Efficient Evaluation of Security against Generalized Interpolation Attack

Kazumaro Aoki

NTT Laboratories
1-1 Hikarinooka, Yokosuka-shi, Kanagawa-ken, 239-0847 Japan
maro@isl.ntt.co.jp

Abstract. Interpolation attack was presented by Jakobsen and Knudsen at FSE'97. Interpolation attack is effective against ciphers that have a certain algebraic structure like the $\mathcal{PURE}$ cipher which is a prototype cipher, but it is difficult to apply the attack to real-world ciphers. This difficulty is due to the difficulty of deriving a low degree polynomial relation between ciphertexts and plaintexts. In other words, it is difficult to evaluate the security against interpolation attack. This paper generalizes the interpolation attack. The generalization makes easier to evaluate the security against interpolation attack. We call the generalized interpolation attack *linear sum attack*. We present an algorithm that efficiently evaluates the security of byte-oriented ciphers against linear sum attack. Moreover, we show the relationship between linear sum attack and higher order differential attack. In addition, we show the security of CRYPTON, E2, and RIJNDAEL against linear sum attack using the algorithm.

1 Introduction

Interpolation attack [4] was presented for attacking the $\mathcal{PURE}$ cipher [14,4], though the $\mathcal{PURE}$ cipher is provably secure [14,13] against differential cryptanalysis [1] and linear cryptanalysis [8]. The basic idea of interpolation attack is as follows: First, the attack focuses on the algebraic structure in the cipher. Next, the attack tries to express ciphertexts using a polynomial of a plaintext. The applicability of the attack is determined by the degree of the polynomial above, more precisely, by the number of the unknown coefficients of the polynomial.

It is easy to find the degree of the polynomial for the $\mathcal{PURE}$ cipher since the non-linear operation in the $\mathcal{PURE}$ cipher is only a cubic operation in $\mathrm{GF}(2^n)$. However, it is basically difficult to find the degree for real-world ciphers. We know of only two examples of successful interpolation attacks against ciphers. One is an attack [4] on a modified version of SHARK [15]. The other is an attack [10] on SNAKE [6]. The non-linear operation of both ciphers is an inversion in $\mathrm{GF}(2^n)$, which is also a simple operation in $\mathrm{GF}(2^n)$. On the other hand, nobody knows a cipher which is provably secure against interpolation attack.

First, this paper introduces the concept of linear sum attack, a generalization of interpolation attack. Introducing linear sum attack leads to a clear vista on studying the security against interpolation attack. Next, the paper proposes an

Howard Heys and Carlisle Adams (Eds.): SAC'99, LNCS 1758, pp. 135–146, 2000.

effective algorithm which judges whether linear sum attack is applicable or not for a given cipher. Moreover, we show a relationship between linear sum attack and higher order differential attack [5,4]; provable security against linear sum attack implies provable security against higher order differential attack. Finally, we evaluate the security of CRYPTON [7], E2 [11], and RIJNDAEL [3] against linear sum attack using the security evaluation algorithm for linear sum attack.

2 Preliminaries

2.1 Notations and Analysis Target

This paper studies the following situation. Let p be a plaintext and c be a ciphertext. Let $c = E_k(p)$ be a block cipher whose block is n-bits long with a product structure. The encryption key k is in the set K $(= \{k_1, k_2, \ldots, k_L\})$. $E_k(p)$ consists of R round functions $F_{k^{(r)}}$ $(r = 1, 2, \ldots, R)$ as follows:

$$E_k(p) = (F_{k^{(R)}} \circ F_{k^{(R-1)}} \circ \cdots \circ F_{k^{(1)}})(p) \ ,$$

where $k^{(r)}$ is the rth round subkey, generated from k by a key scheduling algorithm.

We define $\tilde{c}$ as the input of the last round,

$$\tilde{c} = F^{-1}_{k^{(R)}}(c) = \tilde{E}_k(p) = (F_{k^{(R-1)}} \circ F_{k^{(R-2)}} \circ \cdots \circ F_{k^{(1)}})(p) \ .$$

Moreover, we consider the following maps used in the interpolation attack (see Fig. 1)

$$\begin{aligned} \mathsf{p} &: \ \mathrm{GF}(q) \to \mathrm{GF}(2)^n \\ \tilde{\mathsf{c}}' &: \ \mathrm{GF}(2)^n \to \mathrm{GF}(q) \ , \end{aligned}$$

where $\mathrm{GF}(q)$ is a finite field that contains q elements. This paper considers interpolation attacks using polynomials in $\mathrm{GF}(q)$. Note that we do not assume that q is a power of 2 and p and c' are bijective.

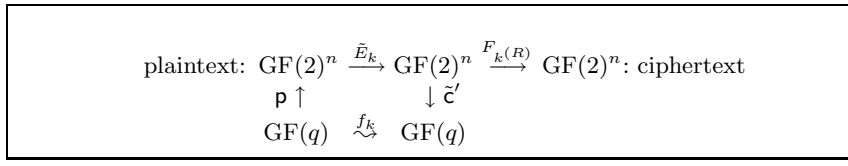

Fig. 1. Attack diagram

2.2 Interpolation Attack

Although several types of interpolation attack are known, this section describes the basic interpolation attack. If the reader is not familiar with interpolation attack, please refer to [4].

An outline of the attack is as follows.

Preparation: Find p and $\tilde{\mathsf{c}}'$ that satisfy

$$\tilde{\mathsf{c}}'(\tilde{E}_k(\mathsf{p}(x))) = f_k(x) \in \mathrm{GF}(q)[x] \ ,$$

by analyzing the target cipher. Let N be the number of the unknown coefficients of the polynomial $f_k(x)$.

Attack:

Step 1: Obtain $N+1$ ciphertexts $c = E_k(\mathsf{p}(x))$ that are derived from the chosen plaintexts $\mathsf{p}(x)$ $(x \in \mathrm{GF}(q))$.

Step 2: Guess $k^{(R)}$ using exhaustive search.

2-1: Calculate $\tilde{c} = F^{-1}_{k^{(R)}}(c)$ from obtained c and guessed $k^{(R)}$ to decrypt 1 round.

2-2: Find $f_k(x)$ from N pairs of $(x, \tilde{\mathsf{c}}'(\tilde{c}))$ using polynomial interpolation.

2-3: Verify the correctness of $f_k(x)$ derived in Step 2-2 using a pair of $(x, \tilde{\mathsf{c}}'(\tilde{c}))$ not used in Step 2-2.

This interpolation attack can be applied if $N < q$ holds. However, it is very difficult to estimate N precisely for a real-world cipher. We give an answer to solve the problem in the following sections.

3 Linear Sum Attack

Consider the interpolation attack replacing the polynomial interpolation with Gaussian elimination in Step 2-2 described in Sect. 2.2. In this case, we can attack a cipher in the same way even if $f_k(x)$ is represented by a linear sum of linearly independent polynomials $b_i(x) \in \mathrm{GF}(q)[x]$ as in

$$f_k(x) = \sum_{i=1}^{q} a_i(k) b_i(x) \quad (a_i(k) \in \mathrm{GF}(q)) \ .$$

We call this attack the *linear sum attack*.

The attack succeeds if the number of unknown $a_i(k)$s is less than q. We estimate the worst case complexity. The number of chosen plaintexts is at most q. The attack requires Gaussian eliminations corresponding to all possible values of $k^{(R)}$. It is well known that Gaussian elimination requires $O(q^3)$ arithmetic operations in $\mathrm{GF}(q)$. So, the attack requires $O(Lq^3)$ arithmetic operations in $\mathrm{GF}(q)$ and L evaluations of F^{-1}.

Linear sum attack is equivalent to interpolation attack, if $b_i(x) = x^{i-1}$ holds, that is $b_i(x)$ is a monomial. Consider the case of $f_k(x) = g(k) \cdot x^1 + 2g(k) \cdot 1$,

for example. If we apply interpolation attack described in Sect. 2.2, we need 3 chosen plaintexts since the number of unknown coefficients is 2, which are $g(k)$ and $2g(k)$. On the other hand, applying linear sum attack, we can factorize the polynomial to $f_k(x) = g(k) \cdot (x + 2)$. This means that we need only 2 chosen plaintexts, since the number of unknown coefficients is 1, which is $g(k)$. As shown by this example, linear sum attack requires less or equal number of chosen plaintexts than interpolation attack.

4 Search for Effective Basis

This section discusses how to find an effective basis $\{b_1(x), b_2(x), \ldots, b_q(x)\}$ for linear sum attack. Linear sum attack requires a basis while interpolation attack requires a polynomial expression of ciphertexts, where we regard a plaintext as a variable for interpolation attack. This section introduces an effective search algorithm for finding an effective basis.

We focus on the following properties of GF(q).

1. Any function over GF(q) can be expressed by a polynomial over GF(q).
2. The set of all functions over GF(q) is a q-dimensional vector space, GF(q)[x][1].
3. Any polynomial over GF(q) can be expressed by a linear sum of a basis $\{b_1(x), b_2(x), \ldots, b_q(x)\}$, where $b_i(x) \in \mathrm{GF}(q)[x]$ $(i = 1, 2, \ldots, q)$.

Using the above facts, we developed an algorithm for finding a basis $\{b_1(x), b_2(x), \ldots, b_q(x)\}$ so that $f_k(x) = \tilde{\mathsf{c}}'(\tilde{E}_k(\mathsf{p}(x)))$ has the fewest unknown coefficients when $f_k(x)$ is expressed by a linear sum using the basis.

Assume that $f_k(x)$ is expressed as

$$f_k(x) = \sum_{i=1}^{q} a_i(k) b_i(x) \quad (a_i(k) \in \mathrm{GF}(q)) \ .$$

The smallest number of unknown coefficients we want to find is

$$N = \mathrm{rank} \begin{bmatrix} a_1(k_1) & a_2(k_1) & \cdots & a_q(k_1) \\ a_1(k_2) & a_2(k_2) & \cdots & a_q(k_2) \\ \cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots \\ a_1(k_L) & a_2(k_L) & \cdots & a_q(k_L) \end{bmatrix} \ .$$

It is practically impossible to calculate the rank described above for all bases and for all keys $k_1, k_2, \ldots, k_L$, since the complexity exceeds an exhaustive search for a key. We solve the problem by the following theorems.

Theorem 1. *The expectation of d is less than $q + 2$, where d is defined as*

$$\dim_{\mathrm{GF}(q)} \langle v_1, v_2, \ldots, v_d \rangle = q \ ,$$

for randomly chosen v_i in the q-dimensional vector space over GF(q).

[1] For simple description, we use GF(q)[x] for GF(q)[x]/($x^q - x$).

Proof. Since a randomly chosen element in the q-dimensional vector space over $\mathrm{GF}(q)$ is contained in a particular i-dimensional ($i \leq q$) subspace with probability $\frac{q^i}{q^q}$, we need to choose, on average, $\frac{1}{1-\frac{q^i}{q^q}}$ elements in order to find one that is not in the subspace. Thus, the expectation of d can be evaluated as follows.

$$\begin{aligned}
&\sum_{i=0}^{q-1} \frac{1}{1-\frac{q^i}{q^q}} = \sum_{i=0}^{q-1}(1+\frac{q^i}{q^q-q^i}) = q + \sum_{i=1}^{q} \frac{1}{q^i-1} \\
&\leq q + \frac{1}{q-1} + \sum_{i=2}^{q} \frac{1}{q^{i-1}} = q + \frac{2-(\frac{1}{q})^{q-1}}{q-1} \\
&\leq q+2
\end{aligned}$$

□

Theorem 2. *$q+r$ ($r \geq 0$) randomly chosen vectors in the q-dimensional vector space over $\mathrm{GF}(q)$ span at least the $(q-1)$-dimensional subspace with probability at most q^{-r}.*

Proof.

$$\begin{aligned}
&\Pr_{v_1,v_2,\ldots,v_{q+r}}[\dim_{\mathrm{GF}(q)} \langle v_1, v_2, \ldots, v_{q+r}\rangle \leq q-1] \\
&= \sum_{i=0}^{q-1} \Pr_{v_1,v_2,\ldots,v_{q+r}}[\dim_{\mathrm{GF}(q)} \langle v_1, v_2, \ldots, v_{q+r}\rangle = i]
\end{aligned}$$

Since the dimension of the vector space $\langle v_1, v_2, \ldots, v_{q+r}\rangle$ is i, we can choose i vectors which span the i-dimensional vector space from $\{v_1, v_2, \ldots, v_{q+r}\}$. We assume $\dim_{\mathrm{GF}(q)} \langle v_1, v_2, \ldots, v_i\rangle = i$ without loss of generality.

$$\begin{aligned}
&\sum_{i=0}^{q-1} \Pr_{v_1,v_2,\ldots,v_{q+r}}[\dim_{\mathrm{GF}(q)} \langle v_1, v_2, \ldots, v_{q+r}\rangle = i] \\
&= \sum_{i=0}^{q-1} \Pr_{v_1,v_2,\ldots,v_{q+r}}[\{v_{i+1}, v_{i+2}, \ldots, v_{q+r}\} \subseteq \langle v_1, v_2, \ldots, v_i\rangle] \\
&= \sum_{i=0}^{q-1} \left(\frac{q^i}{q^q}\right)^{q+r-(i+1)+1} \\
&= \sum_{i=0}^{q-1} q^{-(q-i)(q+r-i)} \\
&\leq q \cdot q^{-(q-(q-1))(q+r-(q-1))} \\
&= q^{-r}
\end{aligned}$$

□

Corollary 3. *$q + r$ ($r \geq 0$) randomly chosen vectors in the q-dimensional vector space over $\mathrm{GF}(q)$ span the q-dimensional subspace with probability at least $1 - q^{-r}$.*

Assume that $f_k(x)$ is random in $\mathrm{GF}(q)[x]$ if we randomly choose k. Then, according to Theorem 1 and Corollary 3, it is sufficient to calculate N, i.e. the rank, using $q + 2$ randomly chosen keys k.

Thus, we can find the basis for the smallest number of coefficients with probability at least $1 - q^{-2}$ by calculating

$$N = \mathrm{rank} \begin{bmatrix} a_1(k_{i_1}) & a_2(k_{i_1}) & \cdots & a_q(k_{i_1}) \\ a_1(k_{i_2}) & a_2(k_{i_2}) & \cdots & a_q(k_{i_2}) \\ \cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots \\ a_1(k_{i_{q+2}}) & a_2(k_{i_{q+2}}) & \cdots & a_q(k_{i_{q+2}}) \end{bmatrix},$$

where $\{k_{i_1}, k_{i_2}, \ldots, k_{i_{q+2}}\}$ is a random subset of K and $a_1(k_{i_j}), a_2(k_{i_j}), \ldots, a_q(k_{i_j})$ ($j = 1, 2, \ldots, q + 2$) are coefficients of the polynomial basis $\{1, x, \cdots, x^{q-1}\}$ derived by some polynomial interpolation algorithm. Since a rank is an invariable with different bases, it is sufficient to consider only the polynomial basis.

We summarize the basis search algorithm.

Algorithm 4.

Step 1: *Choose appropriate parameters for the attack:*
- *a finite field* $\mathrm{GF}(q)$
- *a map* $\mathsf{p} : \mathrm{GF}(q) \to \mathrm{GF}(2)^n$
- *a map* $\tilde{\mathsf{c}}' : \mathrm{GF}(2)^n \to \mathrm{GF}(q)$

Step 2: *Generate $q + 2$ randomly chosen keys $k_{i_1}, k_{i_2}, \ldots, k_{i_{q+2}} \in K$.*

Step 3: *Calculate all input-output pairs of $f_{k_{i_j}}$ $(= \tilde{\mathsf{c}}' \circ \tilde{E}_{k_{i_j}} \circ \mathsf{p})$,*

$$(x, f_{k_{i_j}}(x))$$

for all $x \in \mathrm{GF}(q)$ and $1 \leq \forall j \leq q + 2$.

Step 4: *Using some polynomial interpolation algorithm, determine coefficients*

$$a_1(k_{i_j}), a_2(k_{i_j}), \ldots, a_q(k_{i_j})$$

of polynomial $f_{k_{i_j}}(x) = \sum_{l=1}^{q} a_l(k_{i_j}) x^{l-1}$ for $q+2$ keys k_{i_j} $(1 \leq j \leq q+2)$ using the input-output pairs of $f_{k_{i_j}}$ calculated in Step 3.

Step 5: *Calculate the number of effective coefficients*

$$N = \mathrm{rank} \begin{bmatrix} a_1(k_{i_1}) & a_2(k_{i_1}) & \cdots & a_q(k_{i_1}) \\ a_1(k_{i_2}) & a_2(k_{i_2}) & \cdots & a_q(k_{i_2}) \\ \cdots\cdots & \cdots\cdots & \cdots & \cdots\cdots \\ a_1(k_{i_{q+2}}) & a_2(k_{i_{q+2}}) & \cdots & a_q(k_{i_{q+2}}) \end{bmatrix}$$

using Gaussian elimination.

A proper program for Gaussian elimination to calculate the rank can also find the effective basis for an attack.

A cipher is secure against linear sum attack if N equals q. In other words, linear sum attack is effective if N is less than q.

We studied the complexity of the above algorithm. Note that in Step 4 we can interpolate polynomials for each k_{i_j} by calculating only 1 Gaussian elimination, which requires $O(q^3)$ arithmetic operations in GF(q). The algorithm requires $O(q^4)$ $(=(q+2) \times O(q^3) + O(q))$ arithmetic operations in GF(q), with the assumption that the encryption time is much less than Gaussian elimination. Thus, it is sufficient for recent computers to calculate N if $q \approx 2^8$.

5 Experimental Results

This section evaluates the security of CRYPTON, E2, and RIJNDAEL, using Algorithm 4. CRYPTON, E2, and RIJNDAEL have 12, 12, and 10 rounds, respectively, and the basic operations of these ciphers are 8 bits long.

Unfortunately, since it is infeasible to check all combinations of GF(q), p, $\tilde{\mathsf{c}}'$, we ran the algorithm for only the following combinations.

- $\mathrm{GF}(q) = \mathrm{GF}(2^8)$
- $\mathsf{p}_i : x \mapsto (0, \ldots, 0, \overset{i\text{th}}{\breve{x}}, 0, \ldots, 0)$ $(i = 1, 2, \ldots, 16)$
- $\tilde{\mathsf{c}}'_j : (x_1, x_2, \ldots, x_{16}) \mapsto x_j$ $(j = 1, 2, \ldots, 16)$

The results are summarized in Table 1. We evaluated only the 128-bit key versions of the ciphers[2]. We count the number of rounds as 0 in the case of the cipher with only initial transformation.

Table 1. Smallest number of unknown coefficients

Number of Rounds	CRYPTON	E2	E2*	RIJNDAEL
0	1	1	—	1
1	1	1	0	1
2	252	1	1	255
3	255	1	1	255
≥ 4	256	256	256	256

*: without IT- and FT-Functions

According to Table 1, there are no long linear relation of these ciphers comparing with the number of rounds of the specification of these ciphers. It seems that these ciphers are secure against generalized interpolation attack, linear sum attack.

[2] We evaluated only the 128-bit block length version of RIJNDAEL.

The goal of this paper is the security evaluation of a given cipher against linear sum attack. Thus, we do not go into the details of the attacks for these ciphers, however, a rough sketch of the attacks using Table 1 are shown in the Appendix.

6 Relationship between Linear Sum Attack and Higher Order Differential Attack

This section describes the strength of linear sum attack in comparison with higher order differential attack.

Definition 5. *$E_k(p)$ is* secure against linear sum attack *with respect to* $\mathrm{GF}(q)$, p, *and* $\tilde{\mathsf{c}}'n \overset{\text{def}}{\Leftrightarrow} N = q$ *holds, where N is determined by Algorithm 4.*

Definition 6. *Let $\tilde{e}_k^{(i)}(p)$ be the ith output bit of $\tilde{E}_k(p)$, i.e.,*

$$\tilde{E}_k(p) = (\tilde{e}_k^{(1)}(p), \tilde{e}_k^{(2)}(p), \ldots, \tilde{e}_k^{(n)}(p)) \ .$$

Let p be a map

$$\mathsf{p} : \mathrm{GF}(2)^t \to \mathrm{GF}(2)^n;\ (x_1, x_2, \ldots, x_t) \mapsto (p_1, p_2, \ldots, p_n) \ ,$$

where $p_i = \begin{cases} x_{\pi^{-1}(i)} & \text{if } \pi^{-1}(i) \text{ is defined} \\ \text{constant} & \text{otherwise} \end{cases}$ *and π is an injective map from $\{1, 2, \ldots, t\}$ to $\{1, 2, \ldots, n\}$.*

$E_k(p)$ is secure against higher order differential attack *with respect to* p *and* $u \overset{\text{def}}{\Leftrightarrow} \deg_{\{x_1, x_2, \ldots, x_t\}} \tilde{e}_k^{(u)}(\mathsf{p}(x_1, x_2, \ldots, x_t)) = t$ *holds.*

Note that Definition 6 does not consider improved higher order differential attacks such as proposed in [9] and the case of $t = n$, and if a cipher is not secure against higher order differential attack according to Definition 6, we cannot conclude that the cipher is insecure against an actual higher order differential attack.

The following theorem means that resistance against linear sum attack, which is a generalized interpolation attack, implies resistance against higher order differential attack.

Theorem 7. *Let* p *be a map*

$$\mathsf{p} : \mathrm{GF}(2^t) \to \mathrm{GF}(2)^n;\ (x_1, x_2, \ldots, x_t) \mapsto (p_1, p_2, \ldots, p_n)\ ,$$

where $p_i = \begin{cases} x_{\pi^{-1}(i)} & \text{if } \pi^{-1}(i) \text{ is defined} \\ \text{constant} & \text{otherwise} \end{cases}$ *and* π *is an injective map from* $\{1, 2, \ldots, t\}$ *to* $\{1, 2, \ldots, n\}$. *Let* $\tilde{\mathsf{c}}'$ *be a map*

$$\tilde{\mathsf{c}}' : \mathrm{GF}(2)^n \to \mathrm{GF}(2^t);\ (\tilde{c}_1, \tilde{c}_2, \ldots, \tilde{c}_n) \mapsto (y_1, y_2, \ldots, y_t)\ ,$$

where $y_i = \tilde{c}_{\tau(i)}$ *and* τ *is an injective map from* $\{1, 2, \ldots, t\}$ *to* $\{1, 2, \ldots, n\}$.

For $1 \le \forall i \le t$, $E_k(p)$ *is secure against linear sum attack with respect to* $\mathrm{GF}(2^t)$, p, *and* $\tilde{\mathsf{c}}' \Rightarrow E_k(p)$ *is secure against higher order differential attack with respect to* p *and* $\tau(i)$.

Note that we regard an element $(a_1, a_2, \ldots, a_t) \in \mathrm{GF}(2)^t$ *as* $a \in \mathrm{GF}(2^t)$ *with* $\mathrm{GF}(2)$ *basis.*

Before proving Theorem 7, we show a well-known lemma. This lemma was introduced by [12, Proposition 4, p.60], for example.

Lemma 8. *Let* $y = x^d$ *in* $\mathrm{GF}(2^t)$ *and regard* $(y_1, y_2, \ldots, y_t) \in \mathrm{GF}(2)^t$ *as* y *with* $\mathrm{GF}(2)$ *basis.*

$$\deg_{\{x_1, x_2, \ldots, x_t\}} y_i = w_{\mathrm{H}}(d) \quad \text{for } 1 \le \forall i \le t$$

holds, where x *is regarded as* $(x_1, x_2, \ldots, x_t) \in \mathrm{GF}(2)^t$ *with* $\mathrm{GF}(2)$ *basis and* $w_{\mathrm{H}}(d)$ *is the Hamming weight of the binary representation of* d.

Proof (of Theorem 7). According to the assumption of the theorem and Definition 5, $\tilde{E}_k(\mathsf{p}(x))$ should be expressed as

$$\tilde{E}_k(\mathsf{p}(x)) = \sum_{i=1}^{2^t} a_i(k) x^{i-1}\ ,$$

where $a_i(k) \in \mathrm{GF}(2^t)$ is an unknown coefficient for $1 \le \forall i \le 2^t$. Using Lemma 8,

$$\begin{aligned} w_{\mathrm{H}}(x^d) &= t \text{ if } d = 2^t - 1 \\ &< t \text{ otherwise} \end{aligned}$$

holds. Since the degree-t term of Boolean representation of $\tilde{e}_k^{(\tau(i))}(\mathsf{p}(x))$ comes only from x^{2^t-1} and never comes from x^d $(d < 2^t - 1)$,

$$\deg_{\{x_1, x_2, \ldots, x_t\}} \tilde{e}_k^{(\tau(i))}(\mathsf{p}(x)) = t$$

holds for $1 \le \forall i \le t$. □

7 Conclusion

This paper presented linear sum attack, which is a generalized form of interpolation attack, and presented an algorithm that efficiently evaluates the security of a cipher against linear sum attack. We applied the algorithm to 128-bit key CRYPTON, E2, and RIJNDAEL, which have 12, 12, and 10 rounds, respectively, and showed that the ciphers reduced to 3 rounds have non-trivial linear sum relations. Moreover, we showed that resistance against linear sum attack implies resistance against higher order differential attack.

There are 2 open problems remaining.

1. How to find effective $\mathrm{GF}(q)$, p, $\tilde{\mathsf{c}}'$?
2. How to construct a rational version of linear sum attack like interpolation attack?

Acknowledgment

We wish to thank T. Shimoyama and anonymous referees of workshops and conferences for giving us comments. Some of them conflicted, but they significantly improved the presentation of our paper.

References

1. Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991. (The extended abstract was presented at CRYPTO'90).
2. Joan Daemen, Lars Ramkilde Knudsen, and Vincent Rijmen. The block cipher SQUARE. In Eli Biham, editor, *Fast Software Encryption — 4th International Workshop, FSE'97*, volume 1267 of *Lecture Notes in Computer Science*, pages 54–68, Berlin, Heidelberg, New York, 1997. Springer-Verlag.
3. Joan Daemen and Vincent Rijmen. *AES Proposal: Rijndael*, 1998. (`http://www.esat.kuleuven.ac.be/~rijmen/rijndael/`).
4. Thomas Jakobsen and Lars Ramkilde Knudsen. The interpolation attack on block cipher. In Eli Biham, editor, *Fast Software Encryption — 4th International Workshop, FSE'97*, volume 1267 of *Lecture Notes in Computer Science*, pages 28–40, Berlin, Heidelberg, New York, 1997. Springer-Verlag.
5. Lars Ramkilde Knudsen. Truncated and higher order differentials. In Bart Preneel, editor, *Fast Software Encryption — Second International Workshop*, volume 1008 of *Lecture Notes in Computer Science*, pages 196–211. Springer-Verlag, Berlin, Heidelberg, New York, 1995.
6. Chang-Hyi Lee and Young-Tae Cha. The block cipher: SNAKE with provable resistance against DC and LC attacks. In *1997 Korea-Japan Joint Workshop on Information Security and Cryptology (JW-ISC'97)*, pages 3–17, Seoul, KOREA, 1997. KIISC (Korea) and ISEC Group of IEICE (Japan).

7. Chae Hoon Lim. *CRYPTON: A New 128-bit Block Cipher – Specification and Analysis –*. Future Systems, 1998.
(urlhttp://crypt.future.co.kr/ chilim/crypton.html).
8. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology — EUROCRYPT'93*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer-Verlag, Berlin, Heidelberg, New York, 1994. (A preliminary version written in Japanese was presented at SCIS93-3C).
9. Shiho Moriai, Takeshi Shimoyama, and Toshinobu Kaneko. Higher order differential attack using chosen higher order differences. In Stafford Tavares and Henk Meijer, editors, *Selected Areas in Cryptography — 5th Annual International Workshop, SAC'98*, volume 1556 of *Lecture Notes in Computer Science*, pages 106–117, Berlin, Heidelberg, New York, 1999. Springer-Verlag.
10. Shiho Moriai, Takeshi Shimoyama, and Toshinobu Kaneko. Interpolation attacks of the block cipher: SNAKE. In Lars Ramkilde Knudsen, editor, *Fast Software Encryption — 6th International Workshop, FSE'99*, volume 1636 of *Lecture Notes in Computer Science*, pages 275–289, Berlin, Heidelberg, New York, 1999. Springer-Verlag. (A preliminary version written in Japanese was presented at SCIS'98-7.2.C).
11. Nippon Telegraph and Telephone Corporation. *Specification of E2 — a 128-bit Block Cipher*, 1998. (`http://info.isl.ntt.co.jp/e2/`).
12. Kaisa Nyberg. Differentially uniform mappings for cryptography. In Tor Helleseth, editor, *Advances in Cryptology — EUROCRYPT'93*, volume 765 of *Lecture Notes in Computer Science*, pages 55–64. Springer-Verlag, Berlin, Heidelberg, New York, 1994.
13. Kaisa Nyberg. Linear approximation of block ciphers. In Alfredo De Santis, editor, *Advances in Cryptology — EUROCRYPT'94*, volume 950 of *Lecture Notes in Computer Science*, pages 439–444. Springer-Verlag, Berlin, Heidelberg, New York, 1995.
14. Kaisa Nyberg and Lars Ramkilde Knudsen. Provable security against a differential attack. *Journal of Cryptology*, 8(1):27–37, 1995. (A preliminary version was presented at CRYPTO'92 rump session).
15. Vincent Rijmen, Joan Daemen, Bart Preneel, Antoon Bosselaers, and Erik De Win. The cipher SHARK. In Dieter Gollmann, editor, *Fast Software Encryption — Third International Workshop*, volume 1039 of *Lecture Notes in Computer Science*, pages 99–111. Springer-Verlag, Berlin, Heidelberg, New York, 1996.

Appendix: Linear Sum Attack of Reduced Round Variants of CRYPTON, E2, and RIJNDAEL

We evaluate the security against linear sum attack for CRYPTON, E2, and RIJNDAEL using the results shown in Table 1. Since these linear sum attacks are not superior than the known attacks against the ciphers and the attack procedures are almost the same as the interpolation attack described in Sect. 2.2, we do not analyze and describe the details.

First, we consider CRYPTON and RIJNDAEL. Both ciphers are based on the same structure of SQUARE [2], and there are 3-round linear sum relations with $N < q$. Applying the 3-round linear sum relation from the 2nd round to the 4th round, and guessing the 1st, the 5th, and the 6th round subkeys related to the linear sum relation exhaustively, we can attack the ciphers reduced to 6 rounds faster than exhaustive search. The attack is almost the same as SQUARE attack [3, pp.28–31].

Next, we consider E2. There exists 3-round linear sum relations with $N < q$ in spite of the existence of IT- and FT-Functions. We can attack E2 with IT- and FT-Functions reduced to 3 rounds faster than exhaustive search, by applying the linear sum relation from the 1st round to the 3rd round, and guessing key bits used in FT-Function related to the linear sum relation. We can attack E2 without IT- and FT-Functions reduced to 5 rounds faster than exhaustive search, by applying the linear sum relation from the 2nd round to the 4th round, and guessing the 1st and the 5th round subkey bits related to the linear sum relation.