The Inverse of χ and Its Applications to Rasta-like Ciphers

Fukang Liu[1], Santanu Sarkar[4], Willi Meier[5], Takanori Isobe[1,2,3]

[1] University of Hyogo, Hyogo, Japan
liufukangs@gmail.com,takanori.isobe@ai.u-hyogo.ac.jp
[2] National Institute of Information and Communications Technology, Tokyo, Japan
[3] PRESTO, Japan Science and Technology Agency, Tokyo, Japan
[4] Indian Institute of Technology Madras, Chennai, India
santanu@iitm.ac.in
[5] FHNW, Windisch, Switzerland
willimeier48@gmail.com

Abstract. At ASIACRYPT 2021, Liu et al. pointed out a weakness of the Rasta-like ciphers neglected by the designers. The main strategy is to construct exploitable equations of the n-bit χ operation denoted by χ_n. However, these equations are all obtained by first studying χ_n for small n. In this note, we demonstrate that if the explicit formula of the inverse of χ_n denoted by χ_n^{-1} is known, all these exploitable equations would have been quite obvious and the weakness of the Rasta-like ciphers could have been avoided at the design phase. However, the explicit formula of χ_n^{-1} seems to be not well-known and the most relevant work was published by Biryukov et al. at ASIACRYPT 2014. In this work, we give a very simple formula of χ_n^{-1} that can be written down in only one line and we prove its correctness in a rigorous way. Based on its formula, the formula of exploitable equations for Rasta-like ciphers can be easily derived and therefore more exploitable equations are found.

Keywords: Rasta · the inverse of χ · affine variety · algebraic attack

1 Preliminaries

Definition 1. [3] *Let $\mathbb{K}$ be a field, and let $l_1, l_2, \ldots, l_s$ be polynomials in $\mathbb{K}[v_1, v_2, \ldots, v_m]$. Then we set*

$$V(l_1, l_2, \ldots, l_s) = \{(a_1, a_2, \ldots, a_m) \in \mathbb{K}^m \mid l_i(a_1, a_2, \ldots, a_n) = 0 \ \forall i \in [1, m]\}.$$

We call $V(l_1, l_2, \ldots, l_s)$ the ***affine variety*** *defined by $l_1, l_2, \ldots, l_s$.*

From this definition, the affine variety $V(l_1, l_2, \ldots, l_s) \subseteq \mathbb{K}^m$ is the set of all solutions of the system of equations

$$l_1(a_1, a_2, \ldots, a_m) = l_2(a_1, a_2, \ldots, a_m) = \cdots = l_s(a_1, a_2, \ldots, a_m) = 0.$$

Throughout this paper, we consider the field $\mathbb{F}_2$, i.e. $\mathbb{K} = \mathbb{F}_2$.

The n-bit χ operation. The n-bit χ operation denoted by $\chi_n : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is defined as follows:

$$y_i = x_i + \overline{x_{i+1}} x_{i+2} \text{ for } i \in [0, n-1],$$

where $X = (x_0, x_1, \ldots, x_{n-1})$ and $Y = (y_0, y_1, \ldots, y_{n-1})$ denote the n-bit input and output, respectively. Moreover, the indices are considered within modulo n. To ensure χ_n is invertible, n has to be an odd number. For convenience, let

$$h = (n-1)/2.$$

Consider the ideal $\mathcal{G} = \langle g_0, g_1, \ldots, g_{n-1} \rangle$ where g_i is defined as follows:

$$g_i = y_i + x_i + \overline{x_{i+1}} x_{i+2}.$$

For convenience, the affine variety defined by $g_0, g_1, \ldots, g_{n-1}$ is denoted by $V(\mathcal{G})$. Obviously, $V(\mathcal{G})$ represents the mapping table of χ_n.

Finding the inverse of χ_n denoted by χ_n^{-1} is equivalent to finding another ideal $\mathcal{G}' = \langle g'_0, g'_1, \ldots, g'_{n-1} \rangle$ such that $V(\mathcal{G}') = V(\mathcal{G})$ and g'_i is of the following form

$$g'_i = x_i + P_i,$$

where P_i is a polynomial in $\mathbb{F}_2[y_0, y_1, \ldots, y_{n-1}]$.

As far as we know, the formula of χ_n^{-1} is not explicitly given in the literature. However, algorithmic procedures to efficiently compute χ_n^{-1} for any value of Y have been given in Daemen's thesis [4] and Biryukov et al.'s work [2] at ASIACRYPT 2014, respectively.

1.1 On Daemen's Method to Compute χ_n^{-1}

In Daemen's thesis, the method to compute χ_n^{-1} is called **seed-and-leap**. The procedure takes an arbitrary value Y as input and outputs X. For convenience, 0^n denotes $\underbrace{(0, 0, \ldots, 0)}_{n\ 0}$ and 1^n denotes $\underbrace{(1, 1, \ldots, 1)}_{n\ 1}$. When $Y = 0^n$, simply output $X = 0^n$. When $Y \neq 0^n$, X is computed in a sequential manner as described below:

1. **Seed.** Find an index j such that $y_{j+1} = 1$. Then, $x_j = y_j$.
2. **Leap.** If x_j is known, x_{j-2} can be found. Since n is an odd number, all x_i for $i \in [0, n-1]$ can be found by repeating this step.

We now show that the above procedure to compute χ_n^{-1} is directly derived from the definition of χ_n. Specifically, since

$$\begin{aligned} y_{j-2} &= x_{j-2} + \overline{x_{j-1}} x_j, \\ y_{j-1} &= x_{j-1} + \overline{x_j} x_{j+1}, \\ y_j &= x_j + \overline{x_{j+1}} x_{j+2}, \\ y_{j+1} &= x_{j+1} + \overline{x_{j+2}} x_{j+3}, \end{aligned}$$

when $y_{j+1} = 1$, we have $\overline{x_{j+1}} = \overline{x_{j+2}} x_{j+3}$, thus resulting $\overline{x_{j+1}} x_{j+2} = 0$ and $x_j = y_j$, i.e. x_j is known. Whatever x_j is, either $\overline{x_{j-1}} x_j$ or $\overline{x_j} x_{j+1}$ will be 0, thus resulting either x_{j-1} or x_{j-2} can be uniquely computed. If it is x_{j-1} that can be computed, i.e. $x_j = 1$, we can then also compute x_{j-2} since (x_j, x_{j-1}) are known. In other words, after x_j is determined, x_{j-2} can always be uniquely determined. One may notice that there may exist two ways to determine some x_i because we may leap back to these x_i and wonder whether contradictions will occur. This can be easily checked and no contradictions will occur. In other words, the above procedure will always output a valid $X \neq 0^n$ for any $Y \neq 0^n$.

Since an algorithmic procedure to compute χ_n^{-1} is given, the invertibility of χ_n is proved, which is how Daemen proved the invertibility of χ_n. It is now clear that the invertibility is not proved by giving a general formula of χ_n^{-1} and that deducing this general formula from the above **seed-and-leap** procedure is as difficult as deducing it from the definition of χ_n.

1.2 On Biryukov et al.'s Method to Compute χ_n^{-1}

The algorithm to compute χ_n^{-1} is placed in Appendix D[1] of [1] and no specific proof for its correctness is given. In addition, they also gave the explicit expression of x_3 for χ_9^{-1} which has a nice structure, as shown below:

$$x_3 = y_3 + (y_5 + (y_7(y_0 + y_2\overline{y_1})\overline{y_8})\overline{y_6})\overline{y_4}. \tag{1}$$

As far as we can understand, the algorithm described in Appendix D of [1] is unclear and there seem to be typos. Consequently, we will interpret it with our own description. Specifically, the original algorithm to compute χ_n^{-1} in [1] is shown in Algorithm 1, while our new interpretation is shown in Algorithm 2.

Algorithm 1 Given $(y_0, y_1, \ldots, y_{n-1})$, find $\chi_n^{-1}(y_0, y_1, \ldots, y_{n-1})$ [1]

$(x_0, x_1, \ldots, x_{n-1}) \leftarrow (y_0, y_1, \ldots, y_{n-1})$
for $0 \le i < \frac{3(n-1)}{2}$ **do**
  $-x_{(n-2)i} \leftarrow x_{(n-2)i} + y_{(n-2)i+2} \cdot y_{(n-2)i+1}$
return $(x_0, x_1, \ldots, x_{n-1})$

Algorithm 2 Given $(y_0, y_1, \ldots, y_{n-1})$, find $\chi_n^{-1}(y_0, y_1, \ldots, y_{n-1})$ [Our interpretation]

$(x_0, x_1, \ldots, x_{n-1}) \leftarrow (y_0, y_1, \ldots, y_{n-1})$
for $0 \le i < \frac{3(n-1)}{2}$ **do**
  $x_{(n-2)i} \leftarrow x_{(n-2)i} + x_{(n-2)i+2} \cdot \overline{x_{(n-2)i+1}}$
return $(x_0, x_1, \ldots, x_{n-1})$

We show that with Algorithm 2, the expression of x_3 for χ_9^{-1} can be simply derived, as shown below:

$$\begin{aligned}
i = 0: &\quad x_0 = y_0 + y_2\overline{y_1},\\
i = 1: &\quad x_7 = y_7 + x_0\overline{y_8},\\
i = 2: &\quad x_5 = y_5 + x_7\overline{y_6},\\
i = 3: &\quad x_3 = y_3 + x_5\overline{y_4}.
\end{aligned}$$

Hence, we have

$$x_3 = y_3 + (y_5 + (y_7 + (y_0 + y_2\overline{y_1})\overline{y_8})\overline{y_6})\overline{y_4}.$$

As a result, we believe our interpretation is clearer and what the authors of [1] wanted to express should be Algorithm 2.

Again, we take χ_9^{-1} for example to see how the algorithm ends. Let us continue the above procedure, as shown below:

$$\begin{aligned}
i = 4: &\quad x_1 = y_1 + x_3\overline{y_2},\\
i = 5: &\quad x_8 = y_8 + x_1\overline{x_0},\\
i = 6: &\quad x_6 = y_6 + x_8\overline{x_7},\\
i = 7: &\quad x_4 = y_4 + x_6\overline{x_5},\\
i = 8: &\quad x_2 = y_2 + x_4\overline{x_3},\\
i = 9: &\quad x_0 = y_0 + x_2\overline{x_1},
\end{aligned}$$

[1] The eprint version.

$$
\begin{aligned}
i = 10: &\qquad x_7 = y_7 + x_0\overline{x_8},\\
i = 11: &\qquad x_5 = y_5 + x_7\overline{x_6}.
\end{aligned}
$$

In this way, it is possible to deduce the expressions of x_i for all $i \in [0, 8]$ in terms of $(y_0, y_1, \ldots, y_8)$ and they are found in the order:

$$x_3 \to x_1 \to x_8 \to x_6 \to \cdots \to x_7 \to x_5.$$

More generally, with Algorithm 2, the expressions of x_i for χ_n^{-1} for all $i \in [0, n-1]$ can be found in the order:

$$x_{h-1} \to x_{h-3} \to \cdots \to x_{h+3} \to x_{h+1}.$$

How to find Algorithm 2 and prove its correctness? In [1], only this algorithm and the expression of x_3 for χ_9^{-1} are given, while how this algorithm is obtained and how to prove its correctness are missing. Different from Daemen's seed-and-leap method whose correctness can be easily verified, it is not intuitive to prove the correctness of Algorithm 2.

The following is our understanding. Specifically, let us slightly explain why the expression of x_3 for χ_9^{-1} is correct. Based on the definition of χ_9, there are

$$
\begin{aligned}
x_3 + y_3 &= \overline{x_4}x_5 = (\overline{y_4} + \overline{x_5}x_6)x_5 = \overline{y_4}x_5,\\
x_5 + y_5 &= \overline{x_6}x_7 = (\overline{y_6} + \overline{x_7}x_8)x_7 = \overline{y_6}x_7,\\
x_7 + y_7 &= \overline{x_8}x_0 = (\overline{y_8} + \overline{x_0}x_1)x_0 = \overline{y_8}x_0,\\
x_0 + y_0 &= \overline{x_1}x_2 = (\overline{y_1} + \overline{x_2}x_3)x_2 = \overline{y_1}x_2,\\
x_2 + y_2 &= \overline{x_3}x_4 = (\overline{y_3} + \overline{x_4}x_5)x_4 = \overline{y_3}x_4.
\end{aligned}
$$

Therefore, we have

$$x_3 = y_3 + (y_5 + (y_7 + (y_0 + (y_2 + \overline{y_3}x_4)\overline{y_1})\overline{y_8})\overline{y_6})\overline{y_4}.$$

Based on Algorithm 2, we indeed have

$$x_3 = y_3 + (y_5 + (y_7 + (y_0 + y_2\overline{y_1})\overline{y_8})\overline{y_6})\overline{y_4}.$$

Hence, it is necessary to prove $x_4\overline{y_3}\ \overline{y_1}\ \overline{y_8}\ \overline{y_6}\ \overline{y_4} = 0$ always holds. It is easy to observe that the above procedure can also be generalized for χ_n^{-1} of any valid n. We leave this observation here, and it can be found later that we will prove the same problem for our formula of χ_n^{-1}.

1.3 Motivation to Study χ_n^{-1}

For the stream cipher Rasta [5], the trivial algebraic attack is to solve a system of equations of degree 2^R where R denotes the number of rounds. However, it has been shown in [6] that the last nonlinear layer can almost be peeled off by finding exploitable equations in terms of (X, Y) of the following form:

$$P(Y) + \sum_{j=0}^{n-1} x_j L_j(Y) + c = 0,$$

where $c \in \mathbb{F}_2$ is a constant, $P(Y) \in \mathbb{F}_2[y_0, y_1, \ldots, y_{n-1}]$ with $Deg(P) \leq 2^{R-1} + 1$, and $L_j(Y) \in \mathbb{F}_2[y_0, y_1, \ldots, y_{n-1}]$ with $Deg(L_j) \leq 1$. In this way, the algebraic attack is reduced to solving a system of equations of degree $2^{R-1} + 1$ because the degree of the expressions of X and Y in terms of the key bits is upper bounded by 2^{R-1} and 1, respectively. The data complexity of this algebraic attack is related to the number of exploitable equations, the length of the key and the degree of the constructed equations. Increasing the number of exploitable equations by a factor of q can reduce the data complexity by a factor of q.

2 Main Results

It has been observed in [6] that the found exploitable equations belong to the ideal $\mathcal{F} = \langle f_0, f_1, \ldots, f_{n-1}\rangle$ where

$$f_i = x_i + y_i + \overline{y_{i+1}} x_{i+2} \text{ for } i \in [0, n-1].$$

In the following, we will study the affine variety defined by $f_0, f_1, \ldots, f_{n-1}$ denoted by $V(\mathcal{F})$.

2.1 The Formula of χ_n^{-1}

Lemma 1. *$V(\mathcal{G})$ and $V(\mathcal{F})$ satisfy $V(\mathcal{G}) = V(\mathcal{F})\backslash\{(1^n, 0^n)\}$.*

Proof. First, we prove $V(\mathcal{G}) \subseteq V(\mathcal{F})$. For any $(X, Y) \in V(\mathcal{G})$, we have

$$\begin{aligned} y_i &= x_i + \overline{x_{i+1}} x_{i+2}, \\ y_{i+1} &= x_{i+1} + \overline{x_{i+2}} x_{i+3}, \end{aligned}$$

Hence,

$$f_i = x_i + y_i + \overline{y_{i+1}} x_{i+2} = \overline{x_{i+1}} x_{i+2} + \overline{x_{i+1}} x_{i+2} = 0,$$

which implies $V(\mathcal{G}) \subseteq V(\mathcal{F})$. As the point $(X, Y) = (1^n, 0^n)$ does not satisfy $g_i = 0$ for $i \in [0, n-1]$, $V(\mathcal{G}) \subseteq V(\mathcal{F})\backslash\{(1^n, 0^n)\}$.

Next, we prove $V(\mathcal{F})\backslash\{(1^n, 0^n)\} \subseteq V(\mathcal{G})$. For any $(X, Y) \in V(\mathcal{F})\backslash\{(1^n, 0^n)\}$, we have

$$\begin{aligned} y_i &= x_i + \overline{y_{i+1}} x_{i+2}, \\ y_{i+1} &= x_{i+1} + \overline{y_{i+2}} x_{i+3}, \end{aligned}$$

Hence,

$$g_i = y_i + x_i + \overline{x_{i+1}} x_{i+2} = x_{i+2}(x_{i+1} + y_{i+1}).$$

As $x_i + y_i = x_{i+2}\overline{y_{i+1}}$, we have

$$\begin{aligned} g_i &= x_{i+2} x_{i+3} \overline{y_{i+2}} \\ &= x_{i+2} x_{i+3} x_{i+4} \overline{y_{i+3}} \\ &= \ldots = x_{i+2} x_{i+3} \ldots x_{i+k} \overline{y_{i+k-1}} \\ &= \ldots = x_{i+2} x_{i+3} \ldots x_i \overline{y_{i-1}} \\ &= x_{i+2} x_{i+3} \ldots x_i x_{i+1} \overline{y_i} \\ &= x_{i+2} x_{i+3} \ldots x_{i+1} x_{i+2} \overline{y_{i+1}}, \end{aligned}$$

which implies $g_i = 0$ always holds when $Y \neq 0^n$. Thus, we are left to prove $V(\mathcal{F})\backslash\{(1^n, 0^n)\} \subseteq V(\mathcal{G})$ for $Y = 0^n$.

When $Y = 0^n$, we immediately obtain a system of linear equations in terms of $(x_0, x_1, \ldots, x_{n-1})$, as shown below:

$$0 = x_i + x_{i+2} \text{ for } i \in [0, n-1].$$

There are only 2 solutions to this equation system, which are $X = 0^n$ and $X = 1^n$. When $X = 0^n$, $g_i = 0$ for $i \in [0, n-1]$. When $X = 1^n$, we obtain the point $(X, Y) = (1^n, 0^n)$, thus proving $V(\mathcal{F})\backslash\{(1^n, 0^n)\} \subseteq V(\mathcal{G})$. In other words, $V(\mathcal{G}) = V(\mathcal{F})\backslash\{(1^n, 0^n)\}$ is proved. □

Theorem 1. *The expression of* χ_n^{-1} *is*

$$x_i = y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}. \tag{2}$$

Proof. Let

$$w_i = x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}.$$

Denote the affine variety defined by $w_0, w_1, \ldots, w_{n-1}$ by $V(\mathcal{W})$. If we can prove $V(\mathcal{W}) = V(\mathcal{G})$, Theorem 1 is proved.

First, we prove $V(\mathcal{W}) \subseteq V(\mathcal{G}) = V(\mathcal{F}) \setminus \{(1^n, 0^n)\}$. For any $(X, Y) \in V(\mathcal{W})$, there are

$$\begin{aligned}
x_i &= y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}, \\
x_{i+2} &= y_{i+2} + \sum_{j=1}^{h} y_{i-2(j-1)+1} \prod_{k=j}^{h} \overline{y_{i-2(k-1)}} = y_{i+2} + \sum_{j=0}^{h-1} y_{i-2j+1} \prod_{k=j}^{h-1} \overline{y_{i-2k}}.
\end{aligned}$$

Since

$$\begin{aligned}
x_{i+2}\overline{y_{i+1}} &= y_{i+2}\overline{y_{i+1}} + \overline{y_{i+1}} \sum_{j=0}^{h-1} y_{i-2j+1} \prod_{k=j}^{h-1} \overline{y_{i-2k}} \\
&= y_{i-2h+1}\overline{y_{i-2h}} + \overline{y_{i-2h}} \sum_{j=0}^{h-1} y_{i-2j+1} \prod_{k=j}^{h-1} \overline{y_{i-2k}} \\
&= \sum_{j=0}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\
&= y_{i+1} \prod_{k=0}^{h} \overline{y_{i-2k}} + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\
&= \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \Leftarrow (y_{i-2h} = y_{i+1}),
\end{aligned}$$

we have

$$x_{i+2}\overline{y_{i+1}} = x_i + y_i.$$

Hence, $V(\mathcal{W}) \subseteq V(\mathcal{F})$. As the point $(X, Y) = (1^n, 0^n) \notin V(\mathcal{W})$, we have $V(\mathcal{W}) \subseteq V(\mathcal{F}) \setminus \{(1^n, 0^n)\} = V(\mathcal{G})$.

Next, we prove $V(\mathcal{G}) \subseteq V(\mathcal{W})$. For any $(X, Y) \in V(\mathcal{G})$, there is

$$y_i = x_i + \overline{x_{i+1}} x_{i+2}.$$

To prove

$$x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} = 0, \tag{3}$$

we first study

$$\begin{aligned}
& y_{i+1}(x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}) \\
= \; & (\overline{x_{i+1}} x_{i+2})(x_{i+1} + \overline{x_{i+2}} x_{i+3}) + y_{i+1} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\
= \; & y_{i+1} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}.
\end{aligned}$$

Since $i - 2h = i + 1 \mod n$, $\overline{y_{i+1}}$ is a factor of $\prod_{k=j}^{h} \overline{y_{i-2k}}$. In other words,

$$y_{i+1}(x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}) = y_{i+1} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} = 0$$

holds for any $(X, Y) \in V(\mathcal{G})$. Therefore, for any $i \in [0, n-1]$ and $(X, Y) \in V(\mathcal{G})$, when $y_{i+1} = 1$, Equation 3 always holds. Thus, we are left to prove Equation 3 for $y_{i+1} = 0$.

We now prove by induction that if Equation 3 holds for any $(X, Y) \in V(\mathcal{G})$ with $(y_{i+1}, y_{i+3}, \ldots, y_{i+2t+1}) \neq (0, 0, \ldots, 0)$ where $t \in [0, h-1]$, Equation 3 also holds for any $(X, Y) \in V(\mathcal{G})$ with $(y_{i+1}, y_{i+3}, \ldots, y_{i+2t+1}, y_{i+2(t+1)+1}) \neq (0, 0, \ldots, 0)$.

We have proved above that Equation 3 holds for $y_{i+1} \neq 0$. Assuming Equation 3 holds for the case $t = b$, we now prove that it also holds for $t = b + 1$. In other words, we now prove Equation 3 for $y_{i+2(b+1)+1} = 1$ and $(y_{i+1}, y_{i+3}, \ldots, y_{i+2b+1}) = (0, 0, \ldots, 0)$. In this case, Equation 3 can be rewritten as

$$\begin{aligned}
& x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\
= \; & x_i + y_i + \sum_{j=h-b}^{h} y_{i-2j+1} + \sum_{j=1}^{h-(b+1)} y_{i-2j+1} \prod_{k=j}^{h-(b+1)} \overline{y_{i-2k}} \\
= \; & x_i + y_i + \sum_{j=h-b}^{h} y_{i-2j+1}
\end{aligned}$$

due to $(y_{i-2h}, y_{i-2(h-1)}, \ldots, y_{i-2(h-b)}) = (y_{i+1}, y_{i+3}, \ldots, y_{i+2b+1}) = (0, 0, \ldots, 0)$ and $\overline{y_{i-2(h-(b+1))}} = \overline{y_{i+2(b+1)+1}} = 0$.

For any $(X, Y) \in V(\mathcal{G})$ with $(y_{i+1}, y_{i+3}, \ldots, y_{i+2b+1}) = (0, 0, \ldots, 0)$, we also have

$$x_{i+2d} + y_{i+2d} \;=\; x_{i+2d+2} \overline{y_{i+2d+1}} = x_{i+2d+2}$$

for $d \in [0, b]$ due to $V(\mathcal{G}) \subseteq V(\mathcal{F})$.

Therefore,

$$\begin{aligned}
x_i + y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \;&=\; x_i + y_i + \sum_{j=h-b}^{h} y_{i-2j+1} \\
&=\; x_{i+2} + y_{i+2} + \sum_{j=h-b}^{h-1} y_{i-2j+1} \\
&=\; \cdots = x_{i-2(h-b)+1} + y_{i-2(h-b)+1} = x_{i+2(b+1)} + y_{i+2(b+1)}.
\end{aligned}$$

As

$$y_{i+2(b+1)+1}(x_{i+2(b+1)}+y_{i+2(b+1)})=0$$

holds for any $(X,Y)\in V(\mathcal{G})$,

$$y_{i+2(b+1)+1}(x_i+y_i+\sum_{j=1}^{h}y_{i-2j+1}\prod_{k=j}^{h}\overline{y_{i-2k}})=0$$

holds for any $(X,Y)\in V(\mathcal{G})$ with $(y_{i+1},y_{i+3},\ldots,y_{i+2t+1},y_{i+2(t+1)+1})=(0,0,\ldots,0,1)$. In other words, the case when $t=b+1$ is proved.

Based on the above proof, for any $(X,Y)\in V(\mathcal{G})$ with $(y_{i+1},y_{i+3},\ldots,y_{i+2h+1})\neq(0,0,\ldots,0)$, Equation 3 always holds. Thus, we are only left with the case when $(y_{i+1},y_{i+3},\ldots,y_{i+2h+1})=(y_{i+1},y_{i+3},\ldots,y_{i+2h-1},y_i)=(0,0,\ldots,0)$. In this case,

$$x_i+y_i+\sum_{j=1}^{h}y_{i-2j+1}\prod_{k=j}^{h}\overline{y_{i-2k}} \quad = \quad x_i+y_i+\sum_{j=1}^{h}y_{i-2j+1}=x_{i-1}+y_{i-1}=\overline{x_i}x_{i+1}.$$

We prove by contradiction that when $(y_{i+1},y_{i+3},\ldots,y_{i+2h+1})=(0,0,\ldots,0)$, $\overline{x_i}x_{i+1}=0$ holds for any $(X,Y)\in V(\mathcal{G})$.

If $\exists(X,Y)\in V(\mathcal{G})$ with $(y_{i+1},y_{i+3},\ldots,y_i)=(0,0,\ldots,0)$ such that $\overline{x_i}x_{i+1}=1$, we immediately obtain

$$x_i=0,x_{i+1}=1. \tag{4}$$

Since $(y_{i+1},y_{i+3},\ldots,y_i)=(0,0,\ldots,0)$, we have

$$\begin{aligned}
0 &= y_{i+1}=x_{i+1}+\overline{x_{i+2}}x_{i+3},\\
0 &= y_{i+3}=x_{i+3}+\overline{x_{i+4}}x_{i+5},\\
&\cdots\\
0 &= y_{i+2h-1}=x_{i-2}+\overline{x_{i-1}}x_i,\\
0 &= y_i=x_i+\overline{x_{i+1}}x_{i+2}.
\end{aligned}$$

Taking Equation 4 into account, we immediately obtain

$$\begin{aligned}
&x_{i+2}=0,x_{i+3}=1,\\
&x_{i+4}=0,x_{i+5}=1,\\
&\ldots,\\
&x_{i-1}=0,x_i=1,\\
&x_{i+1}=0,x_{i+2}=1.
\end{aligned}$$

Therefore, contradictions occur in (x_i,x_{i+1}). Hence, $\overline{x_i}x_{i+1}=0$ holds for any $(X,Y)\in V(\mathcal{G})$ with $(y_{i+1},y_{i+3},\ldots,y_{i+2h+1})=(0,0,\ldots,0)$. In other words, Equation 3 holds for any $(X,Y)\in V(\mathcal{G})$, thus implying $V(\mathcal{G})\subseteq V(\mathcal{W})$ and completing the proof. $\square$

Corollary 1. *For any $t\in[0,h]$ and $i\in[0,n-1]$, we have*

$$x_iy_{i+2t+1}=\begin{cases} y_{i+2t+1}y_i \text{ if } t=0,\\ y_{i+2t+1}(y_i+\sum_{j=h-t+1}^{h}y_{i-2j+1}\prod_{k=j}^{h}\overline{y_{i-2k}}) \text{ if } t\in[1,h].\end{cases} \tag{5}$$

Proof. Based on Theorem 1,

$$x_i = y_i + \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}.$$

Hence,

$$\begin{aligned} x_i y_{i+2t+1} &= x_i y_{i-2(h-t)} \\ &= y_i y_{i+2t+1} + y_{i-2(h-t)} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}. \end{aligned}$$

When $t = 0$, we have

$$\begin{aligned} x_i y_{i+2t+1} &= x_i y_{i-2h} \\ &= y_i y_{i+2t+1} + y_{i-2h} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\ &= y_i y_{i+2t+1}. \end{aligned}$$

When $t \in [1, h]$, we have

$$\begin{aligned} x_i y_{i+2t+1} &= x_i y_{i-2h} \\ &= y_i y_{i+2t+1} + y_{i-2(h-t)} \sum_{j=1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\ &= y_i y_{i+2t+1} + y_{i-2(h-t)} \sum_{j=h-t+1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}} \\ &= y_{i+2t+1}(y_i + \sum_{j=h-t+1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}). \end{aligned}$$

$\square$

Corollary 2. *The degree of the equation*

$$y_{i+2t+1}(x_i + y_i + \sum_{j=h-t+1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}) \quad = \quad 0 \text{ for } t \in [1, h]$$

in terms of (X, Y) *is* $t + 2$.

Proof. The monomial of the highest degree in this equation is $y_{i+2t+1} y_{i+2t+2} \prod_{k=h-t+1}^{h} y_{i-2k}$. Moreover, neither y_{i+2t+1} nor y_{i+2t+2} is a factor of $\prod_{k=h-t+1}^{h} y_{i-2k}$. Therefore, the degree of this equation is $2 + t$. $\square$

Corollary 3. *For* $R \geq 2$ *rounds of* Rasta *of block size* n*, there are at least* $n(2^{R-1} + 1)$ *exploitable equations, as specified below:*

$$\begin{cases} x_i + \overline{y_{i+1}} x_{i+2} + y_i = 0, \\ y_{i+1}(x_i + y_i) = 0, \\ y_{i+2t+1}(x_i + y_i + \sum_{j=h-t+1}^{h} y_{i-2j+1} \prod_{k=j}^{h} \overline{y_{i-2k}}) = 0 \text{ for } t \in [1, 2^{R-1} - 1], \end{cases}$$

where $i \in [0, n-1]$.

Proof. This is directly from $V(\mathcal{G}) \subseteq V(\mathcal{F})$ (Lemma 1), Corollary 1 and Corollary 2. $\square$

Application to Rasta. Based on Corollary 3, for attacks on $r \geq 3$ rounds of Rasta of block size n, we can improve the data complexity by a factor of $\frac{2^{r-1}+1}{5}$ as we now can construct $n(2^{r-1}+1)$ rather than $5n$ equations in terms of the key bits to describe r rounds of Rasta. These equations are obviously linear independent as there is at least one monomial in each equation that does not appear in other equations. Moreover, it can be found that all the $5n$ exploitable equations found in [6] correspond to the cases $t \in [1, 3]$, as shown below:

$$
\begin{aligned}
0 &= x_i + \overline{y_{i+1}}x_{i+2} + y_i, \\
0 &= y_{i+1}(x_i + y_i), \\
0 &= y_{i+3}(x_i + y_i + y_{i+2}\overline{y_{i+1}}), \\
0 &= y_{i+5}(x_i + x_{i+2} + y_i + y_{i+1}y_{i+2} + y_{i+1}\overline{y_{i+3}}y_{i+4}), \\
0 &= y_{i+7}(x_i + y_i + y_{i+6}\overline{y_{i+5}}\ \overline{y_{i+3}}\ \overline{y_{i+1}} + y_{i+4}\overline{y_{i+3}}\ \overline{y_{i+1}} + y_{i+2}\overline{y_{i+1}}).
\end{aligned}
$$

The only equation that does not seem to follow our formula is

$$0 = y_{i+5}(x_i + x_{i+2} + y_i + y_{i+1}y_{i+2} + y_{i+1}\overline{y_{i+3}}y_{i+4}). \tag{6}$$

Indeed, based on our formula, we have

$$
\begin{aligned}
0 &= y_{i+5}(x_i + y_i + y_{i+4}\overline{y_{i+3}}\ \overline{y_{i+1}} + y_{i+2}\overline{y_{i+1}}), \\
0 &= y_{i+5}(x_{i+2} + y_{i+2} + y_{i+4}\overline{y_{i+3}}).
\end{aligned}
$$

Hence,

$$
\begin{aligned}
0 &= y_{i+5}(x_i + y_i + y_{i+4}\overline{y_{i+3}}\ \overline{y_{i+1}} + y_{i+2}\overline{y_{i+1}} + x_{i+2} + y_{i+2} + y_{i+4}\overline{y_{i+3}}) \\
&= y_{i+5}(x_i + x_{i+2} + y_i + y_{i+1}y_{i+2} + y_{i+1}\overline{y_{i+3}}y_{i+4}).
\end{aligned}
$$

In other words, Equation 6 is just a linear combination of the exploitable equations derived based on our formula.

References

[1] A. Biryukov, C. Bouillaguet, and D. Khovratovich. Cryptographic Schemes Based on the ASASA Structure: Black-box, White-box, and Public-key. *IACR Cryptol. ePrint Arch.*, page 474, 2014.

[2] A. Biryukov, C. Bouillaguet, and D. Khovratovich. Cryptographic Schemes Based on the ASASA Structure: Black-Box, White-Box, and Public-Key (Extended Abstract). In *ASIACRYPT (1)*, volume 8873 of *Lecture Notes in Computer Science*, pages 63–84. Springer, 2014.

[3] D. A. Cox, J. Little, and D. O'Shea. *Ideals, varieties, and algorithms - an introduction to computational algebraic geometry and commutative algebra (4. ed.).* Undergraduate texts in mathematics. Springer, 2015.

[4] J. Daemen. Cipher and Hash Function Design Strategies based on Linear and Differential Cryptanalysis. *Ph.D. thesis*, 1995.

[5] C. Dobraunig, M. Eichlseder, L. Grassi, V. Lallemand, G. Leander, E. List, F. Mendel, and C. Rechberger. Rasta: A Cipher with Low ANDdepth and Few ANDs per Bit. In *CRYPTO (1)*, volume 10991 of *Lecture Notes in Computer Science*, pages 662–692. Springer, 2018.

[6] F. Liu, S. Sarkar, W. Meier, and T. Isobe. Algebraic Attacks on Rasta and Dasta Using Low-Degree Equations. In *ASIACRYPT (1)*, volume 13090 of *Lecture Notes in Computer Science*, pages 214–240. Springer, 2021.