n}$, and $\boldsymbol{e}$ ranges over exponent tuples.

We define the **exponent set** of $\boldsymbol{F}$, denoted $\mathcal{E}(\boldsymbol{F})$, as

$$\mathcal{E}(\boldsymbol{F}) = \{\boldsymbol{e} = (e_1, \ldots, e_t) \in \mathbb{F}_{2^n}^t \mid \phi(\boldsymbol{e}) \neq 0\}.$$

For multiple variables, we use $\preceq (e_1, e_2, \ldots, e_s)$ to denote the Cartesian product $\{0, 1, \ldots, e_1\} \times \{0, 1, \ldots, e_2\} \times \cdots \times \{0, 1, \ldots, e_s\}$. Then, the **algebraic degree** of $\boldsymbol{F}$, denoted $\mathcal{D}^a(\boldsymbol{F})$, is defined as:

$$\mathcal{D}^a(\boldsymbol{F}) = \max_{\boldsymbol{e}\in\mathcal{E}(\boldsymbol{F})} \left\{ \sum_{j=1}^{t} \mathrm{H}(e_j) \right\}.$$

Higher-order differential attack over binary extension fields. Higher-order differential attacks [25,24] are significant cryptographic attack techniques that exploit the low algebraic degree of nonlinear transformations, like classical block ciphers. When the algebraic degree of a Boolean function is sufficiently low, this method can differentiate it from a random function. Let $F : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ be a function with algebraic degree $\mathcal{D}^a(F)$. Then, by selecting an affine subspace of dimension $\geq \mathcal{D}^a(F)+1$, the sum of the function F over this subspace equals 0. In other words, for any affine subspace $\mathcal{V} \oplus c$ where the dimension is $\geq \mathcal{D}(F)^a + 1$ and $c \in \mathbb{F}_2^n$, we have:

$$\sum_{X\in\mathcal{V}\oplus c} F(X) = 0.$$

2.3 Description of YuX

The YuX [26] block cipher is constructed over the finite field $\mathbb{F}_q^{16}$, employing a Substitution–Permutation Network (SPN) structure. Depending on the choice of $q \in \{2^8, 2^{16}\}$, or the prime $p = 65537$, the cipher is referred to as $\mathsf{Yu_2X}$-8, $\mathsf{Yu_2X}$-16, or $\mathsf{Yu_pX}$, respectively. In the following sections, both $\mathsf{Yu_2X}$-8 and $\mathsf{Yu_2X}$-16 will be collectively referred to as $\mathsf{Yu_2X}$. The encryption process begins with an initial key whitening step, followed by multiple rounds of transformations. Each round consists of the following three layers:

- **Substitution Layer $\mathcal{SL}$**: The substitution layer partitions the 16-word input into four 4-word blocks and applies an S-box $\mathcal{S}$ to each block independently:

  $$\mathcal{SL}(\boldsymbol{x}) = (\mathcal{S}(x_0, \dots, x_3), \mathcal{S}(x_4, \dots, x_7), \mathcal{S}(x_8, \dots, x_{11}), \mathcal{S}(x_{12}, \dots, x_{15})).$$

  Here, the S-box $\mathcal{S}$ is a nonlinear permutation over $\mathbb{F}_q^4$, defined by: $\mathcal{S}(x_0, x_1, x_2, x_3) = \mathrm{Pf}^{-4}(x_0, x_1, x_2, x_3), \mathcal{S}^{-1}(x_0, x_1, x_2, x_3) = \mathrm{Pf}^4(x_0, x_1, x_2, x_3)$. The permutation Pf over $\mathbb{F}_q^4$ is defined as $\mathrm{Pf}(x_0, x_1, x_2, x_3) = (x_1, x_2, x_3, x_0 + x_1x_2 + x_3 + \alpha)$. Then, by recursive composition, the 4-fold application Pf^4 yields the output (y_0, y_1, y_2, y_3) as $\mathrm{Pf}^4(x_0, x_1, x_2, x_3) = (y_0, y_1, y_2, y_3)$, where:

  $$\begin{aligned} y_0 &= x_0 + x_1 \cdot x_2 + x_3 + \alpha \\ y_1 &= x_1 + x_2 \cdot x_3 + y_0 + \alpha \\ y_2 &= x_2 + x_3 \cdot y_0 + y_1 + \alpha \\ y_3 &= x_3 + y_0 \cdot y_1 + y_2 + \alpha \end{aligned} \tag{1}$$

- **Linear Transformation Layer $\mathcal{L}$**: This layer applies linear diffusion using the inverse of a fixed matrix $\mathbf{M} \in \mathbb{F}_q^{16\times16}$. The transformation is expressed as $\mathcal{L}(\boldsymbol{x}) = \mathbf{M}^{-1} \cdot \boldsymbol{x}^T$. The matrix $\mathbf{M}$ is designed such that:

  $$\mathbf{M} \cdot \boldsymbol{x}^T = \sum_{j\in\{0,3,4,8,9,12,14\}} \mathrm{Rot}_j(\boldsymbol{x}),$$

  where $\mathrm{Rot}_j(\boldsymbol{x})$ denotes a word-wise left rotation of the state by j positions. Consequently, the inverse linear operation is given directly by summing the rotated versions $\mathcal{L}^{-1}(\boldsymbol{x}) = \sum_{j\in\{0,3,4,8,9,12,14\}} \mathrm{Rot}_j(\boldsymbol{x})$.
- **Round Key Addition $\mathcal{RKA}$**: This step introduces round-dependent key material by adding the round key $\boldsymbol{k}^{(i+1)}$ to the state element-wise:

  $$\mathcal{RKA}^{i+1}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{k}^{i+1}$$

  with $\boldsymbol{k}^{i+1} = (k_0^{i+1}, k_1^{i+1}, \dots, k_{15}^{i+1}) \in \mathbb{F}_q^{16}$ being the round key derived from the key schedule.

Each round operates as follows: $R(\boldsymbol{x}^i) = \mathcal{RAK}^{i+1} \circ \mathcal{L} \circ \mathcal{SL}(\boldsymbol{x}^i)$, where $\boldsymbol{x}^i = (x_0^i, x_1^i, \dots, x_{15}^i) \in \mathbb{F}_q^{16}$ represents the state before round $i+1$. Since the final round lacks the linear transformation layer $\mathcal{L}$, the encryption algorithm for N_r

rounds of YuX can be expressed as:$\mathsf{YuX}_{enc}^{N_r}(\boldsymbol{P}) = \mathcal{RAK}^{N_r} \circ \mathcal{SL} \circ R \circ \cdots \circ R \circ \mathcal{RAK}^0(\boldsymbol{P})$.The decryption algorithm for N_r rounds of YuX is similarly represented as: $\mathsf{YuX}_{dec}^{N_r}(\boldsymbol{C}) = \mathcal{RAK}^{N_r} \circ \mathcal{SL}^{-1} \circ R^{-1} \circ \cdots \circ R^{-1} \circ \mathcal{RAK}^0(\boldsymbol{C})$. YuX's different variants, such as those defined in Table 2, are specified accordingly.

Table 2. Summary of YuX Cipher Variants

Cipher	Block	Defining Polynomial	Constant	Security Level	Rounds
Yu$_2$X-8	$\mathbb{F}_{2^8}^{16}$	$x^8 + x^4 + x^3 + x + 1$	205	128 bit	12
Yu$_2$X-16	$\mathbb{F}_{2^{16}}^{16}$	$x^{16} + x^{12} + x^3 + x + 1$	205	128 bit	12* 14
Yu$_p$X	$\mathbb{F}_{65537}^{16}$	—	205	118 bit* 128 bit	9* 14

$*$Recommended number of rounds for FHE applications.

3 Tracking the Algebraic Degree Upper Bounds of Yu$_2$X

Let the input to the cipher be denoted as $\mathbf{X} = (X_1, \ldots, X_{16}) \in \mathbb{F}_{2^n}^{16}$. The intermediate state before the S-box layer in round r is denoted by x_i^r for $1 \leq i \leq 16$, and the the output of the S-box layer is denoted by y_i^r.Each y_i^r is subsequently transformed by the linear layer and round key addition to produce the input to the next round, x_i^{r+1}. The round structure is illustrated in Figure .

Since the S-box in YuX does not rely on a power mapping, we propose a more tractable approach for analyzing its algebraic degree. Specifically, by symbolically evaluating the polynomial expressions of the S-box, we observe that the algebraic degree of Pf^4 is lower than that of Pf^{-4}. Therefore, we focus our analysis on the *decryption direction* of the cipher, in which the S-box is implemented as Pf^4, whose expression is given in Equation 1.

In the following, we analyze the growth of the algebraic degree upper bounds of the YuX round function under both the **univariate** and **multi-variable** settings.

3.1 The Algebraic Degree Upper Bounds in the Univariate Setting

In this scenario, we fix all but one input variable to a constant value (e.g., 1), and compute the algebraic degree of each output coordinate as a univariate polynomial. Let the cipher's input state be $(X, c_1, c_2, \ldots, c_{15})$, where X is a variable that ranges over all elements in $\mathbb{F}_{2^n}$, and $c_i \in \mathbb{F}_{2^n}$ are constants. Naturally, the variable X can be placed at any of the 16 positions. However, due to the rotational symmetry inherent in the design of YuX, placing the variable at position i or $i + 4$ (modulo 16) results in equivalent behavior under the cipher's round structure.

Moreover, our experiments show that placing the variable at the first position (i.e., index 0) results in a more favorable propagation of algebraic degree in subsequent rounds. This configuration leads to a sharper characterization of the upper bounds growth and provides a better foundation for constructing distinguishers or attacks. Therefore, without loss of generality, we choose to inject the variable into the first position of the input state for our analysis. For simplicity, we assume that round keys are independently and randomly generated. After the initial $\mathcal{RAK}$ operation, the first S-box in round 1 receives input $(x_0^0, x_1^0, x_2^0, x_3^0) = (X + k_0^0, k_1^0 + c_1, k_2^0 + c_2, k_3^0 + c_3)$. The output of this S-box is given by:

$$(y_0^0, y_1^0, y_2^0, y_3^0) = \mathcal{S}^{-1}(x_0^0, x_1^0, x_2^0, x_3^0) = \mathrm{Pf}^4(x_0^0, x_1^0, x_2^0, x_3^0).$$

To trace algebraic degree growth, we represent each intermediate value x_i^r or y_i^r as a univariate polynomial in X:

$$x_i^r = \sum_{j=1}^{|\mathcal{E}(x_i^r)|} a_{i,j}^r X^{e_{i,j}^r}, \quad y_i^r = \sum_{j=1}^{|\mathcal{E}(y_i^r)|} b_{i,j}^r X^{e_{i,j}^r},$$

where $a_{i,j}^r$, $b_{i,j}^r$ are coefficients dependent on the round key, and $\mathcal{E}(x_i^r) \subseteq \mathbb{N}$ is the exponent set:

$$\mathcal{E}(x_i^r) = \{e_{i,1}^r, e_{i,2}^r, \ldots, e_{i,d}^r\}.$$

To analyze the transformation of $\mathcal{E}$ through the S-box, we rely on the following algebraic properties of polynomial addition and multiplication.

Proposition 1. *Let $x = \sum a_i X^{e_{x,i}}$, $y = \sum b_j X^{e_{y,j}}$ be polynomials over $\mathbb{F}_{2^n}$. Then:*

- *(Addition) For $z = x + y$, the exponent set satisfies $\mathcal{E}(z) = \mathcal{E}(x) \cup \mathcal{E}(y)$.*
- *(Multiplication) For $z = x \cdot y$, the exponent set satisfies $\mathcal{E}(z) = \{Mod_n(e_{x,i} + e_{y,j}) \mid e_{x,i} \in \mathcal{E}(x),\ e_{y,j} \in \mathcal{E}(y)\}$.*

Proof. **(Addition)** Let $x = \sum_{i=1}^m a_i X^{e_{x,i}}$ and $y = \sum_{j=1}^n b_j p^{e_{y,j}}$. Then their sum is:

$$z = x + y = \sum_{i=1}^m a_i X^{e_{x,i}} + \sum_{j=1}^n b_j X^{e_{y,j}}.$$

The result z is also a polynomial in X, and its exponent set $\mathcal{E}(z)$ consists of the union of the distinct exponents appearing in both x and y, i.e.,

$$\mathcal{E}(z) = \mathcal{E}(x) \cup \mathcal{E}(y).$$

(Multiplication) The product of x and y is:

$$z = x \cdot y = \left(\sum_{i=1}^m a_i X^{e_{x,i}}\right) \cdot \left(\sum_{j=1}^n b_j X^{e_{y,j}}\right) = \sum_{i=1}^m \sum_{j=1}^n (a_i b_j) X^{e_{x,i}+e_{y,j}}.$$

Thus, the exponent set of z is the set of all possible sums of degrees from $\mathcal{E}(x)$ and $\mathcal{E}(y)$, that is:

$$\mathcal{E}(z) = \{\mathrm{Mod}_n(e_{x,i} + e_{y,j}) \mid e_{x,i} \in \mathcal{E}(x),\ e_{y,j} \in \mathcal{E}(y)\}.$$

This completes the proof.

These rules allow us to compute algebraic degree growth symbolically. We now unfold the internal computation of Pf^4, layer by layer:

$$\begin{aligned}
y_0^r &= \mathrm{Pf}(x_0^r, x_1^r, x_2^r, x_3^r) = x_0^r + x_1^r \cdot x_2^r + x_3^r + \alpha \\
&= \sum_{j=1}^{|\mathcal{E}(x_0^r)|} a_{0,j}^r X^{e_{0,j}^r} + \left(\sum_{j=1}^{|\mathcal{E}(x_1^r)|} a_{1,j}^r X^{e_{1,j}^r} \right) \cdot \left(\sum_{j=1}^{|\mathcal{E}(x_2^r)|} a_{2,j}^r X^{e_{2,j}^r} \right) + \sum_{j=1}^{|\mathcal{E}(x_3^r)|} a_{3,j}^r X^{e_{3,j}^r} + \alpha \\
&= \sum_i b_{0,i}^r X^{e_{0,i}^r}, \\
&\vdots \\
y_3^r &= \mathrm{Pf}(x_3^r, y_0^r, y_1^r, y_2^r) \\
&= x_3^r + y_0^r \cdot y_1^r + y_2^r + \alpha \\
&= \sum_{j=1}^{|\mathcal{E}(x_3^r)|} a_{3,j}^r X^{e_{3,j}^r} + \left(\sum_{j=1}^{|\mathcal{E}(y_0^r)|} b_{0,j}^r X^{e_{0,j}^r} \right) \cdot \left(\sum_{j=1}^{|\mathcal{E}(y_1^r)|} b_{1,j}^r X^{e_{1,j}^r} \right) + \sum_{j=1}^{|\mathcal{E}(y_2^r)|} b_{2,j}^r X^{e_{2,j}^r} + \alpha \\
&= \sum_{i=1}^{|\mathcal{E}(y_3^r)|} b_{3,i}^r X^{e_{3,i}^r}.
\end{aligned}$$

Exponent Set Propagation Through the S-Box. Based on the recursive structure of the S-box and the exponent set propagation rules established in Proposition 1, we now analyze how the exponent sets evolve through the computation of the S-box. Recall that each output coordinate y_i^r is computed from a combination of previous x_i^r and y_i^r variables via a nonlinear permutation Pf^4, involving addition and multiplication over polynomials in the symbolic input p.

Specifically, we have:

$$\begin{aligned}
\mathcal{E}(y_0^r) &= \mathcal{E}(x_0^r) \cup \mathcal{E}(x_1^r \cdot x_2^r) \cup \mathcal{E}(x_3^r), \\
\mathcal{E}(y_1^r) &= \mathcal{E}(x_1^r) \cup \mathcal{E}(x_2^r \cdot x_3^r) \cup \mathcal{E}(y_0^r), \\
\mathcal{E}(y_2^r) &= \mathcal{E}(x_2^r) \cup \mathcal{E}(x_3^r \cdot y_0^r) \cup \mathcal{E}(y_1^r), \\
\mathcal{E}(y_3^r) &= \mathcal{E}(x_3^r) \cup \mathcal{E}(y_0^r \cdot y_1^r) \cup \mathcal{E}(y_2^r).
\end{aligned}$$

By applying this to all four S-box instances in the 16-word state of the cipher, we can obtain all the exponent sets $\mathcal{E}(y_0^r), \ldots, \mathcal{E}(y_{15}^r)$. Since the S-boxes act independently, the global degree of the state after the nonlinear layer is simply the maximum among the degrees of all y_i^r.

Exponent Set Propagation Through the Linear Layer. Following the Substitution Layer, the YuX applies a linear diffusion layer based on the inverse of a fixed linear transformation $\mathcal{L}^{-1}$. This layer is defined over the state $\boldsymbol{y}^r = (y_1^r, y_1^r, \ldots, y_{16}^r)^T \in \mathbb{F}_{2^n}^{16}$, and produces the next round input $\boldsymbol{x}^{r+1} = (x_1^{r+1}, x_2^{r+1}, \ldots, x_{16}^{r+1})^T \in \mathbb{F}_{2^n}^{16}$ as follows:

$$\boldsymbol{x}^{r+1} = M^{-1} \cdot \boldsymbol{y}^r,$$

where $M^{-1} \in \mathbb{F}_q^{16\times 16}$ is a fixed invertible matrix derived from word rotations. For each index $i \in \{0, 1, \ldots, 15\}$, the transformation is given by:

$$x_i^{r+1} = \sum_{j\in\{0,3,4,8,9,12,14\}} y^r_{(i+j) \bmod 16}.$$

Thus, each output word is the sum of seven input words. Let $\mathcal{I}_i \subseteq \{0, \ldots, 15\}$ denote the set of indices involved in the computation of x_i^{r+1}:

$$\mathcal{I}_i = \{i, (i+3), (i+4), (i+8), (i+9), (i+12), (i+14)\} \bmod 16.$$

Given the additive structure of this layer and using Proposition 1, the exponent set after the linear layer satisfies:

$$\mathcal{E}(x_i^{r+1}) = \bigcup_{j\in\mathcal{I}_i} \mathcal{E}(y_j^r), \quad \text{for all } i = 0, \ldots, 15.$$

To prove that the algebraic degree of the round function in YuX grows linearly, with an increase of 2 per round, we first present the following lemma.

Lemma 1. *Let $\mathcal{E}(x) = \{\preceq e\} = \{0, 1, \ldots, e\}$, and define the algebraic degree $\mathcal{D}^a(x)$ as the maximum Hamming weight over all elements in $\mathcal{E}(x)$, i.e., $\mathcal{D}^a(x) = \max_{i\in\mathcal{E}(x)} H(i)$. Suppose a transformation causes the exponent set to expand to $\mathcal{E}(y) = \{0, 1, \ldots, m \cdot e\}$, for some integer $m \geq 1$. Then the algebraic degree satisfies:*

$$\mathcal{D}^a(y) \leq \mathcal{D}^a(x) + \lfloor \log_2(m+1) \rfloor.$$

Proof. Let $\mathcal{H}(n) = \max_{0\leq i\leq n} H(i)$ denote the maximum Hamming weight of all integers from 0 to n. Then we have:

$$\mathcal{D}^a(x) = \mathcal{H}(e), \quad \mathcal{D}^a(y) = \mathcal{H}(m \cdot e).$$

Observe that:

$$m \cdot e + 1 < (m+1)(e+1),$$

which implies:

$$\lfloor \log_2(m \cdot e + 1) \rfloor < \lfloor \log_2((m+1)(e+1)) \rfloor = \lfloor \log_2(m+1) + \log_2(e+1) \rfloor.$$

Since $\lfloor \log_2(e+1) \rfloor = \mathcal{D}^a(x)$, we conclude:

$$\mathcal{D}^a(y) = \mathcal{H}(m \cdot e) \leq \mathcal{D}^a(x) + \lfloor \log_2(m+1) \rfloor.$$

Hence, the algebraic degree increases by at most $\lfloor \log_2(m+1) \rfloor$.

□

Proposition 2. *In the univariate setting, the algebraic degree of the decryption direction in $\mathsf{Yu}_2\mathsf{X}$ grows linearly, with an increase of 2 per round.*

Proof. Recall that the algebraic degree $\mathcal{D}^a(x_i^r)$ of a variable x_i^r is defined as the maximum Hamming weight of the exponents in the exponent set $\mathcal{E}(x_i^r)$. Assume the input to the r-th round is $\boldsymbol{x}^r = (x_0^r, \ldots, x_{15}^r)$. Since $\boldsymbol{x}^r$ is obtained from $\boldsymbol{y}^{r-1}$ by the linear transformation, the exponent set of all state variables is the same, i.e.,

$$\mathcal{E}(x_i^r) = \{0, 1, \ldots, e\}.$$

According to the S-box exponent set propagation, we have:

$$\begin{aligned} \mathcal{E}(y_0^r) &= \mathcal{E}(x_0^r) \cup \mathcal{E}(x_1^r \cdot x_2^r) \cup \mathcal{E}(x_3^r), \\ \mathcal{E}(x_1^r \cdot x_2^r) &= \{e_i + e_j | e_i \in \mathcal{E}(x_1^r), e_j \in \mathcal{E}(x_2^r)\} = \{0, \ldots, 2e\}. \end{aligned}$$

Thus, $\mathcal{E}(x_1^r \cdot x_2^r) = \{0, \ldots, 2e\}$. By the Lemma 1 on degree propagation, we have:

$$\mathcal{D}^a(\{0, \ldots, 2e\}) \leq \mathcal{D}^a(\{0, \ldots, e\}) + 1,$$

which implies:

$$\mathcal{D}^a(y_0^r) \leq \mathcal{D}^a(x_0^r) + 1.$$

Similarly, for y_1^r, we have $\mathcal{D}^a(y_1^r) = \mathcal{D}^a(y_0^r)$. For y_2^r, we know:

$$\begin{aligned} \mathcal{E}(y_2^r) &= \mathcal{E}(x_2^r) \cup \mathcal{E}(x_3^r \cdot y_0^r) \cup \mathcal{E}(y_1^r), \\ \mathcal{E}(x_3^r \cdot y_0^r) &= \{e_i + e_j | e_i \in \mathcal{E}(x_3^r), e_j \in \mathcal{E}(y_0^r)\} = \{0, \ldots, 3e\}. \end{aligned}$$

Thus, the degree of y_2^r can increase by at most 2, i.e., $\mathcal{D}^a(y_2^r) \leq \mathcal{D}^a(x_2^r) + 2$. For y_3^r, a similar reasoning applies:

$$\begin{aligned} \mathcal{E}(y_3^r) &= \mathcal{E}(x_3^r) \cup \mathcal{E}(y_0^r \cdot y_1^r) \cup \mathcal{E}(y_2^r), \\ \mathcal{E}(y_0^r \cdot y_1^r) &= \{e_i + e_j | e_i \in \mathcal{E}(y_0^r), e_j \in \mathcal{E}(y_1^r)\} = \{0, \ldots, 4e\}. \end{aligned}$$

Therefore, the maximum increase in degree is 2, i.e., $\mathcal{D}^a(y_3^r) \leq \mathcal{D}^a(x_3^r) + 2$. Finally, since each state's exponent set is combined via linear transformation, the algebraic degree of the round function increases by at most 2 per round, showing that the degree growth is linear.

□

Remark 1. It is important to note that our proof assumes the variable X is sufficiently mixed within the 16-state system. In practice, if we assume the input is $(X, c_1, c_2, \ldots)$, the algebraic degree upper bounds does not immediately reach its upper bound of 2 in the initial rounds. Rather, it requires several rounds of mixing before the degree upper bounds attains this growth. Therefore, a detailed analysis of the algebraic degree upper bounds growth in the early rounds is necessary.

3.2 The Algebraic Degree Upper Bounds in the Multi-variable Setting

Having understood the algebraic degree upper bounds growth in the univariate setting, we further analyze the growth of the algebraic degree upper bounds in the multi-variable setting. Given that the $\mathsf{Yu_2X}$ state size is $\mathbb{F}_{2^n}^{16}$, we consider the number of input variables as s, with $(X_1, \ldots, X_s) \in \mathbb{F}_{2^n}^s$. The positions of the input variables are selected from the 16 available positions, with the remaining positions filled by random constant inputs. Since the substitution layer of $\mathsf{Yu_2X}$ consists of four parallel S-boxes, some of the input positions are equivalent for analysis purposes.

After r rounds, the input state to the round function is denoted by $\boldsymbol{x}^r = (x_0^r, \ldots, x_{15}^r)$. According to the definition of the multi-variable exponent set, we define the exponent set of the input state at round r as:

$$\mathcal{E}(x_i^r) = \{(e_1, \ldots, e_s) \in \mathbb{N}^s \mid \text{the monomial } X_1^{e_1} \cdots X_s^{e_s} \text{ appears in } x_i^r\}.$$

The algebraic degree of x_i^r, denoted $\mathcal{D}(x_i^r)$, is defined as the maximum total Hamming weight of the exponent vectors in $\mathcal{E}(x_i^r)$, i.e.,

$$\mathcal{D}(x_i^r) = \max_{(e_1,\ldots,e_s)\in\mathcal{E}(x_i^r)} \sum_{j=1}^{s} H(e_j).$$

Since $\boldsymbol{x}^r$ is obtained by applying a confusion operation to $\boldsymbol{y}^{r-1}$, we assume that the exponent set of each component of the state is identical.

Proposition 3. *In the multi-variable setting, the algebraic degree of the decryption direction in $\mathsf{Yu_2X}$ increases linearly, with an increase of $2s$ per round, where s is the number of input variables.*

Proof. The state $\boldsymbol{x}^r$ is passed through the S-box layer, where each variable undergoes nonlinear transformations. For the first S-box layer, the exponent set $\mathcal{E}(x_0^r)$ gets transformed by the S-box. The multiplication operation introduced by the S-box (such as $x_1^r \cdot x_2^r$) introduces new terms in the exponent set, with the exponent vector being scaled up, as follows:

$$\mathcal{E}(y_0^r) = \mathcal{E}(x_0^r) \cup \mathcal{E}(x_1^r \cdot x_2^r) \cup \mathcal{E}(x_3^r),$$

where the exponent set of $x_1^r \cdot x_2^r$ becomes:

$$\mathcal{E}(x_1^r \cdot x_2^r) = \{\preceq (2e_1, 2e_2, \ldots, 2e_s)\}.$$

From Lemma 1, we know that multiplying each exponent by 2 implies that the algebraic degree increases by at most 1. Therefore, the algebraic degree of y_0^r increases by s, i.e.,

$$\mathcal{D}^a(y_0^r) \leq \mathcal{D}^a(x_0^r) + s.$$

Similarly, for y_1^r, the exponent set is updated as follows $\mathcal{E}(y_1^r) = \mathcal{E}(x_1^r) \cup \mathcal{E}(x_2^r \cdot x_3^r) \cup \mathcal{E}(y_0^r)$. The multiplication $x_2^r \cdot x_3^r$ gives $\mathcal{E}(x_2^r \cdot x_3^r) = \{\preceq (2e_1, 2e_2, \ldots, 2e_s)\}$.

Again, following Lemma 1, the algebraic degree increases by at most s. Hence, $\mathcal{D}^a(y_1^r) \leq \mathcal{D}^a(y_0^r) + s$.

For y_2^r, we have $\mathcal{E}(y_2^r) = \mathcal{E}(x_2^r) \cup \mathcal{E}(x_3^r \cdot y_0^r) \cup \mathcal{E}(y_1^r)$, and the multiplication $x_3^r \cdot y_0^r$ results in $\mathcal{E}(x_3^r \cdot y_0^r) = \{\preceq (3e_1, 3e_2, \ldots, 3e_s)\}$. The algebraic degree increases by at most $2s$, so we have $\mathcal{D}^a(y_2^r) \leq \mathcal{D}^a(y_1^r) + 2s$.

The process continues similarly for y_3^r, where the exponent set becomes:

$$\mathcal{E}(y_3^r) = \mathcal{E}(x_3^r) \cup \mathcal{E}(y_0^r \cdot y_1^r) \cup \mathcal{E}(y_2^r),$$

with the multiplication $y_0^r \cdot y_1^r$ giving $\mathcal{E}(y_0^r \cdot y_1^r) = \{\preceq (4e_1, 4e_2, \ldots, 4e_s)\}$. The algebraic degree increases by at most $2s$, so:

$$\mathcal{D}^a(y_3^r) \leq \mathcal{D}^a(y_2^r) + 2s.$$

This pattern continues for subsequent rounds, where the algebraic degree grows by $2s$ per round due to the nonlinear operations and mixing of the variables. Therefore, the algebraic degree increases linearly with a maximum growth of $2s$ per round. Finally, the algebraic degree of the round function grows as $\mathcal{D}(x_i^{r+1}) \leq \mathcal{D}(x_i^r) + 2s$.

□

3.3 Refined Algebraic Degree Upper Bounds via Automated General Monomial Prediction Techniques

To precisely determine the upper bounds on the algebraic degree of Yu$_2$X, we need to identify at which round the algebraic degree upper bounds begins to grow linearly, as analyzed in the previous section. We effectively assessed the algebraic degree upper bounds of Yu$_2$X under different variable settings by utilizing Cui et al.'s general monomial prediction technique [16] with the Satisfiability Modulo Theories (SMT) automation method. Additionally, we validated that our proof regarding the growth of Yu$_2$X's algebraic degree upper bounds is correct.

The general monomial prediction technique is used to determine the presence of a specific monomial in the product of coordinate functions of $\boldsymbol{F} : \mathbb{F}_{2^n}^m \mapsto \mathbb{F}_{2^n}^n$, particularly when directly constructing it is computationally infeasible. We utilize this technique to more accurately evaluate the algebraic degree bound in the early rounds, when the input variables have not yet been fully mixed. Once the mixing is complete, the algebraic degree bound will increase linearly, as predicted by our theoretical analysis. The related concepts of monomial prediction and automated modeling can be found in the referenced literature [16].

Our experiments show that, under the univariate setting, the algebraic degree upper bounds of the round function in Yu$_2$X begins to increase linearly—by 2 per round—starting from round i. In the multi-variable setting, after a few rounds of confusion, the growth of the algebraic degree upper bound aligns with our theoretical analysis. To obtain the longest distinguisher, we only list the input variable position selections with the maximum analyzed round count, as shown in Table 3,Table 4. Due to space constraints, we only include the algebraic degree bounds for univariable and 2-variable settings. The remaining upper bounds, monomial prediction modeling source code, and zero-sum verification experiments, along with the code, are available at: `https://anonymous.4open.science/r/HD-YuX-94FC`

Table 3. The algebraic degree upper bounds of $\mathsf{Yu_2X}$ in univariate setting

Cipher	**Input index**	**Output index**	**Rounds**										
			0	1	2	3	4	5	6	7	8	9	10
Yu₂X-8	0/1/2	0	1	1	1	3	5	7	**8**	-	-	-	-
		1	1	1	2	3	5	7	**8**	-	-	-	-
		2/3	1	1	2	4	6	**8**	-	-	-	-	-
	3	0/1/2	1	1	2	4	6	**8**	-	-	-	-	-
		3	1	1	3	5	7	**8**	-	-	-	-	-
Yu₂X-16	0/1/2	0	1	1	1	3	5	7	9	11	13	15	**16**
		1	1	1	2	3	5	7	9	11	13	15	**16**
		2/3	1	1	2	4	6	8	10	12	14	**16**	-
	3	0/1/2	1	1	2	4	6	8	10	12	14	**16**	-
		3	1	1	3	5	7	9	11	13	15	**16**	-

Table 4. The algebraic degree upper bounds of $\mathsf{Yu_2X}$ in two variables setting

Cipher	**Input index**	**Output index**	**Rounds**										
			0	1	2	3	4	5	6	7	8	9	10
Yu₂X-8	(0,1)/(0,2)	0/1	1	1	2	5	9	13	**16**	-	-	-	-
		2	1	1	3	6	10	14	**16**	-	-	-	-
		3	1	2	3	7	11	15	**16**	-	-	-	-
Yu₂X-16	(0,1)/(0,2)	0/1	1	1	2	5	9	13	17	21	25	29	**32**
		2	1	1	3	6	10	14	18	22	26	30	**32**
		3	1	2	3	7	11	15	19	23	27	31	**32**

4 Higher-order Differential Distinguishers and Key Recovery Attack on Round-Reduced $\mathsf{Yu_2X}$

In the previous section, we exploited the structural property that the S-box of YuX does not rely on a power function, and introduced the notion of the exponent set to track the evolution of algebraic degrees. This allowed us to characterize the growth of the algebraic degree and prove that the degree bound increases linearly with the number of rounds. To further validate our theoretical findings, we employed the general monomial prediction Technique to confirm the correctness of our bound and determine from which round the degree begins to increase uniformly.

In this section, we leverage the established algebraic degree upper bounds to construct high-order differential distinguishers for Yu$_2$X. Based on these distinguishers, we develop key-recovery attacks. Specifically, we mount a 7-round key-recovery attack on Yu$_2$X-8 and a 11-round key-recovery attack on Yu$_2$X-16, with time complexities of 2^{56} and 2^{100}, respectively.

4.1 Higher-order Differential Distinguishers of Round-Reduced Yu$_2$X

Based on the algebraic degree bound evaluated in the previous section, we can easily derive the higher-order differential distinguishers for Yu$_2$X. As the designers claim that the security level of Yu$_2$X-16 is 128 bits, the maximum data complexity is 2^{127}. We derive the longest higher-order differential distinguishers for both Yu$_2$X-8 and Yu$_2$X-16, which will be used in the key recovery attack in the next section. The longest higher-order differential distinguishers for different data complexities are summarized in Table 5.

For Yu$_2$X-8, the longest 6-round higher-order differential distinguisher can be found, with both time and data complexities being 2^{31}, and the zero-sum positions located at $i \equiv \pmod 4$. For Yu$_2$X-16, the longest 10-round higher-order differential distinguisher can be found, with both time and data complexities being 2^{63}, and the zero-sum positions located at $i \equiv \pmod 4$. For the distinguishers used in key recovery attacks, we aim for all positions to exhibit the zero-sum property. Therefore, the longest distinguishers for key recovery are 6 rounds for Yu$_2$X-8 and 10 rounds for Yu$_2$X-16, with time and data complexities of 2^{48} and 2^{92}, respectively.

4.2 Key Recovery Attack on Round-Reduced Yu$_2$X

Based on the higher-order differential distinguisher in Sect. 4.1, we can apply it to perform key recovery attacks on Yu$_2$X. The overall approach involves adding one more round to the r-round higher-order distinguisher, utilizing the zero-sum property to construct key-related equations and solving them to recover the key. Since we are attacking the decryption function of Yu$_2$X, this key recovery attack is a chosen ciphertext attack.

Strategy of the Attack. Assume that we aim to use an r_{HD}-round high-order differential distinguisher and extend it to r_{KR} rounds for performing key recovery attacks over $r = r_{HD} + r_{KR}$ rounds. Typically $r_{KR} \in \{1, 2\}$. The overall attack framework is shown in the Figure 1. The key recovery attack consists of the following two parts:

1. **Constructing a multivariate equation system related to the round keys using the zero-sum property of the higher-order differential distinguisher.** The state word size of Yu$_2$X is 16. Suppose we can use an affine subspace $\mathcal{V} + c \in \mathbb{F}_{2^m}$ with dimension m to construct an r_{HD}-round

Table 5. High-order differential distinguishers of Yu$_2$X with different data complexities

Cipher	No.of variables	Time	Data	Data limit	Rounds
Yu$_2$X-8	1	2^{8}	2^{8}	2^{8}	5
	2	2^{16}	2^{16}	2^{16}	5
	3	2^{23}	2^{23}	2^{24}	5
	4	2^{31}	2^{31}	2^{32}	6
	5	2^{37}	2^{37}	2^{40}	6
	6	2^{42}	2^{42}	2^{48}	6
	7	2^{49}	2^{49}	2^{56}	6
	8	2^{54}	2^{54}	2^{64}	6
	9	2^{59}	2^{59}	2^{72}	6
	10	2^{64}	2^{64}	2^{80}	6
	11	2^{69}	2^{69}	2^{88}	6
	12	2^{75}	2^{75}	2^{96}	6
	13	2^{81}	2^{81}	2^{104}	6
	14	2^{87}	2^{87}	2^{112}	6
	15	2^{97}	2^{97}	2^{120}	6
Yu$_2$X-16	1	2^{16}	2^{16}	2^{16}	9
	2	2^{32}	2^{32}	2^{32}	9
	3	2^{43}	2^{43}	2^{48}	9
	4	2^{63}	2^{63}	2^{64}	10
	5	2^{77}	2^{77}	2^{80}	10
	6	2^{85}	2^{85}	2^{96}	10
	7	2^{105}	2^{105}	2^{112}	10

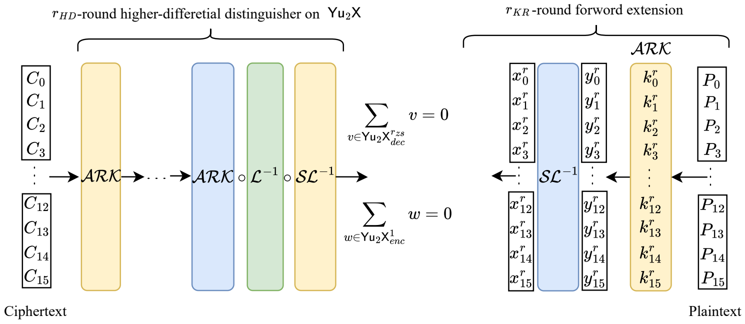
r_{HD}-round higher-differetial distinguisher on Yu$_2$X
r_{KR}-round forword extension
$\mathcal{ARK}$
C_0 C_1 C_2 C_3
C_{12} C_{13} C_{14} C_{15}
$\mathcal{ARK}$
$\mathcal{ARK} \circ \mathcal{L}^{-1} \circ \mathcal{SL}^{-1}$
$\sum_{v \in \mathsf{Yu_2X}^{r_{zs}}_{dec}} v = 0$
$\sum_{w \in \mathsf{Yu_2X}^{1}_{enc}} w = 0$
x^r_0 x^r_1 x^r_2 x^r_3
x^r_{12} x^r_{13} x^r_{14} x^r_{15}
$\mathcal{SL}^{-1}$
y^r_0 y^r_1 y^r_2 y^r_3
y^r_{12} y^r_{13} y^r_{14} y^r_{15}
k^r_0 k^r_1 k^r_2 k^r_3
k^r_{12} k^r_{13} k^r_{14} k^r_{15}
P_0 P_1 P_2 P_3
P_{12} P_{13} P_{14} P_{15}
Ciphertext
Plaintext

Fig. 1. High-order differential key recovery attack framework

higher-order differential distinguisher on n_z states, $1 \leq n_z \leq 16$. Then, for these n_z states, the zero-sum property holds:

$$\sum_{v \in \mathcal{V}+c} \mathsf{Yu_2X}_{dec}^{r_{HD}} = 0.$$

Now, we can use the zero-sum property of these n_z output words to construct a multivariate equation system related to the round keys. The system consists of 16 round keys (the master key when the number of rounds is 12), and the number of equations is n_z. We denote these equations as $F_i(k_0^r, \ldots, k_{16}^r)$ for $1 \leq i \leq n_z$. If $r_{KR} = 1$, the equations might be independent because the final round does not involve a linear transformation. The algebraic degree of the equations is related to the algebraic degree of the S-box in the encryption direction, which are 8, 5, 3, and 2, respectively. If $r_{KR} = 2$,the algebraic degree of the equations is 64.

2. **Solving the equation system using Gröbner Basis method to recover the key.** If $n_z = 16$, we can directly solve the equation system using Gröbner Basis method to obtain the key. If $n_z < 16$, we first guess the remaining $16 - n_z$ key words and substitute them into the equations to solve for the remaining keys.

Complexity Analysis. In Step 1, when constructing the multivariate equation system, we first use 2^m data to construct an r_{HD}-round higher-order differential distinguisher. This requires 2^m rounds of r_{HD}-round $\mathsf{Yu_2X}$ decryption operations. Similar to the calculations in [19], extending r_{KR} rounds and then constructing n_z equations regarding the round keys $\boldsymbol{k^r}$ requires $r_{KR} \cdot 2^m$ $\mathsf{Yu_2X}$ encryption operations per round. Assuming that the complexity of a single YuX encryption and decryption operation is the same, the time complexity for constructing the equation system is denoted as $T_c = 2^m \cdot \frac{r_{HD}}{r} + 2^m \cdot \frac{r_{KR}}{r} = 2^m$ r-round of $\mathsf{Yu_2X}$ decryption operations.

For Step 2, solving the equation system requires n_z equations and 16 variables. The time complexity is denoted as $T_s = T_{guess} \cdot T_{gb}$, where $T_{guess} = 2^{n \cdot (16 - n_z)}, n = \{8, 16\}$ is the complexity for guessing the key, and

$$T_{gb} = \mathcal{O}\left(\binom{n_z + d_{mac}}{n_z}^{\omega} + (16 - n_z) \cdot B^{\omega}\right),$$

is the time complexity for solving the equation system using the Gröbner Basis method, where d_{max} is Macaulay bound define as $d_{mac} := \sum_{i=0}^{n_z} (\mathcal{D}(F_i) - 1) + 1$, B is Bezout bound $B := \prod_{i=1}^{n_z} \mathcal{D}(F_i)$, ω is the linear algebra constant. The specific process for solving the equation system using Gröbner Basis can be found in the full version of the paper [Appdendix C]. Thus, the overall time complexity for the key recovery attack is:

$$T = T_c + T_s.$$

7-Round Key Recovery Attack on $\mathsf{Yu_2X}$-8. Since the upper bound on algebraic degree at the output positions $i \pmod{16} = 0$ or $1, 0 \leq i \leq 15$ for

7-round $\mathsf{Yu_2X\text{-}8}$ is 126, we can choose 2^{127} ciphertexts to construct the 7-round higher-order differential distinguisher, with zero-sum positions at $i \pmod{16} = 0$ or 1. Thus, $n_z = 8$. At this point, $r_{HD} = 7$. We extend the attack by one round for key recovery, setting $r_{KR} = 1$. A multivariate equation system can be constructed, with 8 equations, each having algebraic degree 3. Additionally, the variables in these 8 equations are independent of each other (because the $\mathcal{SL}$ operation consists of 4 independent S-boxes).

Therefore, the time complexity for constructing the equation system is $T_c = 2^{127}$ 8 rounds of $\mathsf{Yu_2X\text{-}8}$ decryption operations. The time complexity for solving the system of equations is given by $T_s = 2^{8 \cdot 8} \cdot 4 \cdot \max\left\{\binom{2+d_{mac}}{2}^{\omega}, 2 \cdot B^{\omega}\right\} \approx 2^{98.01}$ field operations. Thus, the overall key recovery time complexity is $T = T_c + T_s \approx 2^{127}$ 8 rounds of $\mathsf{Yu_2X\text{-}8}$ decryption operations, with a data complexity of 2^{127} ciphertexts.

11-Round Key Recovery Attack on $\mathsf{Yu_2X\text{-}16}$. For $\mathsf{Yu_2X\text{-}16}$, the maximum available data complexity is 2^{128}. Using 2^{96} chosen ciphertexts, we can construct a 10-round $\mathsf{Yu_2X\text{-}16}$ higher-order differential distinguisher, where all outputs exhibit the zero-sum property. At this stage, $n_z = 16$ and $r_{HD} = 10$. We aim to extend the attack by one additional round to construct a system of equations related to the round key $\boldsymbol{k^r}$. The constructed system consists of 16 equations, each with 16 variables, and the algebraic degree of each equation is $8, 5, 3, 2$.

Therefore, the time complexity to construct the equation system is 2^{96} for performing 11 rounds of $\mathsf{Yu_2X\text{-}16}$ decryption. Solving the system of equations has a time complexity $T_s = 4 \cdot \max\left\{\binom{4+d_{mac}}{4}^{\omega}, 4 \cdot B^{\omega}\right\} \approx 2^{37.76}$ field operations. Thus, the total time complexity for the key recovery attack is: $T = T_c + T_s \approx 2^{96}$ 11 rounds of $\mathsf{Yu_2X\text{-}16}$ decryption and the data complexity remains 2^{96} due to the chosen ciphertexts.

5 Conclusion

In this paper, we focused on analyzing the growth of the algebraic degree bound for the binary extension field version of YuX, introducing the concept of the exponent set. Based on this concept, we analyzed the algebraic degree growth of the S-box in the decryption direction of $\mathsf{Yu_2X}$ and proved that the algebraic degree bound of the round function in the decryption direction grows linearly. We also verified our proof using the general monomial prediction technique and experimental results. Building on this, we constructed a higher-order differential distinguisher for $\mathsf{Yu_2X}$ and utilized it in key recovery attacks. Under weak key settings, we found that the redundant secure rounds of $\mathsf{Yu_2X}$ are reduced to only one round, indicating that the security of $\mathsf{Yu_2X}$ requires further investigation. In future work, we will further analyze the $\mathsf{Yu}_p\mathsf{X}$ version using algebraic techniques, with the aim of providing a more comprehensive analysis of YuX.

References

1. Albrecht, M.R., Grassi, L., Rechberger, C., Roy, A., Tiessen, T.: MiMC: Efficient Encryption and Cryptographic Hashing with Minimal Multiplicative Complexity. In: ASIACRYPT(1). LNCS, vol. 10031, pp. 191–219 (2016)
2. Albrecht, M.R., Rechberger, C., Schneider, T., Tiessen, T., Zohner, M.: Ciphers for MPC and FHE. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015(1). LNCS, vol. 9056, pp. 430–454. Springer (2015)
3. Ashur, T., Mahzoun, M., Toprakhisar, D.: Chaghri - A FHE-friendly Block Cipher. In: CCS. pp. 139–150. ACM (2022)
4. Baudrin, J., Belaïd, S., Bon, N., Boura, C., Canteaut, A., Leurent, G., Paillier, P., Perrin, L., Rivain, M., Rotella, Y., Tap, S.: Transistor: a TFHE-friendly Stream Cipher. IACR Cryptol. ePrint Arch. p. 282 (2025)
5. Beyne, T., Canteaut, A., Dinur, I., Eichlseder, M., Leander, G., Leurent, G., Naya-Plasencia, M., Perrin, L., Sasaki, Y., Todo, Y., Wiemer, F.: Out of Oddity - New Cryptanalytic Techniques Against Symmetric Primitives Optimized for Integrity Proof Systems. In: CRYPTO(3). LNCS, vol. 12172, pp. 299–328. Springer (2020)
6. Bouvier, C., Canteaut, A., Perrin, L.: On the algebraic degree of iterated power functions. Des. Codes Cryptogr. **91**(3), 997–1033 (2023)
7. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) Fully Homomorphic Encryption without Bootstrapping. ACM Trans. Comput. Theory **6**(3), 13:1–13:36 (2014)
8. Canteaut, A., Carpov, S., Fontaine, C., Lepoint, T., Naya-Plasencia, M., Paillier, P., Sirdey, R.: Stream Ciphers: A Practical Solution for Efficient Homomorphic-Ciphertext Compression. In: FSE. LNCS, vol. 9783, pp. 313–333. Springer (2016)
9. Cheon, J.H., Kim, A., Kim, M., Song, Y.S.: Homomorphic Encryption for Arithmetic of Approximate Numbers. In: ASIACRYPT (1). LNCS, vol. 10624, pp. 409–437. Springer (2017)
10. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: TFHE: Fast Fully Homomorphic Encryption Over the Torus. J. Cryptol. **33**(1), 34–91 (2020)
11. Cho, J., Ha, J., Kim, S., Lee, B., Lee, J., Lee, J., Moon, D., Yoon, H.: Transciphering Framework for Approximate Homomorphic Encryption. In: ASIACRYPT (3). LNCS, vol. 13092, pp. 640–669. Springer (2021)
12. Cho, M., Chung, W., Ha, J., Lee, J., Oh, E., Son, M.: FRAST: TFHE-Friendly Cipher Based on Random S-Boxes. IACR Trans. Symmetric Cryptol. **2024**(3), 1–43 (2024)
13. Cid, C., Grassi, L., Gunsing, A., Lüftenegger, R., Rechberger, C., Schofnegger, M.: Influence of the Linear Layer on the Algebraic Degree in SP-Networks. IACR Trans. Symmetric Cryptol. **2022**(1), 110–137 (2022)
14. Cid, C., Indrøy, J.P., Raddum, H.: FASTA - A Stream Cipher for Fast FHE Evaluation. In: CT-RSA. LNCS, vol. 13161, pp. 451–483. Springer (2022)
15. Cosseron, O., Hoffmann, C., Méaux, P., Standaert, F.: Towards Case-Optimized Hybrid Homomorphic Encryption - Featuring the Elisabeth Stream Cipher. In: ASIACRYPT (3). LNCS, vol. 13793, pp. 32–67. Springer (2022)
16. Cui, J., Hu, K., Wang, M., Wei, P.: On the Field-Based Division Property: Applications to MiMC, Feistel MiMC and GMiMC. In: ASIACRYPT(3). LNCS, vol. 13793, pp. 241–270. Springer (2022)
17. Dobraunig, C., Eichlseder, M., Grassi, L., Lallemand, V., Leander, G., List, E., Mendel, F., Rechberger, C.: Rasta: A Cipher with Low ANDdepth and Few ANDs per Bit. In: CRYPTO (1). LNCS, vol. 10991, pp. 662–692. Springer (2018)

18. Dobraunig, C., Grassi, L., Helminger, L., Rechberger, C., Schofnegger, M., Walch, R.: Pasta: A Case for Hybrid Homomorphic Encryption. IACR Trans. Cryptogr. Hardw. Embed. Syst. **2023**(3), 30–73 (2023)
19. Eichlseder, M., Grassi, L., Lüftenegger, R., Øygarden, M., Rechberger, C., Schofnegger, M., Wang, Q.: An Algebraic Attack on Ciphers with Low-Degree Round Functions: Application to Full MiMC. In: ASIACRYPT(1). LNCS, vol. 12491, pp. 477–506. Springer (2020)
20. Fan, J., Vercauteren, F.: Somewhat Practical Fully Homomorphic Encryption. IACR Cryptol. ePrint Arch. p. 144 (2012)
21. Ha, J., Kim, S., Lee, B., Lee, J., Son, M.: Rubato: Noisy Ciphers for Approximate Homomorphic Encryption. In: EUROCRYPT (1). LNCS, vol. 13275, pp. 581–610. Springer (2022)
22. Hebborn, P., Leander, G.: Dasta - Alternative Linear Layer for Rasta. IACR Trans. Symmetric Cryptol. **2020**(3), 46–86 (2020)
23. Hu, K., Sun, S., Wang, M., Wang, Q.: An Algebraic Formulation of the Division Property: Revisiting Degree Evaluations, Cube Attacks, and Key-Independent Sums. In: ASIACRYPT(1). LNCS, vol. 12491, pp. 446–476. Springer (2020)
24. Knudsen, L.R.: Truncated and Higher Order Differentials. In: FSE. LNCS, vol. 1008, pp. 196–211. Springer (1994)
25. Lai, X.: Higher Order Derivatives and Differential Cryptanalysis, pp. 227–233. Springer US, Boston, MA (1994)
26. Liu, F., Li, Y., Chen, H., Jiao, L., Luo, M., Wang, M.: Yux: Finite field multiplication based block ciphers for efficient FHE evaluation. IEEE Trans. Inf. Theory **70**(5), 3729–3749 (2024)
27. Liu, F., Anand, R., Wang, L., Meier, W., Isobe, T.: Coefficient Grouping: Breaking Chaghri and More. In: EUROCRYPT (4). LNCS, vol. 14007, pp. 287–317. Springer (2023)
28. Liu, F., Grassi, L., Bouvier, C., Meier, W., Isobe, T.: Coefficient Grouping for Complex Affine Layers. In: CRYPTO(3). LNCS, vol. 14083, pp. 540–572. Springer (2023)
29. Liu, F., Kalam, A., Sarkar, S., Meier, W.: Algebraic Attack on FHE-Friendly Cipher HERA Using Multiple Collisions. IACR Trans. Symmetric Cryptol. **2024**(1), 214–233 (2024)
30. Liu, F., Sarkar, S., Meier, W., Isobe, T.: Algebraic Attacks on Rasta and Dasta Using Low-Degree Equations. In: ASIACRYPT (1). LNCS, vol. 13090, pp. 214–240. Springer (2021)
31. Méaux, P., Journault, A., Standaert, F., Carlet, C.: Towards Stream Ciphers for Efficient FHE with Low-Noise Ciphertexts. In: EUROCRYPT (1). LNCS, vol. 9665, pp. 311–343. Springer (2016)
32. Naehrig, M., Lauter, K.E., Vaikuntanathan, V.: Can homomorphic encryption be practical? In: CCSW. pp. 113–124. ACM (2011)

A General Monomial Prediction Technique

Definition 2 (General Monomial Trail [16]). *Let F_i be a sequence of polynomials over $\mathbb{F}_{2^n}$ for $0 \le i < r$, while $\boldsymbol{x}^{(i+1)} = F_i(\boldsymbol{x}^{(i)})$. A sequence of monomials $(\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}), \dots, \pi_{\boldsymbol{u}^{(i)}}(\boldsymbol{x}^{(i)}), \dots, \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)}))$ is an r-round general monomial trail connecting $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)})$ and $\pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$ with respect to the composition function $\boldsymbol{F} = F_{r-1} \circ F_{r-2} \circ \cdots \circ F_0$ if*

$$\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \rightarrow \cdots \rightarrow \pi_{\boldsymbol{u}^{(i)}}(\boldsymbol{x}^{(i)}) \rightarrow \cdots \rightarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)}).$$

If there is at least one general monomial trail connecting $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)})$ and $\pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$, we denote $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \rightsquigarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$. Otherwise, $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \not\rightsquigarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$. When $n = 1$, the general monomial trail is equivalent to the monomial trail.

Lemma 2 ([16]). *If $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \rightarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$, then $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \rightsquigarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$; or, equivalent $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \not\rightsquigarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$ implies $\pi_{\boldsymbol{u}^{(0)}}(\boldsymbol{x}^{(0)}) \not\rightarrow \pi_{\boldsymbol{u}^{(r)}}(\boldsymbol{x}^{(r)})$.*

Rule 1 (Field-based t-XOR). *Let $F_{t\text{-}XOR} : \mathbb{F}_{2^n}^t \rightarrow \mathbb{F}_{2^n}, t \geq 3$ be a function that consists of $t-1$ XOR, where the input $\boldsymbol{x} = (x_0, x_1, \ldots, x_{t-1})$ takes values from $\mathbb{F}_{2^n} \times \mathbb{F}_{2^n} \times \cdots \times \mathbb{F}_{2^n}$ and the output y is calculated as $y = x_0 \oplus x_1 \oplus x_2 \oplus \cdots \oplus x_{t-1}$. If the monomial y^v contains $\boldsymbol{x}^{\boldsymbol{u}}$, then*

$$v = u_0 + u_1 + \cdots + u_{t-1} \quad \textit{and} \quad v \succeq u_i, i \in \{0, 1, \ldots, t-1\},$$

where $0 \leq v, u_0, \ldots, u_{t-1} \leq 2^n - 1$.

Rule 2 (Field-based AND [16]). *Let $F_{AND} : \mathbb{F}_{2^n}^2 \rightarrow \mathbb{F}_{2^n}$ be a function that consists of an AND, where the input $\boldsymbol{x} = (x_0, x_1)$ takes values from $\mathbb{F}_{2^n} \times \mathbb{F}_{2^n}$ and the output y is calculated as $y = x_0 \cdot x_1$. If the monomial y^v contains $\boldsymbol{x}^{\boldsymbol{u}}$, then*

$$v = u_0 = u_1 = \cdots = u_{t-1},$$

where $0 \leq v, u_0, u_1 \leq 2^n - 1$.

Rule 3 (Field-based t-COPY [16]). *Let $F_{t-COPY} : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n}^t$ be a t-COPY function, where the input x takes values from $\mathbb{F}_{2^n}$ and the output $\boldsymbol{y} = (y_0, \ldots, y_{t-1}) \in \mathbb{F}_{2^n} \times \mathbb{F}_{2^n} \times \cdots \times \mathbb{F}_{2^n}$ is calculated as $\boldsymbol{y} = (\underbrace{x, x, \ldots, x}_{t})$. If the monomial $\boldsymbol{y}^{\boldsymbol{v}}$ contains x^u, then*

$$\boldsymbol{v} = (v_0, v_1, \ldots, v_{t-1}) = \begin{cases} (0, 0, \ldots, 0), & \textit{if } u = 0; \\ (i_0^s, i_1^s, \ldots, i_{t-1}^s) \textit{ for } 0 \leq s \leq t-1, & \textit{else}. \end{cases}$$

Here, $i_{t-1}^s = u + (s-1)(2^n - 1) - \sum_{j=0}^{t-2} i_j^s$, $0 \leq i_j^s \leq 2^n - 1$ for $0 \leq j < t$, where $u, v_0, v_1, \ldots, v_{t-1} \in [0, 2^n - 1]$.

B The Algebraic Degree Upper Bounds of Yu$_2$X in Different Variables Setting

C Gröbner Basis Attacks

The attacks discussed involve solving a system of multivariate polynomial equations over a finite prime field $\mathbb{F}_p$. The notation used is as follows: Let $\{F_1, \ldots, F_{ne}\} \subset \mathbb{F}_p[x_1, \ldots, x_{nv}]$ represent a set of ne polynomials in nv variables. Let $I := \langle F_1, \ldots, F_{ne} \rangle$ be the ideal generated by these polynomials, and let d_I be the degree of the ideal, defined as $d_I := \dim_{\overline{\mathbb{F}_p}}(\mathbb{F}_p[x_1, \ldots, x_{nv}]/I)$, which represents the dimension of the quotient ring $\mathbb{F}_p[x_1, \ldots, x_{nv}]/I$ as an $\overline{\mathbb{F}_p}$-vector space.

Table 6. The algebraic degree upper bounds of Yu$_2$X-8 in multivariate setting

Cipher	**In**	**Out**	**Rounds**								
			0	1	2	3	4	5	6	7	8
Yu$_2$X-8	(0,4,8)	0/1	1	1	2	6	12	18	**24**	-	-
		2	1	1	3	8	14	20	**24**	-	-
		3	1	1	3	9	15	21	**24**	-	-
	(0,4,8,12)	0/1	1	1	2	6	14	22	30	**32**	-
		2	1	1	3	9	17	25	**32**	-	-
		3	1	1	3	11	18	26	**32**	-	-
	(0,1,4,8,12)	0	1	1	2	8	16	26	36	**40**	-
		1	1	1	3	8	17	27	37	**40**	-
		2	1	1	4	11	20	30	**40**	-	-
		3	1	2	4	12	21	31	**40**	-	-
	(0,1,3,4,8,12)	0	1	1	3	8	19	31	43	**48**	-
		1	1	1	3	8	19	31	43	**48**	-
		2	1	2	4	11	22	34	46	**48**	-
		3	1	2	5	12	23	35	47	**48**	-
	(0,1,3,4,6,8,12)	0	1	1	3	10	21	35	49	**56**	-
		1	1	1	4	10	21	35	49	**56**	-
		2	1	2	5	13	25	39	53	**56**	-
		3	1	2	5	13	26	40	54	**56**	-
	(0,1,4,5,8,9,12,13)	0	1	1	4	12	21	37	53	**64**	-
		1	1	1	4	12	21	37	53	**64**	-
		2	1	1	6	16	32	48	**64**	-	-
		3	1	2	6	17	33	49	**64**	-	-
	(0,1,3,4,5,8,9,12,13)	0	1	1	4	12	22	40	58	**72**	-
		1	1	1	4	12	22	40	58	**72**	-
		2	1	2	6	18	36	54	**72**	-	-
		3	1	2	6	18	36	54	**72**	-	-
	(0,1,3,4,5,8,9,12,13,15)	0	1	1	4	12	23	43	63	**80**	-
		1	1	1	4	12	23	43	63	**80**	-
		2	1	2	6	18	38	58	78	**80**	-
		3	1	2	6	18	38	58	78	**80**	-
	(0,1,3,4,5,7,8,9,12,13,15)	0	1	1	4	12	24	46	68	**88**	-
		1	1	1	4	12	24	46	68	**88**	-
		2	1	2	6	18	30	52	74	**88**	-
		3	1	2	6	18	30	52	74	**88**	-
	(0,1,3,4,5,7,8,9,11,12,13,15)	0	1	1	4	12	26	50	74	**96**	-
		1	1	1	4	12	26	50	74	**96**	-
		2	1	2	6	19	43	67	91	**96**	-
		3	1	2	6	19	43	67	91	**96**	-
	(0,1,2,3,4,5,7,8,9,11,12,13,15)	0	1	2	5	14	28	54	80	**104**	-
		1	1	2	5	14	28	54	80	**104**	-
		2	1	3	7	21	47	73	99	**104**	-
		3	1	3	8	22	48	74	100	**104**	-
	(0,1,2,3,4,5,6,7,8,9,11,12,13,15)	0	1	2	5	16	30	58	86	**112**	-
		1	1	2	5	16	30	58	86	**112**	-
		2	1	3	8	24	52	80	108	**112**	-
		3	1	3	8	24	52	80	108	**112**	-
	(0,1,2,3,4,5,6,7,8,9,10,11,12,13,15)	0	1	2	6	17	36	66	96	126	**128**
		1	1	2	6	17	36	66	96	128	**128**
		2	1	2	9	26	56	86	116	**128**	-
		3	1	2	9	26	56	86	116	**128**	-

Table 7. The algebraic degree upper bounds of Yu$_2$X-16 in multivariate setting

Cipher	Input index	Output index	Rounds											
			0	1	2	3	4	5	6	7	8	9	10	11
Yu$_2$X-16	(0,4,8)	0/1	1	1	2	6	12	18	24	30	36	42	**48**	-
		2	1	1	3	8	14	20	26	32	38	44	**48**	-
		3	1	1	3	9	15	21	27	33	39	45	**48**	-
	(0,4,8,12)	0/1	1	1	2	6	14	22	30	38	46	54	62	**64**
		2	1	1	3	9	17	25	33	41	49	57	**64**	-
		3	1	1	3	11	18	26	34	42	50	58	**64**	-
	(0,1,4,8,12)	0	1	1	2	8	16	26	36	46	56	66	76	**80**
		1	1	1	3	8	17	27	37	47	57	67	77	**80**
		2	1	1	4	11	20	30	40	50	60	70	**80**	-
		3	1	2	4	12	21	31	41	51	61	71	**80**	-
	(0,1,3,4,8,12)	0	1	1	3	8	19	31	43	55	67	79	91	**96**
		1	1	1	3	8	19	31	43	55	67	79	91	**96**
		2	1	2	4	11	22	34	46	58	70	82	94	**96**
		3	1	2	5	12	23	35	47	59	71	83	95	**96**
	(0,1,3,4,6,8,12)	0	1	1	3	10	21	35	49	63	77	91	105	**112**
		1	1	1	4	10	21	35	49	63	77	91	105	**112**
		2	1	2	5	13	25	39	53	67	81	95	109	**112**
		3	1	2	5	13	26	40	54	68	82	96	110	**112**

Multivariate System Solving. A well-established method for solving the system of equations $F_1 = 0, \ldots, F_{ne} = 0$ in nv variables is to first compute the Gröbner basis of the ideal $I = \langle f_1, \ldots, f_{ne} \rangle$ with respect to the lexicographic order. This results in a triangular form, which reduces the multivariate system solving problem to a series of univariate root finding problems. These solutions can then be recovered and extended iteratively by solving one or more univariate equations. While computing the desired Gröbner basis directly can be computationally expensive, an efficient method is available when the ideal I is zero-dimensional, i.e., $d_I < \infty$. The following approach is commonly used:

1. First, compute the Gröbner basis of I using the degree-reverse-lexicographic (Drl) order, applying algorithms such as Faugère's F4 or F5.
2. Then, use the FGLM basis conversion algorithm to transform the drl Gröbner basis into one with respect to the lexicographic (Lex) order.

Complexity Analysis. Assuming the system is well-defined, i.e., $ne = nv$, and the ideal I is zero-dimensional, the following complexity estimates are applicable:

1. **Computing the Drl Gröbner Basis:** Using F4 or F5 algorithms, the drl Gröbner basis can be computed in $\mathcal{O}\left(\binom{nv+d_{reg}}{nv}^{\omega}\right)$ operations over $\mathbb{F}_p$, where ω is the linear algebra constant and d_{reg} is the degree of regularity. Calculating d_{reg} is as difficult as computing the drl Gröbner basis, so tight bounds for d_{reg} are crucial for precise complexity estimation. For regular systems, the degree of regularity can be bounded by the Macaulay bound, defined as $d_{mac} := 1 + \sum_{i=1}^{nv} (\mathcal{D}^u(F_i) - 1)$.

2. **Basis Conversion:** Converting the Gröbner basis from Drl to lex order using the FGLM algorithm has a complexity of $\mathcal{O}(nv \cdot d_I^3)$ operations over $\mathbb{F}_p$. In certain cases, probabilistic algorithms can achieve a sub-cubic complexity of $\mathcal{O}(nv \cdot d_I^\omega)$, where ω is the linear algebra constant.
   The Bézout bound provides an upper limit for the degree d_I of the ideal I, which corresponds to the number of solutions (counted with multiplicities) to the polynomial system. This bound is given by: $B := \prod_{i=1}^{nv} \mathcal{D}^u(F_i)$. If the system is regular, then $d_I = B$.
3. **Univariate Polynomial Root Finding:** Given a univariate polynomial $F \in \mathbb{F}_p[X]$ of degree d, its roots canbe computed in $\mathcal{O}(d \log(d) \cdot (\log(d) + \log(p)) \cdot \log(\log(d)))$ finite field operations. Typically, if F is the unique univariate polynomial in the Lex Gröbner basis, we have $\mathcal{D}(F) \leq d_I$. Thus, in most cases, finding the solutions from a lex Gröbner basis is not computationally expensive compared to the previous steps, especially when the system is not already in lexicographic order.

Estimating Time Complexity. The overall time complexity for computing the set of common solutions for a system of nv polynomials in nv variables generating a zero-dimensional ideal I via Gröbner basis computation is:

$$\mathcal{O}\left(\binom{nv + d_{mac}}{nv}^{\omega} + nv \cdot B^{\omega}\right).$$