Distinguishing Attack on SOBER-128 with linear masking

Joo Yeon Cho and Josef Pieprzyk

Centre for Advanced Computing – Algorithms and Cryptography,
Department of Computing, Macquarie University,
NSW, Australia, 2109
{jcho,josef}@ics.mq.edu.au

Abstract. We present a distinguishing attack against SOBER-128 with linear masking. We found a linear approximation which has a bias of $2^{-8.8}$ for the non-linear filter. The attack applies the observation made by Ekdahl and Johansson that there is a sequence of clocks for which the linear combination of some states vanishes. This linear dependency allows that the linear masking method can be applied. We also show that the bias of the distinguisher can be improved (or estimated more precisely) by considering quadratic terms of the approximation. The probability bias of the quadratic approximation used in the distinguisher is estimated to be equal to $O(2^{-51.8})$, so that we claim that SOBER-128 is distinguishable from truly random cipher by observing $O(2^{103.6})$ keystream words.
Keywords : Distinguishing attack, Stream ciphers, Linear masking, Modular addition, SOBER-128

1 Introduction

One of the recent trends in designing stream ciphers is that stream ciphers are word-oriented. Since the operation of ciphers are based on words and keystreams are produced word by word at each clock, they are fast and efficient when implemented in software. This class of ciphers includes SNOW [3], SOBER [4], MUGI [2] and many others. In particular, among eSTREAM stream cipher submissions, the word-oriented ciphers are Dragon, Phelix, NLS, HC-256 to mention a few [1].

The SOBER-128 is one of recently proposed word-oriented stream ciphers. The cipher is built using the classical structure with a linear feedback shift register (LFSR) and a non-linear filter function. SOBER-128 is an improved version of SOBER-t32 which was a candidate of the stream cipher primitives in NESSIE project [10]. The non-linear function has been strengthened by adding a fixed rotation and the second S-box transformation. The stuttering phase that was present in SOBER-t32 is not used in SOBER-128.

In this work, we develop a distinguishing attack on SOBER-128 with linear masking introduced by Coppersmith, Halevi and Jutla at CRYPTO 2002 [6]. The authors of [6] study two types of distinguishing characteristic of non-linear processes : the linear approximation and the low diffusion. We use the linear approximation to develop the attack against SOBER-128. In addition, we combine a quadratic polynomial with the linear approximations for a precise estimation of the expected probability bias.

The authors of [6] shows that if there is a linear approximation σ of the non-linear function with bias ϵ, then the bit $\xi_j = \bigoplus_{j \in J} \sigma_j$ has the bias of $\epsilon^{|J|}$, where J is a set of steps such that $\bigoplus_{j \in J} s_j = 0$, provided s_j is a state bit of a linear feedback shift register.We claim that the bias of ξ_j could be slightly higher than $\epsilon^{|J|}$ when quadratic terms are considered.

Our attack on SOBER-128 is based on two linear approximations that exhibit a big enough probability bias. We observe that the bias of the quadratic approximation for non-linear filter of SOBER-128 is $O(2^{-51.8})$. Therefore, we claim that SOBER-128 is distinguishable from a random process by observing around $O(2^{103.6})$ keystream words.

This paper is organized as follows. In Section 2, the distinguishing attack with linear masking using a linear approximation is briefly described. In Section 3, the structure of SOBER-128 is given. In Section 4, we derive linear approximations on the nonlinear Filter (NLF). In Section 5, a linear distinguishing attack is applied by using the derived approximation. Section 6 applies an improved distinguishing attack using a quadratic approximation. Conclusions are given in Section 7.

2 Linear masking using linear approximation

We describe briefly the linear masking method for the linear attack which is presented in [6]. The attack is applicable for a class of stream ciphers with a special structure that consist of the linear process (LF) and the non-linear process (NF). The state in a such cipher is identified by a pair: linear state x and non-linear state z. The cipher works in steps (clocks) and at each step i, the cipher

- sets the linear state as $x_i := LF(x_{i-1})$,
- calculates two variables $u_i := L1(x_i)$ and $v_i := L2(x_i)$, where $L1, L2$ are linear functions,
- determines non-linear state $z_i := NF(z_{i-1} \oplus u_i) \oplus v_i$,
- outputs z_i.

Assume that we have a linear function $l : \{0,1\}^{2n} \to \{0,1\}$, such that

$$Pr[l(z, NF(z)) = 0] = \frac{1}{2}(1+\epsilon), \quad |\epsilon| > 0^1$$

in other words, the function l is a linear approximation of the non-linear function NF and ϵ is the bias of the approximation.

Suppose that the adversary observes a bit $\sigma_j = l(z_j \oplus u_j, NF(z_j) \oplus v_j)$ where the variables u and v come from a linear space. Then there is always a linear combination of steps (not necessarily consecutive) for which the variables u and v vanish. Let J be a set of such steps for which $\bigoplus_{j\in J} u_j = \bigoplus_{j\in J} v_j = 0$. Thus, we can write

$$\begin{aligned} &\textstyle\bigoplus_{j\in J} \sigma_j = \bigoplus_{j\in J} l(z_j, NF(z_j)) \oplus \bigoplus_{j\in J} l(u_j + v_j) \\ &= \textstyle\bigoplus_{j\in J} l(z_j, NF(z_j)) \end{aligned} \tag{1}$$

Therefore, if the number of elements in the set J is n, $\bigoplus_{j\in J} \sigma_j$ has the bias of ϵ^n.

Using this bias, an adversary can reliably distinguish the stream cipher from the random process by observing around ϵ^{-2n} outputs. For more details, see [6].

[1] This definition is simple for the bias of multiple approximations when the piling-up lemma is considered. If we have n independent approximations, the probability of n approximations becomes $\frac{1}{2}(1+\epsilon^n)$. Whereas, if p is defined by a form of $p = \frac{1}{2} + \epsilon$, the probability of n approximations becomes $\frac{1}{2}(1+2^{n-1}\epsilon^n)$.

3 Brief description of SOBER-128

The SOBER-128 consists of a linear feedback shift register (LFSR) and a nonlinear filter (NLF). The LFSR consists of 17 words state registers which is denoted by the vector $(s_t, \cdots, s_{t+16})$. Since each s_i is a 32-bit integer, the size of LFSR is 544 bits. The new state of the LFSR is generated by the following connection polynomial

$$s_{t+17} = s_{t+15} \oplus s_{t+4} \oplus \gamma s_t,$$

where γ = 0x00000100 (hexadecimal).

A Nonlinear Filter (NLF) produces an output word z_t by taking s_t, s_{t+1}, s_{t+6}, s_{t+13}, s_{t+16} from the LFSR states and the constant K. The NLF consists of two substitution functions (S-box), one rotation, four adders modulo 2^{32} and three XOR additions. For the detail description of the NLF, see Figure 1.

The K is a 32-bit key-dependent constant. The function f is defined as $f(a) = \text{S-box}(a_H) \oplus a$ where the S-box is 8×32-bit and a_H is the most significant 8 bits of 32-bit word a. The output z_t of the nonlinear filter is described as

$$z_t = f((((f(s_t \boxplus s_{t+16}) \ggg 8) \boxplus s_{t+1}) \oplus K) \boxplus s_{t+6}) \boxplus s_{t+13},$$

where $\boxplus$ denotes an addition modulo 2^{32} and $\ggg 8$ denotes a 8-bit right rotation. The LFSR states and a constant K are initialized from the 128-bit secret key using the initialization procedure. More details can be found in [8].

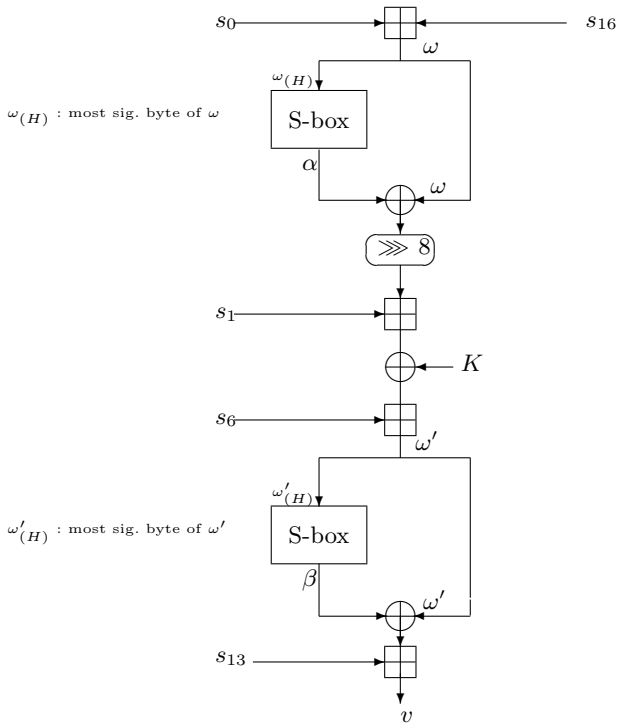

Fig. 1. The non-linear filter (NLF) of SOBER-128

4 Deriving linear approximations on NLF

According to the structure of the non-linear filter, the following equation holds for the least significant bit (see Figure 1). Let us denote that α is 32-bit output of the first S-box, β is 32-bit output of the second S-box and ω is 32-bit output of the addition of s_0 and s_{16}, respectively. Then, the following equation holds at any clock

$$\alpha_{(8)} \oplus \beta_{(0)} \oplus \omega_{(8)} \oplus s_{1,(0)} \oplus s_{6,(0)} \oplus s_{13,(0)} \oplus K_{(0)} = z_{(0)}, \tag{2}$$

where $x_{t,(i)}$ stands for the i-th bit of the 32-bit word x at clock t. (This notation will be also used for the other equations.)

We will find the best linear approximation for $\alpha_{(8)}$, $\beta_{(0)}$ and $\omega_{(8)}$. In order to apply the linear masking method for a distinguisher of SOBER-128, we use a low weight linear relationship among the states of LFSR which was presented for attack on SOBER-t32 [7]. The LFSR of SOBER-128 is not same as that of SOBER-t32 but the following relationship still holds for both stream ciphers

$$s_{t+\tau_1} \oplus s_{t+\tau_2} \oplus s_{t+\tau_3} \oplus s_{t+\tau_4} \oplus s_{t+\tau_5} \oplus s_{t+\tau_6} = 0 \tag{3}$$

with $\tau_1 = 0, \tau_2 = 11, \tau_3 = 13, \tau_4 = 4 \cdot 2^{32} - 4, \tau_5 = 15 \cdot 2^{32} - 4, \tau_6 = 17 \cdot 2^{32} - 4$. This linear recurrence is valid for each bit position individually.

4.1 Linear approximations of $\alpha_{(8)}$.

The bit $\alpha_{(8)}$ is the 8-th output bit of the first S-box. The input of the S-box is the most significant 8-bit of the addition of the state register s_0 and s_{16}. Thus, $\alpha_{(8)}$ is completely determined by both s_0 and s_{16} registers. However, the input of the S-box is mostly affected by the most significant 8 bits of the register s_0 (which is called $s_{0,(H)}$) and s_{16} (which is called $s_{16,(H)}$), respectively. Hence, we try to find the best linear approximation for $\alpha_{(8)}$ from the whole set of linear combinations of $s_{0,(H)}$ and $s_{16,(H)}$. In order to calculate the correlation of each combination, we introduce the carry bit $carry_1$, which is induced from the addition of two 24 least significant bits of s_0 and s_{16}. We regard the bit $carry_1$ as a uniform and independent variable. Then,

$$\text{The input of the first S-box} = s_{0,(H)} \boxplus s_{16,(H)} \boxplus carry_1$$

We build the truth table with 2^{17} rows and 2^{16} columns. Each row corresponds to the unique collection of input variables (8 bits of $s_{0,(H)}$, 8 bits of $s_{16,(H)}$, and a single bit for $carry_1$). Each column relates to the unique linear combination of bits from $s_{0,(H)}$ and $s_{16,(H)}$. In result, we have found four linear approximation for $\alpha_{(8)}$, which have the best bias (see Table 1).

Let us choose the first approximation from the table so

$$\alpha_{(8)} \approx s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)} \tag{4}$$

and the probability is $\frac{1}{2}(1 - 0.057618) = \frac{1}{2}(1 - 2^{-4.1})$.

4.2 Linear approximation of $\beta_{(0)}$.

The best linear approximation of $\beta_{(0)}$ can be obtained by a similar approach we have applied for $\alpha_{(8)}$ with an addition trick.

Table 1. The best linear approximations for $\alpha_{(8)}$

linear approximation	bias
$s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)}$	1/2(1 - 0.057618)
$s_{0,(26)} \oplus s_{0,(29)} \oplus s_{16,(25)} \oplus s_{16,(26)} \oplus s_{16,(28)} \oplus s_{16,(29)}$	1/2(1 - 0.057618)
$s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(25)} \oplus s_{16,(26)} \oplus s_{16,(29)}$	1/2(1 - 0.057618)
$s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(28)} \oplus s_{16,(29)}$	1/2(1 - 0.057618)

The S-box of the SOBER-128 consists of two different S-boxes which are the Skipjack S-box and the S-box that was custom-designed by the researchers from QUT. Using the structure of S-Box, we can observe that not only the input of the second S-box but also the 8-bit output of the S-box determines the bit β_0 completely. The output of the Skipjack S-box is the most significant 8 bits of the subtraction of the state register s_{13} from the output z. Thus, $\beta_{(0)}$ is determined by both s_{13} and z. However, in a similar way to $\alpha_{(8)}$, the most significant 8 bits of the register s_{13} (which is called $s_{13,(H)}$) and the output z (which is called $z_{(H)}$) contribute to β_0. Hence, we try to find the best linear approximation for $\beta_{(0)}$ from the whole set of linear combinations of $s_{13,(H)}$ and $z_{(H)}$.

In order to calculate the best linear approximation, we also introduce the carry bit $carry_2$ which is induced from the addition of two 24 least significant bits of the register s_{13} and the output of the second f-function. We regard the bit $carry_2$ as a uniform and independent variable. So,

$$\text{The output of the Skipjack S-box} \boxplus s_{13,(H)} \boxplus carry_2 = z_{(H)}$$

In a similar way to $\alpha_{(8)}$, we build the truth table with 2^{17} rows and 2^{16} columns for $\beta_{(0)}$. Each row corresponds to the collection of variables (8 bits of $s_{13,(H)}$, 8-bit output of the Skipjack S-box , and a single bit for $carry_2$). Each column relates to the unique linear combination of bits from $s_{13,(H)}$ and $z_{(H)}$.

Table 2 displays the best and the second best linear approximations of $\beta_{(0)}$.

Table 2. Linear approximations on $\beta_{(0)}$

linear approximation	bias
$s_{13,(29)} \oplus s_{13,(30)} \oplus z_{(29)} \oplus z_{(30)}$	1/2(1+0.07666)
$s_{13,(31)} \oplus z_{(31)}$	1/2(1+0.072388)
$s_{13,(30)} \oplus s_{13,(31)} \oplus z_{(30)} \oplus z_{(31)}$	1/2(1+0.072388)

Hence, the best linear approximation on $\beta_{(0)}$ is such that

$$\beta_{(0)} \approx s_{13,(29)} \oplus s_{13,(30)} \oplus z_{(29)} \oplus z_{(30)} \tag{5}$$

with the probability of $\frac{1}{2}(1 + 0.07666) = \frac{1}{2}(1 + 2^{-3.7})$.

Remark. We may improve the bias by considering non-linear approximations for $\beta_{(0)}$ in such a way that the approximations take the following form.

$$\beta_{(0)} = \text{linear}(s_{13,(H)}) \oplus \text{nonlinear}(z_{(H)})$$

Since only linear($s_{13,(H)}$) vanishes by the linear masking method and nonlinear($z_{(H)}$) becomes a part of a distinguisher, we may improve the bias by manipulating all the non-linear monomials which are generated by the 8 bits of $z_{(H)}$.

4.3 Linear approximations of $\omega_{(8)}$

The bit $\omega_{(8)}$ is the 8-th bit of output which is produced by adding the registers s_0 and s_{16}. Clearly $\omega_{(8)}$ is determined by the least significant 9 bits of s_0 and and s_{16} (which are denoted as $s_{0,(L)}$ and $s_{16,(L)}$ respectively). Thus,

$$\omega_{(8)} = (s_{0,(L)} \boxplus s_{16,(L)})_{(8)} \tag{6}$$

In order to find the best approximation for $\omega_{(8)}$, a truth table is constructed by considering all the possible linear combinations among the bit string $s_{0,(L)}$ and $s_{16,(L)}$. In result, we found the four best linear approximations for $\omega_{(8)}$ with same bias (see Table 3). Let us

Table 3. The best linear approximations for $\omega_{(8)}$

linear approximation	bias
$s_{0,(8)} \oplus s_{0,(7)} \oplus s_{16,(8)}$	1/2(1+0.5)
$s_{0,(8)} \oplus s_{16,(8)} \oplus s_{16,(7)}$	1/2(1+0.5)
$s_{0,(8)} \oplus s_{0,(7)} \oplus s_{0,(0)} \oplus s_{16,(8)} \oplus s_{16,(0)}$	1/2(1+0.5)
$s_{0,(8)} \oplus s_{0,(0)} \oplus s_{16,(8)} \oplus s_{16,(7)} \oplus s_{16,(0)}$	1/2(1+0.5)

choose the first approximation from the table. Then,

$$\omega_{(8)} \approx s_{0,(8)} \oplus s_{16,(8)} \oplus s_{0,(7)} \tag{7}$$

and the probability of approximation is $\frac{1}{2}(1 + 2^{-1})$.

5 Distinguishing attack on SOBER-128 with linear masking

Recall Equation (2) on NLF. If we replace $\alpha_{(8)}$, $\beta_{(0)}$ and $\omega_{(8)}$ by Approximations (4), (5) and (7) respectively, we build a linear approximation on NLF as follows.

$$\begin{aligned} &\underbrace{s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)}}_{\alpha_{(8)}} \oplus \underbrace{s_{13,(29)} \oplus s_{13,(30)} \oplus z_{(29)} \oplus z_{(30)}}_{\beta_{(0)}} \\ &\oplus \underbrace{s_{0,(8)} \oplus s_{16,(8)} \oplus s_{0,(7)}}_{\omega_{(8)}} \oplus s_{1,(0)} \oplus s_{6,(0)} \oplus s_{13,(0)} \oplus K_{(0)} = z_{(0)} \end{aligned} \tag{8}$$

where the bias is

$$\frac{1}{2}(1 + 2^{-4.1} \cdot 2^{-3.7} \cdot 2^{-1}) = \frac{1}{2}(1 + 2^{-8.8}) \tag{9}$$

Let us divide Approximation (8) into two parts : a linear combination of the state bits and that of the output bits. Then, Approximation (8) will be

$$\begin{aligned} &s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)} \oplus s_{13,(29)} \oplus s_{13,(30)} \\ &\oplus s_{0,(8)} \oplus s_{16,(8)} \oplus s_{0,(7)} \oplus s_{1,(0)} \oplus s_{6,(0)} \oplus s_{13,(0)} \oplus K_{(0)} \\ &= z_{(0)} \oplus z_{(29)} \oplus z_{(30)} \end{aligned} \tag{10}$$

If we apply the linear masking method described in Section 2, then, the left part of Approximation (10) with linear masking vanishes by the linear connection of Equation (3).

Therefore, we can build a distinguisher of $\bigoplus_{t=\tau_1}^{\tau_6}(z_{(0)} \oplus z_{(29)} \oplus z_{(30)})$ with the bias of $(\mathbf{2^{-8.8}})^{\mathbf{6}} = \mathbf{2^{-52.8}}$.

6 An improved distinguishing attack on SOBER-128

In this section, we improve the bias of the distinguisher by introducing an idea of quadratic approximations with linear masking. This idea is applied to the approximation of $\omega_{(8)}$. We show that the bit $\omega_{(8)}$ with linear masking is $(2^{-1})^5$ rather than $(2^{-1})^6$.

This section is organized as follows. First, we derive a general formula for the bias of a quadratic monomial with linear masking. Then, the formula is applied to the modular addition which is the case of $\omega_{(8)}$.

6.1 Correlation of quadratic monomials

Let us assume that a connection polynomial of LFSR has the weight n. That is, $\bigoplus_{i=1}^{n} x_i = 0$ where x_i represents one bit of the state register. Then, the weight of the vector $\rho = (x_1, x_2, \ldots, x_n)$ is always even. This means that one of the component of ρ is completely determined by the others. In general, the space of ρ is 2^{n-1}.

If we consider a monomial of degree d such that $\sigma_d = x_{i_1}x_{i_2}\cdots x_{i_d}$, then, the monomial σ_d is correlated due to the restriction on the space of ρ. It is clear that such correlation always exists and is dependent on the degree d and the weight n.

Let us consider a quadratic monomial which is the simplest form of non-linear function.

Lemma 1. *Given two vectors $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ and $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)$ such that $\bigoplus_{i=1}^{n} x_i = 0$ and $\bigoplus_{i=1}^{n} y_i = 0$, then $\sigma_{xy} = \bigoplus_{i=1}^{n} x_i y_i$ is a Boolean function in $GF(2^{2n}) \rightarrow GF(2)$ with the bias determined by the following probability*

$$Pr[\sigma_{xy} = 0] = \begin{cases} \frac{1}{2}(1 + 2^{-n+2}) & n \text{ is even} \\ \frac{1}{2}(1 + 2^{-n+1}) & n \text{ is odd} \end{cases} \tag{11}$$

Proof *We count how many times the zero (or one) happens when all the possible values of two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are considered.*

At first, let us consider when n is even. Assuming that $x_1 = \cdots = x_n = 0$. Then, σ_{xy} is always zero for all values of $\boldsymbol{y}$. Thus, the zero count is 2^{n-1}.

Secondly, let assume that $x_1 = \cdots = x_n = 1$. Then, σ_{xy} is again always zero for all values of $\boldsymbol{y}$ because the weight of $\boldsymbol{y}$ is even. Thus, the zero count increases 2^{n-1}.

In other values of $\boldsymbol{x}$, the number of one is equal to that of zero. Thus, the zero count increases $2^{n-2} \cdot (2^{n-1} - 2) = 2^{n-1} \cdot (2^{n-2} - 1)$.

All together, the zero count becomes $2^{n-1} + 2^{n-1} + 2^{n-1} \cdot (2^{n-2} - 1)$. Therefore, the correlation becomes

$$\frac{2^{n-1} + 2^{n-1} + 2^{n-1} \cdot (2^{n-2} - 1)}{2^{n-1} \cdot 2^{n-1}} = \frac{1 + 1 + 2^{n-2} - 1}{2^{n-1}} = \frac{1}{2}(1 + 2^{-n+2})$$

A proof is similar when n is odd. □

The following corollary is useful when a combined monomial with an output bit is considered.

Corollary 1. *If the vector* $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ *satisfies the condition* $\bigoplus_{i=1}^{n} x_i = 0$ *but the vector* $\boldsymbol{y} = (y_1, y_2, \cdots, y_n)$ *does not, then*

$$Pr[\sigma_{xy} = 0] = \frac{1}{2}(1 + 2^{-n+1})$$

Proof *A proof is similar to Lemma 1.*

6.2 Quadratic approximation of $\omega_{(8)}$ with linear masking

Recall Equation (6). The bit $\omega_{(8)}$ can be expressed as a quadratic polynomial by using the previous $\omega_{(7)}$ bit recursively in a following way.

$$\begin{cases} \omega_{(0)} = s_{0,(0)} \oplus s_{16,(0)} \\ \omega_{(1)} = s_{0,(1)} \oplus s_{16,(1)} \oplus s_{0,(0)} s_{16,(0)} \\ \cdots \\ \omega_{(8)} = s_{0,(8)} \oplus s_{16,(8)} \oplus s_{0,(7)} s_{16,(7)} \oplus (s_{0,(7)} \oplus s_{16,(7)})(1 \oplus \omega_{(7)}) \end{cases} \tag{12}$$

If we apply the linear masking method, then,

$$Pr[\bigoplus_{t=\tau_1}^{\tau_6} \omega_{t,(8)} = 0] = Pr[\bigoplus_{t=\tau_1}^{\tau_6} (s_{0,(7)} s_{16,(7)} \oplus (s_{0,(7)} \oplus s_{16,(7)})(1 \oplus \omega_{(7)})) = 0] \tag{13}$$

Note that $\bigoplus_{t=\tau_1}^{\tau_6} (s_{0,(8)} \oplus s_{16,(8)}) = 0$.
Since the bit $\omega_{(7)}$ can be regarded as a (almost) balanced variable, the correlation of Equation (13) can be estimated by building a truth table where there are the condition that $\bigoplus_{t=\tau_1}^{\tau_6} s_{t,(7)} = \bigoplus_{t=\tau_1}^{\tau_6} s_{t+16,(7)} = 0$ but no condition on $\omega_{t,(7)}$, which corresponds the condition of Corollary 1. In result, a bit $\bigoplus_{t=\tau_1}^{\tau_6} \omega_{t,(8)}$ has the bias of around 2^{-5}. Experiments confirmed this result. See Appendix A.

6.3 Improved bias of the distinguisher

Recall again Equation (2) on NLF. If we replace $\alpha_{(8)}$ and $\beta_{(0)}$ by Approximations (4) and (5) respectively, but remain $\omega_{(8)}$, then, we build an approximation on NLF as follows.

$$\begin{aligned} &\underbrace{s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)}}_{\alpha_{(8)}} \oplus \underbrace{s_{13,(29)} \oplus s_{13,(30)} \oplus z_{(29)} \oplus z_{(30)}}_{\beta_{(0)}} \\ &\oplus \omega_{(8)} \oplus s_{1,(0)} \oplus s_{6,(0)} \oplus s_{13,(0)} \oplus K_{(0)} = z_{(0)} \end{aligned} \tag{14}$$

with the bias of $2^{-4.1} \cdot 2^{-3.7} = 2^{-7.8}$.
Let us denote Approximation (14) simply as follows.

$$l_1(s) \oplus \omega_{(8)} = l_2(z) \tag{15}$$

where

$$\begin{aligned} l_1(s) = {} & s_{0,(25)} \oplus s_{0,(26)} \oplus s_{0,(28)} \oplus s_{0,(29)} \oplus s_{16,(26)} \oplus s_{16,(29)} \oplus s_{13,(29)} \\ & \oplus s_{13,(30)} \oplus s_{1,(0)} \oplus s_{6,(0)} \oplus s_{13,(0)} \oplus K_{(0)} \\ l_2(z) = {} & z_{(0)} \oplus z_{(29)} \oplus z_{(30)} \end{aligned} \tag{16}$$

If we apply the linear masking method to Approximation (15),

$$\bigoplus_{t=\tau_1}^{\tau_6}(l_1(s) \oplus \omega_{(8)}) = \bigoplus_{t=\tau_1}^{\tau_6}(z_{(0)} \oplus z_{(29)} \oplus z_{(30)}) \tag{17}$$

Due to the linear connection of state bits by Equation (3) and Approximation (13), the left part of Approximation (17) vanishes with the probability of

$$\frac{1}{2}(1 + (2^{-7.8})^6 * 2^{-5}) = \frac{1}{2}(1 + 2^{-51.8}) \tag{18}$$

Therefore, in fact, a distinguisher of $\bigoplus_{t=\tau_1}^{\tau_6}(z_{(0)} \oplus z_{(29)} \oplus z_{(30)}) = 0$ has the bias of $2^{-51.8}$. Even though the distinguisher has not been changed, the usage of a quadratic terms improves the bias of distinguisher by a factor of 2, which reflects more accurate bias of the distinguisher.

7 Conclusions

In this paper, we show a distinguishing attack with linear masking against SOBER-128 stream cipher. This is the first work which presents an attack on SOBER-128. In particular, this work is interesting to eSTREAM project because the S-box of SOBER-128 is re-used for the NLS cipher [9, 5] which is one of the candidate stream ciphers. We estimate the correlation of a distinguisher by deriving a quadratic approximation on NLF.

Our attack shows that the correlation of the distinguisher with linear masking could be higher than the estimation at the paper [6] by considering a quadratic terms with a factor of at least 2.

References

1. The home page for eSTREAM. http://www.ecrypt.eu.org/stream/.
2. The home page for MUGI. http://www.sdl.hitachi.co.jp/crypto/mugi/index-e.html.
3. The home page for SNOW. http://www.it.lth.se/cryptology/snow/.
4. The home page for SOBER. http://www.qualcomm.com.au/Sober.html.
5. Joo Yeon Cho and Josef Pieprzyk. Linear distinguishing attack on nls. eSTREAM, ECRYPT Stream Cipher Project, Report 2006/018, 2006. http://www.ecrypt.eu.org/stream/papersdir/2006/018.pdf.
6. Charanjit Jutla Don Coppersmith, Shai Halevi. Cryptanalysis of stream ciphers with linear masking. Advances in Cryptology - CRYPTO 2002: 22nd Annual International Cryptology Conference Santa Barbara, California, USA, August 18-22, 2002. Proceedings. M. Yung (Ed.): Lecture Notes in Computer Science, Volume 2442, Jan 2002, Pages 515 - 532.
7. P. Ekdahl and T.Johansson. Distinguishing attacks on SOBER-t16 and t32. In V. Rijmen J. Daemen, editor, *Fast Software Encryption*, volume LNCS 2365, pages 210–224. Springer-Verlag, 2002.
8. G. Rose P. Hawkes. Primitive specification for SOBER-128. http://www.qualcomm.com.au/Sober128.html, Apr. 2003.
9. G. Rose P. Hawkes. Primitive specification for NLS. http://www.ecrypt.eu.org/stream/nls.html, Apr. 2005.
10. P.Hawkes and G.Rose. Sober. Primitive submitted to NESSIE by Qualcomm International, Sep. 2000.

A Experiments for Section 6.2

Experiments are begun by finding the initial states which would satisfy the following linear relation of the LFSR.

$$s_{t+\tau_1} \oplus s_{t+\tau_2} \oplus s_{t+\tau_3} \oplus s_{t+\tau_4} \oplus s_{t+\tau_5} \oplus s_{t+\tau_6} = 0$$

with $\tau_1 = 0, \tau_2 = 11, \tau_3 = 13, \tau_4 = 4 \cdot 2^{32} - 4, \tau_5 = 15 \cdot 2^{32} - 4, \tau_6 = 17 \cdot 2^{32} - 4$. Table 4 displays an example of initial states of $\tau_1, \ldots, \tau_6$. Note that all contents of the table are hexadecimal.

When $t = \tau_1$, we compute ω_8 by conducting $(s_0 \boxplus s_{16})_8$. (e.g. from the table, b0213cbe $\boxplus$ 7c0c7591 = 2c2db24f so that $\omega_8 = 0$) The same calculations are performed for $t = \tau_2$ to $t = \tau_6$. In result, we have 6 bits of ω_8 so that we can compute $\bigoplus_{t=\tau_1}^{\tau_6} \omega_{t,(8)}$. We carry on this process for $t = \{\tau_1 + 1, \ldots, \tau_6 + 1\}, t = \{\tau_1 + 2, \ldots, \tau_6 + 2\}$ and so on. New state is generated by the LFSR connection polynomial.

By counting the number of zeros (or ones) of the bit value $\bigoplus_{t=\tau_1}^{\tau_6} \omega_{t,(8)}$ at every clock, we can compute the probability which is the number of zeros (or ones) divided by the number of clocks.

The experiment shows that $\Pr[\bigoplus_{t=\tau_1}^{\tau_6} \omega_{t,(8)} = 0]$ is around $\frac{1}{2}(1 + 2^{-5})$ which was expected in Section 6.2.

Table 4. An example of initial states for the linear relation of LFSR

Register	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6
s_0	b0213cbe	81144c40	ea5f4936	80b626f2	7daca7b7	2670b88d
s_1	dee601f9	c0849eda	0da3e7a9	fd36421d	08f60296	e6013801
s_2	bb9d85af	18ce1254	a89d02b9	398e7a8a	80b626f2	b2f6c93a
s_3	6ddd2873	3937a5e3	19537890	e8eb08ef	fd36421d	5864bff2
s_4	3b3abd0f	6e162713	9e3a4268	6a8c43fa	398e7a8a	9814e104
s_5	98f4854a	e5ad513c	db7a3b35	387b5c1f	e8eb08ef	76b3bbb3
s_6	e77fc5c1	9983c08f	b100b099	0bfe370f	6a8c43fa	ae8ec122
s_7	b59aa80a	1e709998	a5c26138	1cfa270e	387b5c1f	2aa92bbb
s_8	9d0a4482	48ffd86a	b7368175	3b72bba8	0bfe370f	524f913a
s_9	2c927b9c	8c1aa656	a11f1bfb	983fe11e	1cfa270e	85520021
s_{10}	824e4c06	76126b97	713b00eb	79f12dc9	3b72bba8	c7e4b11b
s_{11}	d2389fa0	910a6bb8	a50ed952	03af6be3	983fe11e	7daca7b7
s_{12}	420962cd	0518c989	ec5437cf	da42b3d4	79f12dc9	08f60296
s_{13}	35949133	cbf0c10f	38fea16b	4583bc46	03af6be3	80b626f2
s_{14}	2be0a38b	3b5e5827	426bdfe3	75a1d586	da42b3d4	fd36421d
s_{15}	183186a9	83fe1b6a	db2c18b4	3cee43bb	4583bc46	398e7a8a
s_{16}	7c0c7591	d05172be	394a13c0	085dc986	75a1d586	e8eb08ef