Crossword Puzzle Attack on NLS

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. NLS is one of the stream ciphers submitted to the eSTREAM project. We present a distinguishing attack on NLS by Crossword Puzzle (CP) attack method which is newly introduced in this paper. We build the distinguisher by using linear approximations of both the non-linear feedback shift register (NFSR) and the non-linear filter function (NLF). Since the bias of the distinguisher depends on the $Konst$ value, which is a key-dependent word, we present the graph showing how the bias of distinguisher vary with $Konst$. In result, we estimate the average bias to be around $O(2^{-30})$. Therefore, we claim that NLS is distinguishable from truly random cipher after observing $O(2^{60})$ keystream words on the average. The experiments also show that our distinguishing attack is successful on 90.3% of $Konst$ among 2^{32} possible values.
Keywords : Distinguishing Attacks, Stream Ciphers, Linear Approximations, eSTREAM, Modular Addition, NLS.

1 Introduction

The European Network of Excellence in Cryptology (ECRYPT) launched a stream cipher project called eSTREAM [1] whose aim is to come up with a collection of stream ciphers that can be recommended to industry and government institutions as secure and efficient cryptographic primitives. It is also likely that some or perhaps all recommended stream ciphers may be considered as de facto industry standards. It is interesting to see a variety of different approaches used by the designers of the stream ciphers submitted to the eSTREAM call. A traditional approach for building stream ciphers is to use a linear feedback shift register (LFSR) as the main engine of the cipher. The outputs of the registers are taken and put into a nonlinear filter that produces the output stream that is added to the stream of plaintext.

One of the new trends in the design of stream ciphers is to replace LFSR by a nonlinear feedback shift register (NFSR). From the ciphers submitted to the eSTREAM call, there are several ciphers that use the structure based on NFSR amongst them the NLS cipher follows this design approach. The designers of the NLS cipher are Philip Hawkes, Gregory Rose, Michael Paddon and Miriam Wiggers de Vries from Qualcomm Australia.

The paper studies the NLS cipher and its resistance against distinguishing attacks using linear approximation. Typically, distinguishing attacks do not allow to recover any secret element of the cipher such as the cryptographic key or the secret initial state of the NFSR but instead they permit to tell apart the cipher from the truly random cipher. In this sense these attacks are relatively weak. However, the existence of a distinguishing attack is considered as an early warning sign of possible major security flaws.

In our analysis, we derive linear approximations of both NFSR and the nonlinear filter (NLF). The main challenge has been to combine the obtained linear approximations in a

such way that the internal state bits of NFSR have been eliminated leaving the observable output bits only. We call this type of attack as **"Crossword Puzzle" attack** since the state bits of approximations vanish by combining them in a horizontal way as well as in a vertical way.

Our approach is an extension of the linear masking method introduced by Coppersmith, Halevi, and Jutla in [3]. Note that the linear masking method was applied for the traditional stream ciphers based on LFSR so it is not directly applicable for the ciphers with NFSR.

The work is structured as follows. Section 2 presents a framework of CP attack. Section 3 briefly describes the NLS cipher. In Section 4, we study best linear approximations for both NFSR and NLF. A simplified NLS cipher is defined in Section 5 and we show how to design a distinguisher for it. Our distinguisher for the original NLS cipher is examined in Section 6. We show how it works and also discuss its limitations. Section 8 concludes our work.

2 Framework of Crossword Puzzle (CP) Attack

The CP attack is a linear distinguishing attack which is applicable to a class of stream ciphers that consist of the non-linear feedback shift register (NFSR) and the non-linear filter (NLF). In general, the roles of the two non-linear components are as follows.

- NFSR transforms the current state s_i into the next state s_{i+1} in a non-linear way using the appropriate function $NF1$, i.e. $s_{i+1} := NF1(s_i)$ where s_0 is the initial state and $i = 0, 1, 2, \ldots$.
- NLF produces an output z_i from the current state s_i through a non-linear function $NF2$, i.e. $z_i := NF2(s_i)$.

Let us define a bias ϵ of an approximation as $p = \frac{1}{2}(1+\epsilon), |\epsilon| > 0$ where p is the probability of the approximation. [1] The CP attack is composed of the following steps. Note that the operation + is a binary addition.

1. Find a linear approximation of the non-linear state transition function $NF1$ used by NFSR : $l_1(s_i, s_{i+1}) = 0$ with bias of ϵ_1.
2. Find a linear approximation of the non-linear function $NF2$ applied by NLF : $l_2(s_j) + l_3(z_j) = 0$ with bias of ϵ_2.
3. Obtain two sets of clocks I and J such that $\sum_{i \in I} l_1(s_i, s_{i+1}) = \sum_{j \in J} l_2(s_j)$.
4. Build a distinguisher by computing

$$\sum_{i \in I} l_1(s_i, s_{i+1}) + \sum_{j \in J} (l_2(s_j) + l_3(z_j)) = \sum_{j \in J} l_3(z_j) = 0$$

   which has bias of $\epsilon_1^{|I|} \cdot \epsilon_2^{|J|}$.

For the CP attack, it is an important task to find the approximations in Step 1 and Step 2 which have the relation described in Step 3. We describe a basic framework for achieving this task.

Given $l_1(s_i, s_{i+1}) = 0$, we divide l_1 into n linear sub-functions $u_1, \ldots, u_n$. That is,

$$l_1(s_i, s_{i+1}) = u_1(s_i) + \cdots + u_{n-1}(s_i) + u_n(s_{i+1}) \quad (1)$$

[1] This definition is useful 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 $\frac{1}{2}(1+\epsilon^n)$.

If we set up a system of m approximations of l_1 on the clocks $i = i_1, \ldots, i_m$, then, we have

$$\begin{array}{ll} l_1(s_{i_1}, s_{i_1+1}) & = u_1(s_{i_1}) + u_2(s_{i_1}) + \cdots + u_n(s_{i_1+1}) \\ l_1(s_{i_2}, s_{i_2+1}) & = u_1(s_{i_2}) + u_2(s_{i_2}) + \cdots + u_n(s_{i_2+1}) \\ & \cdots \\ l_1(s_{i_m}, s_{i_m+1}) & = u_1(s_{i_m}) + u_2(s_{i_m}) + \cdots + u_n(s_{i_m+1}) \end{array} \tag{2}$$

Now, our goal is to find a set of linear approximations from NLF which corresponds to each column of Approximation (2), which are eventually replaced by n linear functions of output which are denoted as $\zeta_1(z_{j_1}), \ldots, \zeta_n(z_{j_n})$. That is,

$$\begin{array}{l} u_1(s_{i_1}) + u_1(s_{i_2}) + \cdots + u_1(s_{i_m}) = \zeta_1(z_{j_1}) \\ u_2(s_{i_1}) + u_2(s_{i_2}) + \cdots + u_2(s_{i_m}) = \zeta_2(z_{j_2}) \\ \cdots \\ u_n(s_{i_1+1}) + u_n(s_{i_2+1}) + \cdots + u_n(s_{i_m+1}) = \zeta_n(z_{j_n}) \end{array} \tag{3}$$

Note that each line of approximation of (3) corresponds each column of Approximation (2). Since NLF is assumed to produces an output z_j from the current (single) state s_j, the states of each approximation in (3) should be unified to a single state.

It is practically possible since most of states of NFSR are linearly shifted except one non-linearly updated state. Thus, any sub-linear function $u_i(s_j)$ can be converted to another linear function $v_j(s_i)$, i.e. $u_i(s_j) = v_t(s_k)$. Therefore, each approximation of (3) is converted to the following approximation which can be derived from NLF.

$$\begin{array}{l} v_{11}(s_{j_1}) + v_{12}(s_{j_1}) + \cdots + v_{1m}(s_{j_1}) = \zeta_1(z_{j_1}) \\ v_{21}(s_{j_2}) + v_{22}(s_{j_2}) + \cdots + v_{2m}(s_{j_2}) = \zeta_2(z_{j_2}) \\ \cdots \\ v_{n1}(s_{j_n}) + v_{n2}(s_{j_n}) + \cdots + v_{nm}(s_{j_n}) = \zeta_n(z_{j_n}) \end{array} \tag{4}$$

If we combine Approximations (2) and (3) (or equivalently Approximation (4)), all the states vanish and a distinguisher is produced by computing $\sum_{i=1}^{n} \zeta_j(z_{j_i}) = 0$. The relation is true with a non-zero bias.

There are more issues in regard to the bias of the distinguisher. Firstly, we assume that all approximations are independent. However, practically, this is not always accurate since many terms in the approximations could be related. This means that $\epsilon_1^{|I|} \cdot \epsilon_2^{|J|}$ provides as a lower bound of the bias of the distinguisher and in practice, the distinguisher is going to work better. The precise value of the bias can be computed by analysis of conditional probabilities of random variables of states involved in the approximations.

Secondly, when we set up a system of m approximations, we may choose different approximations instead of using the same single approximation $l_1(s_i, s_{i+1})$ m times. In general, it is an interesting research problem of selection of approximations for both NFSR and NLF in order to maximize the bias of the distinguisher.

Note that the CP attack can be seen as an extension of the linear masking method introduced by Coppersmith, Halevi, and Jutla in [3] with the reason that the CP attack is reducible to the linear masking method when the NFSR is replaced by a linear feedback shift register (LFSR) with $\epsilon_1 = 1$.

3 Brief description of NLS stream cipher

As we said the NLS keystream generator uses NFSR whose outputs are given to the nonlinear filter NLF that produces output keystream bits. Note that we concentrate on the cipher

itself and ignore its message integrity function as irrelevant to our analysis. For details of the cipher, the reader is referred to [2].

NLS has two components: NFSR and NLF that are synchronised by a clock. The state of NFSR at time t is denoted by $\sigma_t = (r_t[0], \ldots, r_t[16])$ where $r_t[i]$ is a 32-bit word. The state is determined by 17 words (or equivalently 544 bits). The transition from the state σ_t to the state σ_{t+1} is defined as follows:

1. $r_{t+1}[i] = r_t[i+1]$ for $i = 0, \ldots, 15$;
2. $r_{t+1}[16] = f((r_t[0] \lll 19) \boxplus (r_t[15] \lll 9) \boxplus Konst) \oplus r_t[4]$;
3. if $t = 0$ (modulo f16), $r_{t+1}[2] = r_{t+1}[2] \boxplus t$;

where $f16$ is 65537 and $\boxplus$ is the addition modulo 2^{32}. The *Konst* value is a 32-bit key-dependent constant. The function $f : \{0,1\}^{32} \to \{0,1\}^{32}$ is constructed using an S-box with 8-bit input and 32-bit output and defined as $f(a) = \text{S-box}(a_H) \oplus a$ where a_H is the most significant 8 bits of 32-bit word a. Refer Figure 1. Each output keystream word ν_t of NLF is obtained as

$$\nu_t = NLF(\sigma_t) = (r_t[0] \boxplus r_t[16]) \oplus (r_t[1] \boxplus r_t[13]) \oplus (r_t[6] \boxplus Konst). \tag{5}$$

The cipher uses 32-bit words to ensure a fast keystream generation.

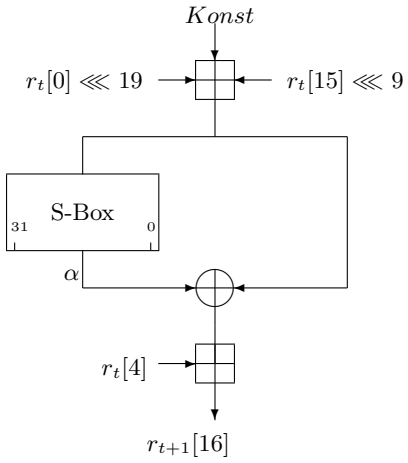

Fig. 1. The f function

4 Analysis of NFSR and NLF

Unlike a LFSR that applies a connection polynomial, the NFSR uses a much more complex nonlinear transition function f that mixes the XOR addition (linear) with the addition modulo 2^{32} (nonlinear). According to the structure of the non-linear shift register, the following equation holds for the least significant bit. Let us denote α_t to be a 32-bit output of the S-box that defines the transition function f. Then, we observe that for the least significant bit, the following equation holds

$$\alpha_{t,(0)} \oplus r_t[0]_{(13)} \oplus r_t[15]_{(23)} \oplus Konst_{(0)} \oplus r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} = 0 \tag{6}$$

where $\alpha_{t,(0)}$ and $x_{(i)}$ stand for the i-th bits of the 32-bit words α_t and x, respectively.
To make our analysis simpler we assume initially that *Konst* is zero. This assumption is later dropped (i.e. *Konst* is non-zero) when we discuss our distinguishing attack on the NLS stream cipher.

4.1 Linear approximations of $\alpha_{t,(0)}$

Recall that α_t is the 32-bit output taken from the S-box and $\alpha_{t,(0)}$ is its least significant bit. The input to the S-box comes from the eight most significant bits of the addition $((r_t[0] \lll 19) \boxplus (r_t[15] \lll 9) \boxplus Konst)$. Assuming that $Konst$=0, the input to S-box is $(r_t[0]' \boxplus r_t[15]')$, where $r_t[0]' = r_t[0] \lll 19$ and $r_t[15]' = r_t[15] \lll 9$. Thus, $\alpha_{t,(0)}$ is completely determined by the contents of two registers $r_t[0]'$ and $r_t[15]'$. Observe that the input of the S-box is affected by the eight most significant bits of the two registers $r_t[0]'$ (we denote the 8 most significant bits of the register by $r_t[0]'_{(H)}$) and $r_t[15]'$ (the 8 most significant bits of the register are denoted by $r_t[15]'_{(H)}$) and by the carry bit c generated by the addition of two 24 least significant bits of $r_t[0]'$ and $r_t[15]'$. Therefore

$$\text{the input of the S-box} = r_t[0]'_{(H)} \boxplus r_t[15]'_{(H)} \boxplus c.$$

Now we would like to find the best linear approximation for $\alpha_{t,(0)}$. 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 $r_t[0]'_{(H)}$, 8 bits of $r_t[15]'_{(H)}$, and a single bit for c). Each column relates to the unique linear combination of bits from $r_t[0]'_{(H)}$ and $r_t[15]'_{(H)}$. Table 1 displays a collection of best linear approximations that are going to be used in our distinguishing attack. In particular, we see that the third approximation of Table 1 has high bias with only two terms. This seems to be caused by the fact that $r_t[0]_{(12)} \oplus r_t[15]_{(22)}$ is the only input to the MSB of input of the S-box that is not diffused to other order bits. Note that

linear approximations of $\alpha_{t,(0)}$	bias
$r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_t[15]_{(20)} \oplus r_t[15]_{(16)} \oplus r_t[15]_{(15)}$	$1/2(1 + 0.048828)$
$r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_t[0]_{(5)} \oplus r_t[15]_{(20)} \oplus r_t[15]_{(16)}$	$1/2(1 + 0.048828)$
$r_t[0]_{(12)} \oplus r_t[15]_{(22)}$	$1/2(1 - 0.045410)$
$r_t[0]_{(12)} \oplus r_t[0]_{(11)} \oplus r_t[0]_{(10)} \oplus r_t[15]_{(22)} \oplus r_t[15]_{(21)} \oplus r_t[15]_{(20)}$	$1/2(1 - 0.020020)$

Table 1. Linear approximations for $\alpha_{t,(0)}$ when $Konst = 0$

$r_t[0]'_{(H)} = (r_t[0] \lll 19)_{(H)} = (r_t[0]_{(12)}, \ldots, r_t[0]_{(5)})$ and $r_t[15]'_{(H)} = (r_t[15] \lll 9)_{(H)} = (r_t[15]_{(22)}, \ldots, r_t[15]_{(15)})$. Note also that none of the approximations contains the carry bit c, in other words, the approximations do not depend on c.

4.2 Linear approximations for NFSR

Having a linear approximation of $\alpha_{t,(0)}$, it is easy to obtain a linear approximation for NFSR. For example, let us choose the first approximation from Table 1. Then, we have the following linear equation:

$$\alpha_{t,(0)} = r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_t[15]_{(20)} \oplus r_t[15]_{(16)} \oplus r_t[15]_{(15)} \tag{7}$$

with the bias $0.048828 = 2^{-4.36}$. Now we combine Equations (6) and (7) and as the result we have the following approximation for NFSR

$$\begin{array}{l} r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_t[15]_{(20)} \oplus r_t[15]_{(16)} \oplus r_t[15]_{(15)} \\ \oplus r_t[0]_{(13)} \oplus r_t[15]_{(23)} \oplus Konst_{(0)} \oplus r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} = 0 \end{array} \tag{8}$$

with the bias of $2^{-4.36}$.

4.3 Linear approximations of modular addition

Let us take a closer look at the modular addition $\boxplus$. We know that the least significant bits are linear so the following equation holds

$$(r[x] \boxplus r[y])_{(0)} = r[x]_{(0)} \oplus r[y]_{(0)}. \tag{9}$$

All consecutive bits $i > 0$ of $\boxplus$ are nonlinear. Consider the function $(r[x] \boxplus r[y])_{(i)} \oplus (r[x] \boxplus r[y])_{(i-1)}$. We observe that the function has a linear approximation as follows

$$(r[x] \boxplus r[y])_{(i)} \oplus (r[x] \boxplus r[y])_{(i-1)} = r[x]_{(i)} \oplus r[y]_{(i)} \oplus r[x]_{(i-1)} \oplus r[y]_{(i-1)} \tag{10}$$

that has the bias of 2^{-1}.

In a similar way, we also observe that the function $(r[x] \boxplus r[y])_{(i)} \oplus (r[x] \boxplus r[y])_{(i-1)} \oplus (r[x] \boxplus r[y])_{(i-2)} \oplus (r[x] \boxplus r[y])_{(i-3)}$ has the following approximation. For $i > 2$,

$$\begin{array}{l} (r[x] \boxplus r[y])_{(i)} \oplus (r[x] \boxplus r[y])_{(i-1)} \oplus (r[x] \boxplus r[y])_{(i-2)} \oplus (r[x] \boxplus r[y])_{(i-3)} = \\ r[x]_{(i)} \oplus r[y]_{(i)} \oplus r[x]_{(i-1)} \oplus r[y]_{(i-1)} \oplus r[x]_{(i-2)} \oplus r[y]_{(i-2)} \oplus r[x]_{(i-3)} \oplus r[y]_{(i-3)} \end{array} \tag{11}$$

that has the bias of 2^{-2}.

4.4 Linear approximation for NLF

Recall that Equation (5) defines the output keystream generated by NLF. By Equation (9), we obtain the relation for the least significant bits of NLF that takes the following form

$$\nu_{t,(0)} = (r_t[0]_{(0)} \oplus r_t[16]_{(0)}) \oplus (r_t[1]_{(0)} \oplus r_t[13]_{(0)}) \oplus (r_t[6]_{(0)} \oplus Konst_{(0)}). \tag{12}$$

This relation holds with probability one.

For $2 \leq i \leq 31$ and using Equation (10), we can argue that NLF function has linear approximations of the following form:

$$\begin{array}{rl} \nu_{t,(i)} \oplus \nu_{t,(i-1)} = & (r_t[0]_{(i)} \oplus r_t[16]_{(i)} \oplus r_t[0]_{(i-1)} \oplus r_t[16]_{(i-1)}) \\ & \oplus (r_t[1]_{(i)} \oplus r_t[13]_{(i)} \oplus r_t[1]_{(i-1)} \oplus r_t[13]_{(i-1)}) \\ & \oplus (r_t[6]_{(i)} \oplus Konst_{(i)} \oplus r_t[6]_{(i-1)} \oplus Konst_{(i-1)}) \end{array} \tag{13}$$

with the bias of $(2^{-1})^2 = 2^{-2}$ under the condition that $Konst = 0$.

Also applying Approximation (11), for $i > 2$, we get the following expression

$$\begin{array}{l} \nu_{t,(i)} \oplus \nu_{t,(i-1)} \oplus \nu_{t,(i-2)} \oplus \nu_{t,(i-3)} = \\ (r_t[0]_{(i)} \oplus r_t[16]_{(i)} \oplus r_t[0]_{(i-1)} \oplus r_t[16]_{(i-1)} \oplus r_t[0]_{(i-2)} \oplus r_t[16]_{(i-2)} \oplus r_t[0]_{(i-3)} \oplus r_t[16]_{(i-3)}) \\ \oplus (r_t[1]_{(i)} \oplus r_t[13]_{(i)} \oplus r_t[1]_{(i-1)} \oplus r_t[13]_{(i-1)} \oplus r_t[1]_{(i-2)} \oplus r_t[13]_{(i-2)} \oplus r_t[1]_{(i-3)} \oplus r_t[13]_{(i-3)}) \\ \oplus (r_t[6]_{(i)} \oplus Konst_{(i)} \oplus r_t[6]_{(i-1)} \oplus Konst_{(i-1)} \oplus r_t[6]_{(i-2)} \oplus Konst_{(i-2)} \oplus r_t[6]_{(i-3)} \oplus Konst_{(i-3)}) \end{array} \tag{14}$$

that has the bias of $(2^{-2})^2 = 2^{-4}$ under the condition that $Konst = 0$.

For non-zero $Konst$, the bias of Approximations (13) and (14) will be studied in Section 6.2.

5 CP attack on a simplified NLS

In this Section we present the CP attack on a simplified NLS. This is a preliminary stage of our attack in which we apply the initial idea of crossword puzzle attack that will be later developed and generalized. We assume that the structure of NFSR is unchanged but the structure of NLF is modified by replacing the addition $\boxplus$ by $\oplus$. Thus, Equation (5) that describes the keystream becomes

$$\mu_t = (r_t[0] \oplus r_t[16]) \oplus (r_t[1] \oplus r_t[13]) \oplus (r_t[6] \oplus Konst). \tag{15}$$

This linear function is valid for 32-bit words so it can be equivalently re-written as a system of 32 equations each equation valid for the particular ith bit. Hence, for $0 \leq i \leq 31$, we can write

$$\mu_{t,(i)} = (r_t[0]_{(i)} \oplus r_t[16]_{(i)}) \oplus (r_t[1]_{(i)} \oplus r_t[13]_{(i)}) \oplus (r_t[6]_{(i)} \oplus Konst_{(i)}). \tag{16}$$

To build a distinguisher we combine approximations of NFSR given by Equation (8) with linear equations defined by Equation (16). For the clocks t, $t+1$, $t+6$, $t+13$, and $t+16$, consider the following approximations of NFSR

$$\begin{array}{l} r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_t[15]_{(20)} \oplus \cdots \oplus r_{t+1}[16]_{(0)} = 0 \\ r_{t+1}[0]_{(10)} \oplus r_{t+1}[0]_{(6)} \oplus r_{t+1}[15]_{(20)} \oplus \cdots \oplus r_{t+2}[16]_{(0)} = 0 \\ r_{t+6}[0]_{(10)} \oplus r_{t+6}[0]_{(6)} \oplus r_{t+6}[15]_{(20)} \oplus \cdots \oplus r_{t+7}[16]_{(0)} = 0 \\ r_{t+13}[0]_{(10)} \oplus r_{t+13}[0]_{(6)} \oplus r_{t+13}[15]_{(20)} \oplus \cdots \oplus r_{t+14}[16]_{(0)} = 0 \\ r_{t+16}[0]_{(10)} \oplus r_{t+16}[0]_{(6)} \oplus r_{t+16}[15]_{(20)} \oplus \cdots \oplus r_{t+17}[16]_{(0)} = 0 \end{array} \tag{17}$$

Since $r_{t+p}[0] = r_t[p]$, we can rewrite the above system of equations (17) equivalently as follows:

$$\begin{array}{l} r_t[0]_{(10)} \oplus r_t[0]_{(6)} \oplus r_{t+15}[0]_{(20)} \oplus \cdots \oplus r_{t+17}[0]_{(0)} = 0 \\ r_t[1]_{(10)} \oplus r_t[1]_{(6)} \oplus r_{t+15}[1]_{(20)} \oplus \cdots \oplus r_{t+17}[1]_{(0)} = 0 \\ r_t[6]_{(10)} \oplus r_t[6]_{(6)} \oplus r_{t+15}[6]_{(20)} \oplus \cdots \oplus r_{t+17}[6]_{(0)} = 0 \\ r_t[13]_{(10)} \oplus r_t[13]_{(6)} \oplus r_{t+15}[13]_{(20)} \oplus \cdots \oplus r_{t+17}[13]_{(0)} = 0 \\ r_t[16]_{(10)} \oplus r_t[16]_{(6)} \oplus r_{t+15}[16]_{(20)} \oplus \cdots \oplus r_{t+17}[16]_{(0)} = 0 \end{array} \tag{18}$$

Consider the columns of the above system of equations. Each column describes a single bit output of the filter (see Equation (16)), therefore the system (18) gives the following approximation:

$$\begin{array}{l} \mu_{t,(10)} \oplus \mu_{t,(6)} \oplus \mu_{t+15,(20)} \oplus \mu_{t+15,(16)} \oplus \mu_{t+15,(15)} \oplus \mu_{t,(13)} \\ \oplus \mu_{t+15,(23)} \oplus \mu_{t+4,(0)} \oplus \mu_{t+17,(0)} = K \end{array} \tag{19}$$

where $K = Konst_{(10)} \oplus Konst_{(6)} \oplus Konst_{(20)} \oplus Konst_{(16)} \oplus Konst_{(15)} \oplus Konst_{(13)} \oplus Konst_{(23)}$. Note that the bit K is constant (zero or one) during the session. Therefore, the bias of Approximation (19) is $(\mathbf{2^{-4.36}})^{\mathbf{5}} = \mathbf{2^{-21.8}}$.

6 The CP attack on NLS

In this Section, we describe the CP attack on the real NLS. The main idea is to find the best combination of approximations for both NFSR and NLF, while the state bits of the shift register vanish and the bias of the resulting approximation is as big as possible. We study the case for $Konst = 0$ at first and then, extend our attack to the case for $Konst \neq 0$. Since NLS allows only a non-zero most significant byte of $Konst$, the second case corresponds to the real NLS.

6.1 Case for $Konst = 0$

The linear approximations of $\alpha_{t,(0)}$ are given in Table 1. For the most effective distinguisher, we choose this time the third approximation from the table which is

$$\alpha_{t,(0)} = r_t[0]_{(12)} \oplus r_t[15]_{(22)} \tag{20}$$

and the bias of this approximation is $0.045410 = 2^{-4.46}$. By combining Equations (6) and (20), we have the following approximation

$$r_t[0]_{(12)} \oplus r_t[15]_{(22)} \oplus r_t[0]_{(13)} \oplus r_t[15]_{(23)} \oplus r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} = 0 \tag{21}$$

that has the same bias.
Let us now divide (21) into two parts : the least significant bits and the other bits, so we get

$$\begin{array}{l} l_1(r_t) = r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} \\ l_2(r_t) = r_t[0]_{(12)} \oplus r_t[0]_{(13)} \oplus r_t[15]_{(22)} \oplus r_t[15]_{(23)} \end{array} \tag{22}$$

Clearly, $l_1(r_t) \oplus l_2(r_t) = 0$ with the bias $2^{-4.46}$. Since $l_1(r_t)$ has only the least significant bit variables, we apply (12) which is true with the probability one. Then, we obtain

$$\begin{array}{l} l_1(r_t) = r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} \\ l_1(r_{t+1}) = r_{t+1}[4]_{(0)} \oplus r_{t+2}[16]_{(0)} \\ l_1(r_{t+6}) = r_{t+6}[4]_{(0)} \oplus r_{t+7}[16]_{(0)} \\ l_1(r_{t+13}) = r_{t+13}[4]_{(0)} \oplus r_{t+14}[16]_{(0)} \\ l_1(r_{t+16}) = r_{t+16}[4]_{(0)} \oplus r_{t+17}[16]_{(0)} \end{array} \tag{23}$$

If we add up all approximations of (23), then, by applying Equation (12), we can write

$$l_1(r_t) \oplus l_1(r_{t+1}) \oplus l_1(r_{t+6}) \oplus l_1(r_{t+13}) \oplus l_1(r_{t+16}) = \nu_{t+4,(0)} \oplus \nu_{t+17,(0)} \tag{24}$$

Now, we focus on $l_2(r_t)$ where the bit positions are 12, 13, 22, and 23, then,

$$\begin{array}{l} l_2(r_t) = r_t[0]_{(12)} \oplus r_t[0]_{(13)} \oplus r_t[15]_{(22)} \oplus r_t[15]_{(23)} \\ l_2(r_{t+1}) = r_{t+1}[0]_{(12)} \oplus r_{t+1}[0]_{(13)} \oplus r_{t+1}[15]_{(22)} \oplus r_{t+1}[15]_{(23)} \\ l_2(r_{t+6}) = r_{t+6}[0]_{(12)} \oplus r_{t+6}[0]_{(13)} \oplus r_{t+6}[15]_{(22)} \oplus r_{t+6}[15]_{(23)} \\ l_2(r_{t+13}) = r_{t+13}[0]_{(12)} \oplus r_{t+13}[0]_{(13)} \oplus r_{t+13}[15]_{(22)} \oplus r_{t+13}[15]_{(23)} \\ l_2(r_{t+16}) = r_{t+16}[0]_{(12)} \oplus r_{t+16}[0]_{(13)} \oplus r_{t+16}[15]_{(22)} \oplus r_{t+16}[15]_{(23)} \end{array} \tag{25}$$

Since $r_{t+p}[0] = r_t[p]$, the above approximations can be presented as follows.

$$\begin{array}{l} l_2(r_t) = r_t[0]_{(12)} \oplus r_t[0]_{(13)} \oplus r_{t+15}[0]_{(22)} \oplus r_{t+15}[0]_{(23)} \\ l_2(r_{t+1}) = r_t[1]_{(12)} \oplus r_t[1]_{(13)} \oplus r_{t+15}[1]_{(22)} \oplus r_{t+15}[1]_{(23)} \\ l_2(r_{t+6}) = r_t[6]_{(12)} \oplus r_t[6]_{(13)} \oplus r_{t+15}[6]_{(22)} \oplus r_{t+15}[6]_{(23)} \\ l_2(r_{t+13}) = r_t[13]_{(12)} \oplus r_t[13]_{(13)} \oplus r_{t+15}[13]_{(22)} \oplus r_{t+15}[13]_{(23)} \\ l_2(r_{t+16}) = r_t[16]_{(12)} \oplus r_t[16]_{(13)} \oplus r_{t+15}[16]_{(22)} \oplus r_{t+15}[16]_{(23)} \end{array} \tag{26}$$

Recall the approximation (13) of NLF. If we combine (26) with (13), then we have the following approximation.

$$\begin{array}{l} l_2(r_t) \oplus l_2(r_{t+1}) \oplus l_2(r_{t+6}) \oplus l_2(r_{t+13}) \oplus l_2(r_{t+16}) = \\ \nu_{t,(12)} \oplus \nu_{t,(13)} \oplus \nu_{t+15,(22)} \oplus \nu_{t+15,(23)} \end{array} \tag{27}$$

By combining the approximations (24) and (27), we obtain the final approximation that defines our distinguisher, i.e.

$$\begin{aligned} & l_1(r_t) \oplus l_1(r_{t+1}) \oplus l_1(r_{t+6}) \oplus l_1(r_{t+13}) \oplus l_1(r_{t+16}) \\ & \oplus l_2(r_t) \oplus l_2(r_{t+1}) \oplus l_2(r_{t+6}) \oplus l_2(r_{t+13}) \oplus l_2(r_{t+16}) \\ & = \nu_{t,(12)} \oplus \nu_{t,(13)} \oplus \nu_{t+15,(22)} \oplus \nu_{t+15,(23)} \oplus \nu_{t+4,(0)} \oplus \nu_{t+17,(0)} \\ & = 0 \end{aligned} \tag{28}$$

The second part of the approximation can be computed from the output keystream that is observable to the adversary. As we use Approximation (21) five times and Approximation (13) twice, the bias of the approximation (28) is $(\mathbf{2^{-4.46}})^{\mathbf{5}} \cdot (\mathbf{2^{-2}})^{\mathbf{2}} = \mathbf{2^{-26.3}}$.

6.2 Case for $Konst \neq 0$

Since the word $Konst$ occurs in NFSR and NLF as a parameter, the biases of linear approximations of both $\alpha_{t,(0)}$ and NLF vary with $Konst$. If we divide $Konst$ into two parts as $Konst = (Konst_{(H)}, Konst_{(L)})$ where $Konst_{(H)} = (Konst_{(31)}, \ldots, Konst_{(24)})$, and $Konst_{(L)} = (Konst_{(23)}, \ldots, Konst_{(0)})$, then, linear approximations of $\alpha_{t,(0)}$ mainly depend on $Konst_{(H)}$ and those of NLF depend on $Konst_{(L)}$.

Biases of $\alpha_{t,(0)}$ with non-zero $Konst_{(H)}$ Since the most significant 8 bits of $Konst$ mainly contribute to form of the bit $\alpha_{t,(0)}$, the bias of Approximation (20) fluctuates according to the 8-bit $Konst_{(H)}$. This relation is illustrated in Figure 2.

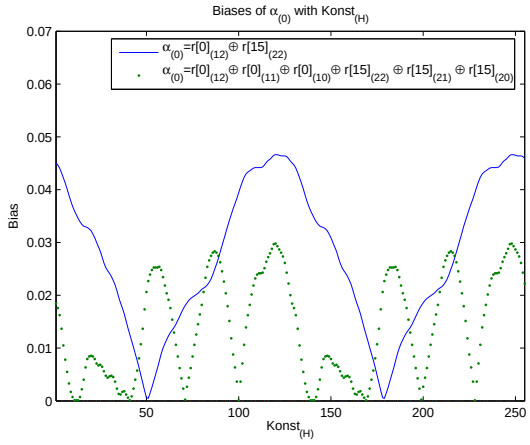

Fig. 2. Biases of approximations of $\alpha_{t,(0)}$ with $Konst_{(H)}$

From this figure, we can see that the bias of Approximation (20) becomes the smallest when $Konst_{(H)}$ is around 51 and 179 whereas the biggest when $Konst_{(H)}$ is around 127 and 255. The average bias of (20) with $Konst_{(H)}$ is $2^{-4.4}$.

Biases of NLF with $Konst_{(L)}$ Figure 3 displays the bias variation of Approximation (13) according to $Konst_{(L)}$ at $i = 13$. Note that the graph shows the bias distribution from 14 LSBs of $Konst_{(L)}$ (that is, 2^{14}) since the bits $Konst_{(23)}, \ldots, Konst_{(14)}$ have not effect on the bias for $i = 13$. We don't display the graph of Approximation (13) at $i = 23$ because the graph is similar to Figure 3 with only the slope changed by considering 24 bits of $Konst_{(L)}$. On the average, the bias of (13) for any $i > 0$ is 2^{-1}.

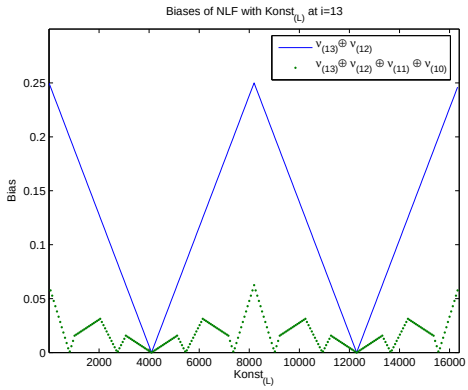

Fig. 3. Biases of NLF with $Konst_{(L)}$ at $i = 13$ and $i = 23$

6.3 Average bias of the distinguisher

According to Section 6.2, the average bias of the distinguisher (28) can be computed in a following way. Note that $Konst$ is expressed in hexadecimal.

1. Set $Konst = 01000000h$ (Note that non-zero $Konst_{(H)}$ is allowed in NLS.)
2. Find the bias ϵ_1 of Approximation (20) for NFSR.
3. Find the bias ϵ_2 of Approximation (13) for NLF.
4. Compute and store the bias ϵ of the distinguisher (28) by $\epsilon = \epsilon_{\mathbf{1}}^{\mathbf{5}} \cdot \epsilon_{\mathbf{2}}^{\mathbf{2}}$.
5. Increase $Konst$ by 1 and repeat Step 2,3 and 4 until $Konst = ffffffffh$.
6. Compute the average of ϵ.

In order to reduce the complexity of computing the average of ϵ, we assume that ϵ_1 is affected by only $Konst_{(H)}$, not by $Konst_{(L)}$ in Step 2. Then, ϵ_1 and ϵ_2 can be computed independently. Therefore, the above algorithm is amended as follows.

1. Set $Konst_{(H)} = 01h$
2. Find the bias ϵ_1 of Approximation (20) and store $\epsilon_1^* = \epsilon_1^5$.
3. Increase $Konst_{(H)}$ by 1 and repeat Step 2 until $Konst_{(H)} = ffh$.
4. Compute the average of ϵ_1^*, which is called $(\epsilon_1^*)_{avg}$.
5. Set $Konst_{(L)} = 000000h$
6. Find two biases of Approximation (13) at $i = 13$ and $i = 23$, which is called $\epsilon_{2,13}$ and $\epsilon_{2,23}$ respectively.
7. Store ϵ_2^* by calculating $\epsilon_2 = \epsilon_{2,13} \cdot \epsilon_{2,23}$.

8. Increase $Konst_{(L)}$ by 1 and repeat Step 6 and 7 until $Konst_{(L)} = 00ffffffh$.
9. Compute the average of ϵ_2^*, which is called $(\epsilon_2^*)_{avg}$.
10. Compute the average bias of the distinguisher (28) by multiplying $(\epsilon_{\mathbf{1}}^*)_{\mathbf{avg}}$ by $(\epsilon_{\mathbf{2}}^*)_{\mathbf{avg}}$.

In result, the distinguisher (28) has $(\epsilon_1^5)_{avg}$ of 2^{-24} and $(\epsilon_2^2)_{avg}$ of 2^{-6} respectively. Therefore, the average bias of distinguisher appears to be $2^{-24} \cdot 2^{-6} = 2^{-30}$.

6.4 The success rate of distinguishing attack

As mentioned in the earlier section, let us denote the bias of the approximation of $\alpha_{t,(0)}$ by ϵ_1, the bias of Approximation (13) at $i = 13$ by $\epsilon_{2,13}$ and the bias of Approximation (13) at $i = 23$ by $\epsilon_{2,23}$.
Since the specification of the NLS cipher allows the adversary to observe up to 2^{80} keystream words per one key/nonce pair, we assume that our attack is successful if the bias of distinguisher satisfies the following condition:

$$\epsilon_1^5 \cdot \epsilon_{2,13} \cdot \epsilon_{2,23} > 2^{-40}. \tag{29}$$

The experiments show that the bias of Distinguisher 28 satisfies the condition (29) on around 85.9% of $Konst$. See Figure 4.

7 Improving distinguishing attack by multiple distinguishers

In this section, we are going to reduce the unsuccessful portion of $Konst$ by considering multiple distinguishers, in particular, by multiple approximations of $\alpha_{(0)}$. Since the NLS produces 32-bit keystream word per a clock, the actual volume of data required is not increased by multiple distinguishers even though more computation is required.
The motivation is that Approximation (20) is not always best choice for the distinguisher on all the possible values of $Konst$. The bias of the distinguisher using Approximation (20) is very small for some values of $Konst_{(H)}$ (e.g. $Konst_{(H)} = 51$ or 179). In order to defeat this problem, let us choose the fourth approximation from Table 1. Then, we have

$$\alpha_{t,(0)} = r_t[0]_{(12)} \oplus r_t[0]_{(11)} \oplus r_t[0]_{(10)} \oplus r_t[15]_{(22)} \oplus r_t[15]_{(21)} \oplus r_t[15]_{(20)} \tag{30}$$

which has the smallest bias when $Konst_{(H)}$ is around $41, 139$ and 169 whereas the biggest when $Konst_{(H)}$ is around 57 and 185. The average bias of (30) is $2^{-6.2}$ when only absolute values are taken. See Figure 2.
Having this approximation, we build an approximation of NFSR as follows

$$\begin{aligned} &r_t[0]_{(10)} \oplus r_t[0]_{(11)} \oplus r_t[0]_{(12)} \oplus r_t[0]_{(13)} \oplus \oplus r_t[15]_{(20)} \oplus r_t[15]_{(21)} \oplus r_t[15]_{(22)} \oplus r_t[15]_{(23)} \\ &\oplus Konst_{(0)} \oplus r_t[4]_{(0)} \oplus r_{t+1}[16]_{(0)} = 0. \end{aligned} \tag{31}$$

Then, we can build a new distinguisher by combining Approximation (14) on NLF (we omit the detail process due to the similarity of Distinguisher (28)). In result, we have a following new distinguisher

$$\begin{aligned} &\nu_{t,(10)} \oplus \nu_{t,(11)} \oplus \nu_{t,(12)} \oplus \nu_{t,(13)} \oplus \nu_{t+15,(20)} \oplus \nu_{t+15,(21)} \oplus \nu_{t+15,(22)} \oplus \nu_{t+15,(23)} \\ &\oplus \nu_{t+4,(0)} \oplus \nu_{t+17,(0)} = 0. \end{aligned} \tag{32}$$

Distinguisher (32) has $(\epsilon_1^*)_{avg}$ of $2^{-27.8}$ and $(\epsilon_2^*)_{avg}$ of 2^{-10}. Therefore, the average bias of distinguisher appears to be $2^{-27.8} \cdot 2^{-10} = 2^{-37.8}$.

By observing two distinguishers together and selecting always the better bias among them, we improve the success rate of the distinguishing attack. The experiments show that the best bias between Distinguisher 28 and 32 satisfies the condition (29) on around 90.3% of $Konst$. See Figure 4.

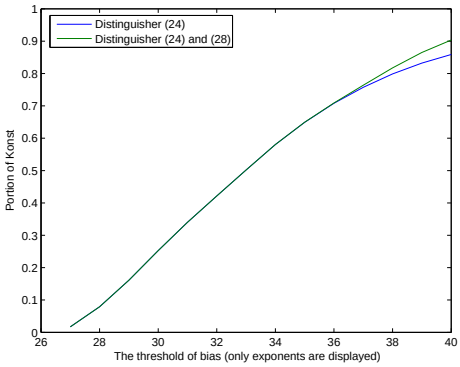

Fig. 4. The success rate of $Konst$ according to the bias

8 Conclusion

We presented a linear distinguishing attack on NLS by Crossword Puzzle attack method newly introduced in this paper. The bias of distinguisher appears to be 2^{-30} on the average so that NLS is distinguishable from a random function by observing 2^{60} keystream words. Even though there are a fraction of $Konst$ which requires the data complexity bigger than 2^{80}, we show that it is possible for attacker to reduce the fraction of $Konst$ by combining multiple distinguishers which have biases of less than 2^{-40} on the average.

Acknowledgment We are grateful to Philip Hawkes for invaluable comments, in particular, Section 6.3. We also grateful to anonymous referees of SASC 2006 for their very helpful comments. The second author acknowledges the support received from Australian Research Council (projects DP0451484 and DP0663452).

References

1. http://www.ecrypt.eu.org/stream/.
2. http://www.ecrypt.eu.org/stream/nls.html.
3. Don Coppersmith, Shai Halevi, and Charanjit Jutla. Cryptanalysis of stream ciphers with linear masking. Cryptology ePrint Archive, Report 2002/020, 2002. http://eprint.iacr.org/.