Attacks based on Conditional Correlations against the Nonlinear Filter Generator

Bernhard Löhlein

ITC Security
T-Systems Nova GmbH
Am Kavalleriesand 3, D-64295 Darmstadt, Germany
bernhard.loehlein@t-systems.com

Abstract. In this paper we extend the conditional correlation attack ([LCPP96]) against the nonlinear filter generator (NLFG) by introducing new conditions and generalisations and present two known-plaintext attacks, called hybrid correlation attack and concentration attack. The NLFG is a well known LFSR-based keystream generator which could be used as a basic building block in a synchronous stream cipher system. Both new attacks use methods from the conditional correlation attack and additional from fast correlation attacks to derive the unknown initial state of the LFSR of the NLFG. The basic principle of iteratively cumulating and updating conditional correlations for the NLFG was proposed in [Löh01] and for general combiners with memory in [GBM02]. With the hybrid correlation attack it is possible to successfully attack the NLFG by applying a fast correlation attack, even if the filter function f of the NLFG is highly nonlinear, e.g. the normalised nonlinearity $p_{e,f}$ is ≥ 0.45. The concentration attack maps all computed conditional correlations to $D - B$ unknown LFSR bits, where $D \geq k$ and $1 \leq B \leq k$ are parameters which can be chosen by the attacker, and k is the length of the LFSR of the NLFG. Even with low values of conditional correlations, it is possible to mount the hybrid correlation attack and the concentration attack successfully. This is not the case for the originally version of the conditional correlation attack ([LCPP96]) in a time lower than a full search over all possible initial states.

Key words: stream ciphers, keystream generator, nonlinear filter generator, NLFG, conditional correlation attack, fast correlation attacks

1 Introduction

The nonlinear filter generator (NLFG, Fig. 1 and left part of Fig. 3) consists of a linear feedback shift register (LFSR) of length k and a boolean function $f : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$, called filter function, whose n inputs are taken from some shift register stages Γ, called taps, to produce the GF(2) keystream sequence $\tilde{z} = z_0, z_1, z_2, \ldots$. In this paper all sequence elements are considered over the field GF(2) which consists of the two elements $\{0, 1\}$.

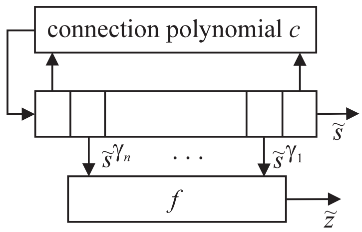

Fig. 1. The nonlinear filter generator with LFSR, consisting of the connection polynomial c and state vector $\underline{s}_t$, the taps $\Gamma = (\gamma_1, \ldots, \gamma_n)$ and the filter function f to produce the GF(2) keystream sequence $\tilde{z}$.

The NLFG can be used as a keystream generator itself or as a building block in a more complex shift register based system in cryptographic stream cipher applications. An example is the use of the nonlinear filter generator as a synchronous secret key encryption and decryption system in a communication system (see Fig. 2). In this scenario we suppose that the secret key K between two parties is used to initialise the stages $\underline{s}_0$ of the LFSR of the NLFG at time $t = 0$. The encryption is done by a bitwise XOR operation of the keystream $\tilde{z}$ and the message $\tilde{m}$ (plaintext) to the ciphertext sequence $\tilde{c}$. The encrypted sequence $\tilde{c}$ is sent to the receiver over an unsecure channel and is decrypted by a bitwise XOR operation of $\tilde{c}$ and $\tilde{z}$.

The LFSR of the NLFG has the connection polynomial

$$c(x) = x^k - \sum_{j=0}^{k-1} c_j x^j,$$

$c \in \mathrm{GF}(2)[x]$. This LFSR produces the GF(2) sequence $\tilde{s} = s_0, s_1, \ldots$, namely

$$s_{t+k} = \sum_{j=0}^{k-1} c_j s_{t+j}$$

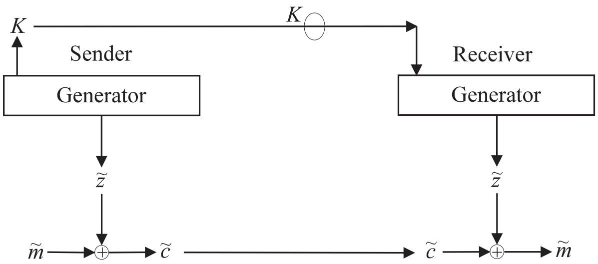

Fig. 2. The application of a keystream generator (i.e. NLFG) in a synchronous secret key encryption and decryption system.

for $t \geq 0$, when it is initialised with $\underline{s}_0 = (s_0, s_1, \ldots, s_{k-1})^T \in \mathrm{GF}(2)^k$, where ^{T} stands for transposition. Let the state of the LFSR at time $t \geq 0$ be $\underline{s}_t = (s_t, s_{t+1}, \ldots, s_{t+k-1})^T \in \mathrm{GF}(2)^k$. Any state $\underline{s}_t$ and sequence element s_t, $t \geq 0$, can be written as a linear combination

$$\underline{s}_t = \left(\underline{C}^t\right)^T \ \underline{s}_0,$$

and

$$s_t = \left(\underline{c}^{(t-k)}\right)^T \ \underline{s}_0$$

where $\underline{C}$ is the $k \times k$ companion matrix over GF(2) w.r.t. to the connection polynomial c and $\underline{c}^{(t-k)} \in \mathrm{GF}(2)^k$ is a vector which can be derived from one of the matrices $\underline{C}^{t-k+1}, \underline{C}^{t-k+2}, \ldots, \underline{C}^t$. The regular matrix $\underline{C}$ is given by

$$\underline{C} = \begin{pmatrix} 0 & 0 & \ldots 0 & c_0 \\ 1 & 0 & \vdots & c_1 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots\ 0 & c_{k-2} \\ 0 & \ldots & 0\ \ 1 & c_{k-1} \end{pmatrix} = \left(\underline{c}^{(1-k)}, \underline{c}^{(2-k)}, \ldots, \underline{c}^{(0)}\right)$$

and the k row vectors of $\underline{C}^t$ are then defined as

$$\underline{C}^t = \left(\underline{c}^{(t-k)}, \underline{c}^{(t-k+1)}, \ldots, \underline{c}^{(t-1)}\right).$$

The GF(2) keystream sequence $\tilde{z} = z_0, z_1, \ldots$ of the NLFG is generated by applying n stages (taps) $\Gamma = (\gamma_1, \gamma_2, \ldots, \gamma_n)$, $0 \leq \gamma_1 < \gamma_2 < \ldots \gamma_n < k$, from the LFSR as inputs to the filter function $f : \mathrm{GF}(2)^n \rightarrow \mathrm{GF}(2)$,

$$z_t = f(s_{t+\gamma_1}, s_{t+\gamma_2}, \ldots, s_{t+\gamma_n})$$

for $t \geq 0$. The span of the taps Γ is denoted by $M = \gamma_n - \gamma_1$. Usually c is chosen as a primitive polynomial so the period of the sequences $\tilde{s}$ and $\tilde{z}$ is $2^k - 1$ if f is balanced ([Sch99]).

In the process of designing a secure NLFG, all known attacks should be considered. The objective of a known-plaintext attack against the nonlinear filter generator is to determine the unknown initial state vector $\underline{s}_0$ by observing N keystream symbols $z_0, \ldots, z_{N-1}$ and assuming knowledge of the complete structure of the system, namely c, Γ, and f.

The paper is organised as follows: In section 2 the published attacks on the nonlinear filter generator are reviewed. The conditional correlation attack is described in section 3, and extended by introducing new conditional correlation coefficients and several generalisations concerning the taps Γ and the time pattern $T^{(m)}$. These are specialised versions of the coefficient defined in [LCPP96] (in our notation B_3) and used for the conditional correlation and our new attacks. In section 4 we present the hybrid correlation attack and in section 5 the concentration attack.

2 Properties and known attacks against the NLFG

In the past years, several properties and attacks on the nonlinear filter generator were published. With the Berlekamp-Massey algorithm it is possible to rapidly construct an equivalent LFSR which produces the sequence $\tilde{z}$ with the knowledge of $N = 2L(\tilde{z})$ keystream symbols in $O(N^2)$, where $L(\tilde{z})$ is the linear complexity of the sequence $\tilde{z}$. General lower and upper bounds for $L(\tilde{z})$ were derived in [Key76] and [FSCG95], i.e. $\binom{k}{2} \leq L(\tilde{z}) \leq \sum_{j=1}^{g} \binom{k}{j}$, where the lower bound is valid for c primitive and k prime and g is the algebraic degree of the filter function f. Better lower bounds for $L(\tilde{z})$ can be found in [Rue86] and [Sch99] for special cases.

The basic correlation attack against the nonlinear filter generator was published in 1985 by Siegenthaler ([Sie85]), where correlations between $\tilde{z}$ and linear transformations of $\tilde{s}$ are used to build an equivalent generator. This generator consists of $m \leq n$ LFSRs with connection polynomial c and a combining function $g : \mathrm{GF}(2)^m \rightarrow \mathrm{GF}(2)$. The drawback of this attack is the huge amount of time of $O(N^2)$, $N \approx 2^k$, needed for computing the necessary correlations and that the filter function f must have high correlation to an affine function. One consequence of the basic correlation attack is that the designer has to choose a highly nonlinear filter function for his NLFG. The nonlinearity N_f of a boolean function $f : \mathrm{GF}(2)^n \rightarrow \mathrm{GF}(2)$ is defined as $N_f = \min_l d(f, l)$, where $l : \mathrm{GF}(2)^n \rightarrow \mathrm{GF}(2)$ is an affine function, i.e. $l(\underline{x}) = \underline{w}^T\ \underline{x} + a$ with $\underline{w}, \underline{x} \in \mathrm{GF}(2)^n$, $a \in \mathrm{GF}(2)$, and $d(f, l)$ is the Hamming-distance between f and l. The Hamming-distance $d(f, l)$ between two boolean functions f and l with n inputs is the number of different outputs over all 2^n inputs:

$$d(f, l) := \sum_{\underline{x} \in \mathrm{GF}(2)^n} (f(\underline{x}) + l(\underline{x})),$$

where the sum is taken over GF(2).

For balanced boolean functions f with n inputs the following bound for N_f holds ([SZZ93]):

$$N_f \leq N_{max,bal}(n) = \begin{cases} 2^{n-1} - 2^{n/2-1} - 2, \text{if } n \text{ even}, \\ \lfloor\lfloor 2^{n-1} - 2^{n/2-1} \rfloor\rfloor, \text{if } n \text{ odd}, \end{cases}$$

where $y = \lfloor\lfloor x \rfloor\rfloor$ denotes the biggest even integer y with $y \leq x$. We consider the normalised nonlinearity $p_{e,f} = 2^{-n} N_f$ of a boolean function $f : \mathrm{GF}(2)^n \rightarrow \mathrm{GF}(2)$. We call a boolean function f highly nonlinear if $p_{e,f} \geq 0.45$. In literature there are many methods to construct highly nonlinear and balanced boolean functions with appropriate cryptographic properties.

In the following years, the concept of correlation attacks of Siegenthaler on LFSR-based keystream generators was improved by the basic fast correlation attack of Meier and Staffelbach ([MS89]). Recently, e.g. [CJS01], [JJ00], [CT00], [MFI01a] and [CJM02], more advanced decoding techniques have been proposed to mount a fast correlation attack. Their common method is to find low weight parity check polynomials of c and/or to apply an iterative decoding procedure to realise the attack. These fast correlation attacks can also be applied with minor modifications to the NLFG ([For90], [GSSD97], [JJ02]). Fast correlation attacks dedicated to the NLFG perform better on the NLFG than on combiner generators because they can exploit all nonzero correlation between the filter function and all linear functions simultaneously. A common disadvantage of these fast correlation attacks is the large number N of observed keystream symbols needed to perform a successful attack and the assumption that the filter function f is not highly nonlinear, i.e. they are only successful, if $p_{e,f} \leq 0.45$, and the computation complexity is lower than a full search.

In [Gol96], [GCD99], [GCD00] the special and the general inversion attacks were published and analysed. The first one only works for filter functions f which are linear-separable in the first or last variable, i.e. $f(x_1, \ldots, x_n) = x_1 + g(x_2, \ldots, x_n)$ or $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{n-1}) + x_n$, where $g : \mathrm{GF}(2)^{n-1} \rightarrow \mathrm{GF}(2)$ is an arbitrary boolean function. The general inversion attack is applicable to any filter function. Both attacks have time complexity of $O(2^{M-1})$ on average and are successful for highly nonlinear filter functions and small N. The inversion attack was improved for certain NLFG configurations in [GG02] to $O(2^{k-r-1})$, where r is the largest gap between LFSR cells, which have taps to the filter function or the connection polynomial c. The filter generator can be made resistant against the inversion attack if one chooses $\gamma_1 = 0$, $\gamma_n = k - 1$ and $\gcd(\gamma_1, \ldots, \gamma_n) = 1$. In [LBGZ01] ideas from the inversion attack and the conditional correlation attack are used to form a trellis based decoding procedure. Like the inversion attack, it has a time complexity of $O(2^{M-1})$ and is conceptually the same as the basic generalized inversion attack from [GCD99] and [GCD00].

In [GR94] and [Fil00] the decimation attack is proposed for LFSR based keystream generators. The idea is to consider a decimated sequence $\tilde{z}[d]$, with $\tilde{z}[d] = z_0, z_d, z_{2d}, \ldots$ of the observed keystream sequence $\tilde{z} = z_0, z_1, z_2, \ldots$. For the NLFG the decimation attack is applicable to a d with $1 \leq d \leq g$, $d|g$ and $g = \gcd(\gamma_1, \ldots, \gamma_n)$ ([Gol96]). For such a d, the decimated keystream sequence $\tilde{z}[d]$ can now be written as

$$z_{dt} = f(s_{dt+\gamma_1}, \ldots, s_{dt+\gamma_n}) = f(s'_{t+\gamma_1/d}, \ldots, s'_{t+\gamma_n/d})$$

for all $t \geq 0$. Thus, the decimated sequence $\tilde{z}[d]$ can be generated from the decimated LFSR sequence $\tilde{s}[d]$. If the decimated sequence $\tilde{s}[d]$ can be generated by a smaller LFSR with length $k' < k$, then all known attacks against the NLFG can be applied to this smaller NLFG. Properties of decimated sequences have been developed in [Rue86]. If k is chosen as prime or $1 \leq k \leq 89$, then it always holds that $k' = k$ and the decimation attack provides no further advantages.

In [BP00] $N = 2k$ keystream symbols are used to build an equation system with $2k$ nonlinear equations of the form $z_t = f(s_{t+\gamma_1}, \ldots, s_{t+\gamma_n})$ for $0 \leq t \leq 2^k - 1$ and $k + \gamma_n$ linear equations for the variables $s_k, s_{k+1}, \ldots, s_{2k+\gamma_n-1}$ from the linear recurrence relation of the LFSR. For any nonlinear equation the solution set is computed, i.e. the set of all tuples fulfil the nonlinear equation. Then the solution sets of two nonlinear equations with overlapping variables are iteratively merged and common values are removed from the merged set and substituted into the other equations. This process is called local reduction technique and is iterated until k independent variables from $\{s_0, s_1, \ldots, s_{2k+\gamma_n-1}\}$ have a solution or no further merging and substituting is possible. In the latter case, a tree-based search is done over the unsolved variables. The attack is only feasible for small values of n and there must be enough overlapping in the solution sets of the nonlinear equations.

While in the last years fast correlation attacks where the main subject of research on attacks against LFSR-based keystream generators, algebraic attacks on stream cipher systems get more attention. If an attacker observes the keystream bits $z_{t_1}, z_{t_2}, \ldots, z_{t_N}$ at time positions $t_1, \ldots, t_N$ he receives N equations $z_{t_i} = f(s_{t_i+\gamma_1}, \ldots, s_{t_i+\gamma_n})$. By substituting the sequence elements s_t, $t \geq k$, with the help of the linear relation to $s_0, \ldots, s_{k-1}$ a nonlinear system of N polynomial equations $z_{t_i} = f_i(s_0, \ldots, s_{k-1})$ in k variables can be obtained, where the algebraic degree of each boolean function f_i is $\leq g$ ([Bab01]). The problem of solving a nonlinear system of multivariate equations is NP-hard even if all the equations are quadratic and the underlying field is GF(2) ([GJ79]). There will be at most $V = \sum_{j=1}^{g} \binom{k}{j}$ distinct monomials of the variables $s_0, \ldots, s_{k-1}$ in the nonlinear system of equations. By substituting any distinct monomial by a new variable a linear system with V variables can be received. If $V \leq N$, then this new system of linear equations can be solved in $O(V^w)$ operations, where $w = 2.3788$ if the matrix inversion algorithm from [CW90] is used, $w = \log_2(7) \approx 2.807$ with Strassen's algorithm ([Str69]), and $w = 3$ with the Gaussian reduction algorithm. Alternativly, if $k \approx N$, then the algebraic method XL ([CKPS00]) or its improved versions XL' or XL2 ([CP03]) can be used to solve the nonlinear system of quadratic equations. The authors of [CKPS00] have evidence that the XL algorithm can solve randomly generated systems of polynomial equations in subexponential time (w.r.t. k) when N exceeds k by a number that increases slowly with k. In [Cou02] the XL method was extended to polynomial equations of arbitrary algebraic degree and applied to stream ciphers[1]. The attack against the NLFG with the extended version of XL can be mounted in two directions:

1. The algebraic degree g of f is low ($\epsilon = 0$).
2. The algebraic degree g of f is high and the function f (f_i) is approximated by a function h (h_i), which has a low algebraic degree, with high probability $1 - \epsilon$. In this case the system of equations consists of $z_{t_i} = h_i(s_0, \ldots, s_{k-1})$ for $0 \leq i \leq N$.

The overall complexity for solving the nonlinear system is then approximately

$$\binom{k}{k/N^{1/g}}^w (1-\epsilon)^{-N}.$$

In the view of the XL attack, the filter function f should be not only highly nonlinear, but also have a large distance to approximations of low algebraic degree.

[1] N. Courtois et al. have announced further articles on algebraic attacks on stream ciphers, see `http://www.cryptosystem.net/stream/`.

In [BS00] tradeoff attacks (Time/Memory/Data tradeoffs) are developed and analysed for synchronous stream cipher systems. Two main variants of a tradeoff attack are discovered, which differ in the generation of special states: Rivest and BSW sampling. Special states generate output prefixes of a keystream generator with a predefined bit pattern of l bit length. In the case of BSW sampling the special states of the keystream generator can be enumerated in an efficient way, i.e. in polynomial time. Both variants have a tradeoff relationship given by

$$TS^2N^2 = Z^2,$$

where T is time complexity in the realtime phase of the attack (i.e. one time unit equals the generation of $O(\log_2(Z))$ bit keystream), S represents the storage requirement (typically access on a hard disk), N is the amount of keystream, and Z is the size of the state space of the stream cipher, i.e. $Z = 2^k$ in the case of the NLFG. The time for preprocessing is $P = Z/N$ and the number of disk operations in the realtime phase is then given by $T_{disk} = \sqrt{T}$ in the case of Rivest sampling and $T_{disk} = \sqrt{T}2^{-l}$ for BSW sampling. In the case of Rivest sampling $D^2 \leq T \leq N$ is allowed and $(2^{-l}D)^2 \leq T \leq N$ for BSW sampling. Such if a keystream generator allows efficient BSW sampling for an appropriate $l > 0$ the number of disk operations and the lower bound on T can be further reduced. Typical values could be $P = Z^{2/3}$, $T = Z^{2/3}$, $S = Z^{1/3}$, $N = Z^{1/3}$.

In [Löh01] the linear transformation attack against the NLFG was presented. It transforms the given NLFG with a linear and regular transformation $\underline{A}^{(u)}$ in an equivalent NLFG with the same connection polynomial c, a filter function g : $\mathrm{GF}(2)^m \to \mathrm{GF}(2)$ with $m \geq n$ inputs and taps $\Gamma' = (\gamma'_1, \ldots, \gamma'_m)$. The objective was to find a transformation $\underline{A}^{(u)}$ so that the parameters of g or the span $\gamma'_m - \gamma'_1$ are better suited for one of the above described attacks. In a theoretical analysis and a case study it was shown that the probability of existence of an equivalent NLFG, which is better suited for an attack, is negligible for reasonable values of k and n ([Löh01]).

Also the BDD attack (binary decision diagrams, [Kra01], [Kra02]), which can be applied to several classes of LFSR based keystream generators, has to be considered in the design and analysis of the NLFG.

In the next section the conditional correlation attack, including our improvements, is described. In appendix A examples for the notations and definitions can be found.

3 The conditional correlation attack

The conditional correlation attack was described in [LCPP96] and the basic concept of searching for optimum correlations between a keystream sequence $\tilde{z}$ and a LFSR sequence $\tilde{s}$ by the augmented function F^m of f was presented in [And95]. The main idea of the conditional correlation attack is to study the statistical dependence between a fixed and known output of length m of the nonlinear filter generator and its corresponding input by analysing the augmented function

F^m. It turns out that some of these correlations can be much larger than what one could expect from the unconditional correlations.

3.1 Notations

For arbitrary taps $\Gamma = (\gamma_1, \ldots, \gamma_n)$, $0 \leq \gamma_1 < \ldots < \gamma_n < k$, and $t \geq 0$ let $\Gamma(t) = (t + \gamma_1, \ldots, t + \gamma_n)$ and for an output pattern $T^{(m)} = (t_1, \ldots, t_m)$, t_i integers, $\Gamma(T^{(m)}) = (i_1, \ldots, i_{M'})$, $0 \leq i_1 < \ldots < i_{M'}$, is defined as the tupel of $\{\gamma_r + t_s : 1 \leq r \leq n, 1 \leq s \leq m\}$ with cardinality $M' = |\Gamma(T^{(m)})|$. For an arbitrary tupel $D = (d_1, \ldots, d_l)$, sequence $\tilde{s}$, and $t \geq 0$ the vector $\underline{s}_t(D)$ is given by $\underline{s}_t(D) = (s_{t+d_1}, \ldots, s_{t+d_l})$. The dependence between the M' input symbols $\underline{s}_t(\Gamma(T^{(m)})) = (s_{t+i_1}, \ldots, s_{t+i_{M'}})^T$ and the corresponding m outputs at time t_1, $\ldots$, t_m by applying f is described by the augmented function $F^m : \mathrm{GF}(2)^{M'} \rightarrow \mathrm{GF}(2)^m$ of f, where

$$F^m(\underline{s}_t(\Gamma(T^{(m)}))) := (f(s_t(\Gamma(t_1))), \ldots, f(s_t(\Gamma(t_m))))^T.$$

If the index t is omitted in the above definitions it will be set to zero.

In the original paper ([LCPP96]) only the special case $\Gamma = (0, \ldots, n-1)$ and $T^{(m)} = (0, \ldots, m-1)$ is considered. With an optimal time pattern $T^{(m)}$ the effort for computing the conditional correlations is minimised and the values of them are maximised.

3.2 Conditional correlation coefficients

The main purpose of the conditional correlation attack is to find linear conditions B_l on the vector $\underline{s}(\Gamma(T^{(m)}))$ such that the conditional correlation coefficients

$$\lambda_l(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), .) := \left| P\left(\text{Cond. } B_l \text{ on } \underline{s}(\Gamma(T^{(m)})) | B\right) - 0.5 \right|$$

are near 0.5. The vector $\underline{s}(\Gamma(T^{(m)}))$ in the computation of the above probability is regarded as a vector of independent bits. The abbreviation B stands for the condition $F^m(\underline{s}(\Gamma(T^{(m)}))) = \underline{y}(T^{(m)}) \in \mathrm{GF}(2)^m$. The symbol "." on the left hand side of the above equation is a wildcard for arguments which depend on the condition B_l.

In our extensions of the conditional attack we define the following four linear conditions B_l , $1 \leq l \leq 4$, on the vector $\underline{s}(\Gamma(T^{(m)}))$, where $i \neq i'$ and $i, i' \in \Gamma(T^{(m)})$.

1. B_1: $s_i = 0$,
2. B_2: $s_i = s_{i'}$,
3. B_3: $\underline{s}(\Gamma(T^{(m)}))^T \ \underline{d} = 0$, with $\underline{d} = (d_{i_1}, d_{i_2}, \ldots, d_{i_{M'}})^T \neq \underline{0}$ and
4. B_4: $\underline{s}(\Gamma(T^{(m)})) = \underline{e}$, with $\underline{e} = (e_{i_1}, \ldots, e_{i_{M'}})^T$.

Note that $\underline{d} \in \mathrm{GF}(2)^{M'}$ and the components of the vector $\underline{e}$ are elements in $\mathrm{GF}(2) \cup \{*\}$, where the symbol $*$ means that in the comparison in condition B_4

the corresponding value of the component of $\underline{s}(\Gamma(T^{(m)}))$ does not count. Note that condition B_1 is a special case of B_3 and B_4, and B_3 is a generalisation of B_2. Currently there is no mathematical method to find high correlations. The only way for finding them is to perform a full search. This is the main reason for the introduction of B_1 and B_2 is that the full search may be feasible for this coefficients and that they are needed in the hybrid correlation attack and the concentration attack. The time complexities $C_l(m)$ for computing a full set of coefficients for condition B_l, $1 \leq l \leq 4$, are: $C_1(m) = M'2^{M'}$, $C_2(m) = (M'-1)/2M'2^{M'}$, $C_3(m) = M'2^{2M'}$ and $C_4(m) = M'2^{M'}3^{M'}$.

In the original description of the conditional correlation attack in [LCPP96] only the condition B_3 has been considered and analysed. In [Löh00] it is shown that for some classes of filter functions the conditional correlation coefficients for B_1 and B_2 can also be used to perform a conditional correlation attack, which have a significant lower time and space complexity as B_3 and B_4. From the conditions B_l, the appropriate conditional probabilities and correlation coefficients can be specified as follows.

1. B_1 and $i \in \Gamma(T^{(m)})$:
   $p_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i) := P(s_i = 0|B)$
   $\lambda_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i) := |P(s_i = 0|B) - 0.5|$
2. B_2, $i, i' \in \Gamma(T^{(m)})$ and $i \neq i'$:
   $p_2(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i, i') := P(s_i = s_{i'}|B)$
   $\lambda_2(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i, i') := |P(s_i = s_{i'}|B) - 0.5|$
3. B_3 and $\underline{d} \neq \underline{0}$:
   $p_3(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{d}) := P(\underline{s}(\Gamma(T^{(m)}))^T\, \underline{d} = 0|B)$
   $\lambda_3(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{d}) := |P(\underline{s}(\Gamma(T^{(m)}))^T\, \underline{d} = 0|B) - 0.5|$
4. B_4:
   $p_4(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{e}) := P(\underline{s}(\Gamma(T^{(m)})) = \underline{e}|B)$

   $$\lambda_4(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{e}) := \begin{cases} 0, \text{if } w^*(\underline{e}) = 0, \\ 0, \text{if } w^*(\underline{e}) \geq 1 \ \wedge p_4(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{e}) \leq 0.5, \\ p_4(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), \underline{e}) - 0.5, \text{else}, \end{cases}$$

   where $w^*(\underline{e})$ is the number of zeros and ones in vector $\underline{e}$.

The main task of an attacker is to find conditional correlation coefficients which are close to 0.5.

As an illustration, we show for condition B_1 how an attacker, who has observed N keystream symbols $z_0, \ldots, z_{N-1}$ and has computed the correlation coefficients $\lambda_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i) > 0$, can obtain a linear equation for the initial state vector $\underline{s}_0$. An occurrence of a vector $\underline{y}(T^{(m)}) = (y_{t_1}, y_{t_2}, \ldots, y_{t_m})^T$ is searched in $z_0, \ldots, z_{N-1}$, i.e. finding an index t with $z_{t+t_1} = y_{t_1}, \ldots, z_{t+t_m} = y_{t_m}$, with a high conditional correlation coefficient $\lambda_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i)$. A linear equation can be obtained as follows: If $p = p_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), i) > 0.5$, the linear equation

$$s_{t+i} = 0 = \left(\underline{c}^{(t+i-k)}\right)^T \underline{s}_0,$$

holds with probability p, and for $p < 0.5$

$$s_{t+i} = 1 = \left(\underline{c}^{(t+i-k)}\right)^T \underline{s}_0$$

holds with probability $1-p$. For the other conditions the procedure is similar. With k independent linear equations of the above form a regular system of linear equations can be obtained, which can be solved for the unknown initial state vector $\underline{s}_0$. We note here that for the conditions B_1, B_2 and B_3 exactly one equation and for condition B_4 $w^*(\underline{e})$ equations can be established simultaneously.

Theorem 1. *Let Γ and $T^{(m)}$ be arbitrary, $g : \mathrm{GF}(2)^{n-2} \to \mathrm{GF}(2)$ be an arbitrary boolean function, $f : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$ defined as $f(x_1, \ldots, x_n) = x_1 + g(x_2, \ldots, x_{n-1}) + x_n$, then it holds*

$$\lambda_l(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), .) = 0$$

for all $\underline{y}(T^{(m)}) \in \mathrm{GF}(2)^m$ and $l = 1, 4$.

Proof. [Löh00] or [Löh01] □

This is currently the only known class of boolean functions with this property. If Γ is considered as a set, and if it is a (n, e) positive difference set, and f is a correlation immune boolean function of order d, then it holds that $\lambda_1(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), .) = 0$ for $1 \le m \le \lfloor d/e \rfloor + 1$ and all $T^{(m)}$ ([Gol96]).

3.3 The algorithm for the conditional correlation attack

Assuming that an attacker knows the components of the NLFG, i.e. c, f, and Γ, and has observed N keystream symbols $z_0, \ldots, z_{N-1}$. For choosing appropriate m, $0 < \lambda_{min} \le 0.5$ and $L \subset \{1, 2, 3, 4\}$, the following steps have to be performed:

Step 1 Determine the set $T^{(m)} = (t_1, t_2, \ldots, t_m)$, so that

$$|\Gamma(T^{(m)})| = \min_{T'^{(m)}} |\Gamma(T'^{(m)})|, \tag{1}$$

where $T'^{(m)} = (t'_1, \ldots, t'_m)$.

Step 2 Compute

$$D = \{(l, \underline{y}(T^{(m)}), .) : \lambda_l(f, \Gamma, T^{(m)}, \underline{y}(T^{(m)}), .) \ge \lambda_{min}\},$$

for $l \in L$.

Step 3 Search in $z_0, \ldots, z_{N-1}$ for an occurrence of $\underline{y}(T^{(m)})$ with $(l, \underline{y}(T^{(m)}), .) \in D$.

Step 4 Obtain $k' \ge k$ linear equations for the initial state vector $\underline{s}_0$ from the tuples $(l, \underline{y}(T^{(m)}), .)$ found in step 3.

Step 5 Choose randomly k linear independent equations from the k' in step 4 and solve the system of linear equations for $\underline{s}_0$.

Step 6 Test possible solutions found in step 5 to see whether they produce the correct observed keystream sequence $z_0, \ldots, z_{N-1}$. If yes, then terminate, else go to step 5.

The steps 1 and 2 can be performed in a pre-computation stage, independent of the observed keystream. In step 4 the required matrices $\underline{C}^t$ or vectors $\underline{c}^{(t)}$ for forming the linear equations can be efficiently computed by a caching procedure ([Löh01]).

3.4 Discussion and drawbacks

The computational effort of the pre-computations (step 2) in the conditional correlation attack is at least exponential in M'. If Γ is a (full positive) difference set ([Gol96]) or is optimised w.r.t. to equ. (1) ([Löh01]), then M' is lower bounded by

$$M' \geq nm - \frac{m(m-1)}{2}$$

for any $T^{(m)}$ such that small values of m are feasible in an attack. In the attacking phase of the conditional correlation attack the expected number of rounds (step 5) is asymptotically given by $(\lambda_{min} + 0.5)^{-k/g}$ ([LCPP96], [Löh01]), where g is the number of equations which can be obtained for one conditional correlation coefficient in the set D. The value of g is 1 for the conditions B_1, B_2 and B_3 and $w^*(\underline{e})$ for B_4. In each round a linear equation system has to be solved with a cost of k^3 operations in GF(2), such that the computational runing time is $O(k^3(\lambda_{min} + 0.5)^{-k/g})$ in the attacking phase.

If m is small, then the values of the conditional correlation coefficients will be very small and near 0 for $n \geq 10$ and the expected number of rounds will be higher than the computational efforts for a full search or a fast correlation attack for large k. This drawback can be avoided with the hybrid correlation attack and the concentration attack presented in the next sections.

4 The hybrid correlation attack

The fast correlation attacks against the NLFG have the disadvantage that they are only successful up to a certain nonlinearity of the filter function f, i.e. $p_{e,f}$ should be lower than 0.45. Even for $n = 10$, balanced filter functions with good cryptographic properties exist, which exceed this nonlinearity bound. In addition, the conditional correlation attack could fail or have a higher running time than other attacks, if only a few conditional correlation coefficients with high values (≥ 0.8) are availible. In this section we introduce the hybrid correlation attack which utilises ideas from conditional correlation attacks (phase 1 and 2) and fast correlation attacks (phase 3) to overcome the mentioned drawbacks. In contrast to the conditional correlation attack, where only one coefficient is used for a linear equation, the hybrid correlation attack uses at most M' coefficients to make a first estimate of a sequence element in phase 1 (condition B_1). The basic principle of iteratively updating conditional correlations for the NLFG was proposed in [Löh01] and for general combiners with memory in [GBM02].

In the next subsections we will give an overview and a detailed description of the hybrid correlation attack.

4.1 Overview

The hybrid correlation attack is divided into three computational phases, where the second phase is optional. First we will give an overview of all involved phases.

Phase 1 In the first phase conditional correlation coefficients are computed for the conditions B_1 and/or B_4. They are used to make a first error correction, e.g. soft-decision decoding ([HOP96]), on the keystream sequence $\tilde{z}$, to derive the intermediate sequence $\tilde{u}$ and log-likelihood values $L(t) \in \mathbb{R}$ for every sequence element s_t, $0 \leq t \leq N-1$. We give a short introduction in the log-likelihood algebra in appendix C. Therefore, $L(t)$ is a abbreviation for $L(S_t)$, where $S_t \in \mathrm{GF}(2)$ is the binary random variable for the event of the LFSR sequence element s_t. If $L(t) > 0$ then the event $s_t = 0$ is more likely than $s_t = 1$. The absolute value of $L(t)$ is a measure of prediction. Phase 1 is mandatory.

Phase 2 In the second phase the attacker has the additional possibility to use conditional correlations (conditions B_2 and/or B_3) to further improve the correction. In this phase only conditional correlation coefficients with a value greater than 0.45 should be applied. The values $L(t)$ of the first phase are transformed to $L'(t) \in \mathbb{R}$ and the resulting sequence will be $\tilde{r}$. If phase 2 is omitted, we set $L'(t) := L(t)$ and $\tilde{r} := \tilde{u}$. The sequences $\tilde{u}$ and $\tilde{r}$ are an estimate for the unknown $\tilde{s}$. Note that phase 2 is optional.

Phase 3 In the third phase the attacker has the choice between applying a fast correlation attack or a majority information set procedure:

1. The attacker can apply a fast correlation attack on the sequence $\tilde{r}$. In contrast to the application of fast correlation attacks on the sequence $\tilde{z}$, it doesn't converge to a linear transformation of $\tilde{s}$, but directly to $\tilde{s}$. We recommend [CT00] as fast correlation attack because the log-likelihood values $L'(t)$ from phase 2 can be utilised as input to the decoding procedure which is based on low-weight parity checks of c. Any other fast correlation attack for LFSR-based keystream generators is also suitable.
2. With a majority decoding rule the most reliable log-likelihood values are selected to form an information set to recover the initial state of the LFSR.

The underlying error model of the hybrid correlation attack is compared with that of the fast correlation attacks in Fig. 3.

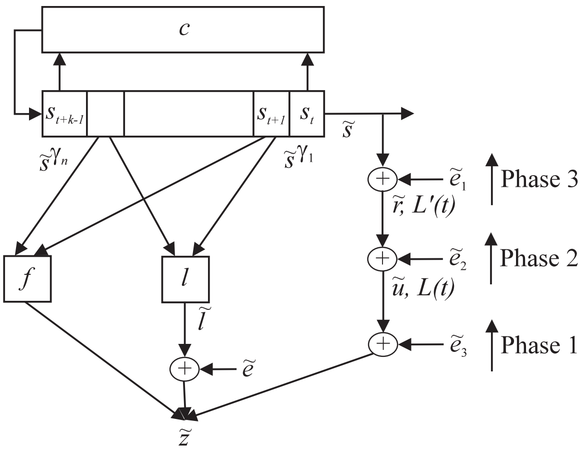

Fig. 3. Comparison of the error models of the fast correlation attacks (middle, l : $\mathrm{GF}(2)^n \to \mathrm{GF}(2)$ is a linear transformation) and the hybrid correlation attack (right). The normal operation of the NLFG is displayed on the left side.

4.2 Details

Phase 1 In the first phase conditional correlations on B_1 and/or B_4 have to be applied on the observed keystream sequence $\tilde{z}$ to obtain log-likelihood values $L(t)$ and an intermediate sequence $\tilde{u}$. Because of lack of space we only describe the case that coefficients have been computed for condition B_1 and given m and $T^{(m)} = (t_1, \ldots, t_m)$. The case for condition B_4 is straight forward. First we set $L(t) := 0$ for $0 \le t \le N-1$.

For $\min(\Gamma(T^{(m)})) - t_m \le t \le N + \max(\Gamma(T^{(m)})) - t_m - 1$ let us define the vectors $\underline{y}_{t,i}(T^{(m)})$, $i \in \Gamma(T^{(m)})$, as

$$\underline{y}_{t,i}(T^{(m)}) = (y_{t,i,t_1}, y_{t,i,t_2}, \ldots, y_{t,i,t_m})^T := (z_{t+t_1-i}, z_{t+t_2-i}, \ldots, z_{t+t_m-i})^T.$$

For every $i \in \Gamma(T^{(m)})$ we have a probability $p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)$ related to a conditional correlation for which we can associate a log-likelihood value

$$\ln\left(\frac{p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)}{1 - p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)}\right),$$

which contributes to $L(t)$:

$$L(t) := \sum_{i \in \Gamma(T^{(m)})} \ln\left(\frac{p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)}{1 - p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)}\right).$$

If a $p_1(f, \Gamma, T^{(m)}, \underline{y}_{t,i}(T^{(m)}), i)$ is 0 resp. 1, we set $L(t) := \infty$ resp. $L(t) := -\infty$.

With the values $L(t)$ we can compute the intermediate sequence $\tilde{u}$ as follows:

$$u_t := \begin{cases} 0, \text{if } \ L(t) > 0, \\ 1, \text{if } \ L(t) < 0, \\ 0 \ \text{ or } \ 1, \text{if } \ L(t) = 0. \end{cases}$$

The computational effort of phase 1, without the pre-computations for the coefficients, is only $O(NM')$.

Phase 2 In the second phase the attacker has the option to use conditional correlations w.r.t. B_2 or B_3, whose correlation should be higher than $\lambda_{min,2} \geq 0.45$ or $\lambda_{min,3} \geq 0.45$, to further correct the sequence $\tilde{u}$. Suppose that after the first phase the values $L(t)$, $0 \leq t \leq N$, are computed. First, the values $L'(t)$ have to be initialised: $L'(t) := L(t)$ for $0 \leq t \leq N-1$.

For $0 \leq t \leq N - t_m - 1$ we set

$$\underline{y}_t(T^{(m)}) = (y_{t,t_1}, y_{t,t_2}, \ldots, y_{t,t_m})^T := (z_{t+t_1}, z_{t+t_2}, \ldots, z_{t+t_m})^T.$$

We first describe the case for condition B_2 and then the general case for B_3. For every $i, i' \in \Gamma(T^{(m)})$ and $i \neq i'$ with

$$\lambda_2(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), i, i') \geq \lambda_{min,2}$$

we receive the equation

$$s_{t+i} = s_{t+i'},$$

which holds with probability $p_2(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), i, i')$.

Then we define

$$g := \begin{cases} 0, \text{if } \ p_2(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), i, i') \geq 0.5, \\ 1, \text{else.} \end{cases}$$

and update the log-likelihood values for $t+i$ and $t+i'$:

$$\begin{aligned} L'(t+i) &:= L'(t+i) + (-1)^g \cdot L(t+i') \quad \text{and} \\ L'(t+i') &:= L'(t+i') + (-1)^g \cdot L(t+i). \end{aligned}$$

If we have correlations for condition B_3, with

$$\lambda_3(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), \underline{d}) \geq \lambda_{min,3},$$

$\underline{d} = (d_{i_1}, d_{i_2}, \ldots, d_{i_{M'}})^T$, and $w(\underline{d}) \geq 2$, we set

$$g := \begin{cases} 0, \text{if } \ p_3(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), \underline{d}) \geq 0.5, \\ 1, \text{else.} \end{cases}$$

From the conditional correlation coefficient we receive the equation

$$\sum_{j=1}^{M'} d_{i_j} s_{t+i_j} = 0$$

which holds with probability $p_3(f, \Gamma, T^{(m)}, \underline{y}_t(T^{(m)}), \underline{d})$.

For any $1 \leq j \leq M'$ und $d_{i_j} \neq 0$ the value $L'(t+i_j)$ is updated as:

$$L'(t+i_j) := L'(t+i_j)+(-1)^g \cdot \prod_{\substack{1 \leq j' \leq M', j \neq j' \\ d_{i_{j'}} \neq 0}} \operatorname{sgn}(L(t+i_{j'})) \cdot \min_{\substack{1 \leq j' \leq M', j \neq j' \\ d_{i_{j'}} \neq 0}} (|L(t+i_{j'})|).$$

With the values $L'(t)$ we can compute the intermediate sequence $\tilde{r}$ as follows:

$$r_t := \begin{cases} 0, \text{if } \; L'(t) > 0, \\ 1, \text{if } \; L'(t) < 0, \\ 0 \; \text{ or } \; 1, \text{if } \; L'(t) = 0. \end{cases}$$

The computational effort of phase 2, without the pre-computations for the coefficients, is at most $O(NM'^2)$ if we only consider condition B_2, and at most $O(N2^{M'})$ if we also consider condition B_3. Note that in general only a few coefficients have values $\geq \lambda_{min,2}$ resp. $\geq \lambda_{min,3}$ so that the real effort is much less.

Phase 3 In the third phase the attacker has the choice between applying a fast correlation attack on the sequence $\tilde{r}$ or forming an information set of the sequence $\tilde{r}$. We describe two possibilities to perform the latter approach and refer the reader to the literature on fast correlation attacks for the former.

1. Choose the largest $k' > k$ values of $|L(t)|$, $0 \leq t \leq N-1$, randomly select k of them, and solve the corresponding linear system to obtain the initial state of the LFSR. If this state generates the observed sequence $\tilde{z}$, then the right solution is found. If not, try the next k randomly chosen values.
2. Alternatively, we can search for a time value t, for which

$$L'_{min}(t) := \min_{0 \leq j \leq k-1} |L'(t+j)|$$

is maximal. Solve the independent set $r_t, r_{t+1}, \ldots, r_{t+k-1}$ to an initial state $\underline{r}_0 = (r_0, r_1, \ldots, r_{k-1})^T$, and check, if it generates the sequence $\tilde{z}$.

4.3 Summary and discussion

We have implemented the conditional correlation attack and the hybrid correlation attack. An illustrative example for conditional correlation coefficients and the error correction of phase 1 of the hybrid correlation attack can be found in the appendix B. Even if the error correction of phase 1 and 2 is small, it may be good enough to let a fast correlation attack succeed in phase 3. Phase 1 and 2 of the hybrid correlation attack can be further improved by choosing not only one $T^{(m)}$, but different $T_1^{(m_1)}, T_2^{(m_2)}, \ldots$ with minimal intersection, small $m_i \geq 1$, $i \geq 1$ and $\sum_{i \geq 1} m_i > m$, and computing and applying the corresponding conditional correlation coefficients $\lambda_l(f, \Gamma, T_i^{(m_i)}, \underline{y}(T_i^{(m_i)}), .)$.

5 The concentration attack

The concentration attack combines methods from the hybrid correlation attack and the fast correlation attack described in [MFI01a]. The idea is to concentrate the log-likelihood values $L'(t)$ after phase 2 of the hybrid correlation attack to the elements at time B, $B+1$, $\ldots$, $D-1$, where $D \geq k$ and the first B bits (positions $0, 1, \ldots, B-1$) of the LFSR sequence $\tilde{s}$ are guessed. The concentration is done via check equations which are derived from the recurrence relation of the LFSR sequence. From the first D bits the most reliable are chosen to form an information set of size k. The concentration attack can be subdivided into the following steps.

Step 1 Perform phase 1 and 2 of the hybrid correlation attack and receive the log-likelihood values $L'(t)$.

Step 2 Choose the parameters $1 \leq B \leq k$, $D \geq k$ and $W \geq 1$ and compute the following consistency and check equations for $1 \leq w \leq W$:

2a Set of consistency equations:

$$E(w) := \left\{ (\{j_1, j_2, \ldots, j_w\}, \sum_{i=1}^{w} \underline{c}^{(j_i-k)}) : k \leq j_1 < \ldots < j_w \leq N-1, \sum_{i=1}^{w} \underline{c}^{(j_i-k)} = \underline{a}_k \right\}$$

2b Set of check equations for $B \leq t \leq k-1$:

$$E(t,w) := \left\{ (\{j_1, j_2, \ldots, j_w\}, \sum_{i=1}^{w} \underline{c}^{(j_i-k)}) : k \leq j_1 < \ldots < j_w \leq N-1, \sum_{i=1}^{w} \underline{c}^{(j_i-k)} = \underline{a}_t \right\}$$

2c Set of check equations for $k \leq t \leq D-1$:

$$E(t,w) := \left\{ (\{j_1, j_2, \ldots, j_w\}, \underline{c}^{(t-k)} + \sum_{i=1}^{w} \underline{c}^{(j_i-k)}) : D \leq j_1 < \ldots < j_w \leq N-1, \right.$$
$$\left. \underline{c}^{(t-k)} + \sum_{i=1}^{w} \underline{c}^{(j_i-k)} = \underline{a}_k \right\},$$

where $\underline{a}_t = (a_{t,0}, a_{t,1}, \ldots, a_{t,k-1})^T \in (\mathrm{GF}(2) \cup \{*\})^k$, with

$$a_{t,i} := \begin{cases} *, \text{if } 0 \leq i \leq B-1, \\ 1, \text{if } i = t \text{ and } t \neq k, \\ 0, \text{else.} \end{cases}$$

For $B \leq t \leq k-1$ we have

$$\underline{a}_t = (\underbrace{*, \ldots, *}_{B}, \underbrace{0, \ldots, 0}_{t-B}, 1, \underbrace{0, \ldots, 0}_{k-t-1})^T$$

and

$$\underline{a}_k = (\underbrace{*, \ldots, *}_{B}, \underbrace{0, \ldots, 0}_{k-B})^T.$$

Step 3 Choose appropriate large log-likelihood bounds L''_{min} and L'''_{min}.
Step 4 Guess the first B bits $s_0, s_1, \ldots, s_{B-1}$, set $L''(t) := L'(t)$ for $0 \leq t \leq N$, and evaluate the consistency and check equations $E(w)$ and $E(t,w)$ to receive the log-likelihood values $L'''(t)$ for $B \leq t \leq D-1$ (details below).
Step 5 Select the most reliable bits to form an information set of size k.
Step 6 Check if the current solution generates the observed keystream $\tilde{z}$. If not, go to step 4 and make the next guess, else the correct solution is found.

Note that the computation of the conditional correlation coefficients in step 1 and of the check equations in step 2 can be done in the pre-computations. Efficient algorithms can be derived from methods presented in [CJM02] and [Wag02].

We will now describe the tests and evaluations in step 4. Suppose we have guessed the bits $s_0, s_1, \ldots, s_{B-1}$. First it is tested if the guess is correct and is consistent:

1. Check if for $0 \leq t \leq B-1$ and $|L''(t)| \geq L''_{min}$, $(-1)^{s_t} = \mathrm{sgn}(L''(t))$ is valid.
2. For a test equation $(\{j_1, j_2, \ldots, j_w\}, (b_0, \ldots, b_{k-1})^T)$ in $E(w)$ we can establish the equation $\sum_{i=1}^{w} s_{j_i} = \sum_{i=0}^{B-1} b_i s_i =: b$. If $\min_{i=1}^{w}(|L''(j_i)|) \geq L''_{min}$, then we test, if $\prod_{i=1}^{w} \mathrm{sgn}(L''(j_i)) = (-1)^b$ holds.

If one of these conditions is not fulfilled, then the guess is not correct with high probability and we try the next guess.

If all conditions are valid, then we set $L''(j) = (-1)^b$, with $b = \sum_{i=0}^{B-1} b_i s_i$, for any $(\{j\}, (b_0, \ldots, b_{k-1})^T)$ in $E(1)$ and afterwards set $L'''(t) = L''(t)$ for $0 \leq t \leq D-1$.

We now briefly describe how to evaluate a check equation $E(t,w)$, $B \leq t \leq k-1$, in step 4 under a guess for $s_0, s_1, \ldots, s_{B-1}$. Suppose we have an equation $(\{j_1, j_2, \ldots, j_w\}, (b_0, \ldots, b_{k-1})^T)$ in $E(t,w)$. Note, that we have

$$b_i = \begin{cases} 0 \text{ or } 1, \text{if } 0 \leq i \leq B-1, \\ 1, \text{if } i = t, \\ 0, \text{else.} \end{cases}$$

Then the following equations can be established

$$\begin{aligned} \sum_{i=1}^{w} \underline{c}^{(j_i - k)} &= (b_1, \ldots, b_{B-1}, 0, \ldots, 0, 1, 0, \ldots, 0)^T \\ \underline{s}_0^T \sum_{i=1}^{w} \underline{c}^{(j_i - k)} &= \underline{s}_0^T \, (b_1, \ldots, b_{B-1}, 0, \ldots, 0, 1, 0, \ldots, 0)^T \\ \sum_{i=1}^{w} s_{j_i} &= \sum_{i=0}^{B-1} b_i s_i + s_t \end{aligned}$$

The last equation can be rewritten as

$$s_t = \sum_{i=1}^{w} s_{j_i} + \sum_{i=0}^{B-1} b_i s_i = \sum_{i=1}^{w} s_{j_i} + b$$

and the transformation of this equation in an update of the corresponding log-likelihood value is as follows

$$L'''(t) := L'''(t) + (-1)^b \cdot \prod_{i=1}^{w} \operatorname{sgn}(L''(j_i)) \cdot \min_{i=1}^{w}(|L''(j_i)|).$$

The evaluation of check equations in $E(t, w)$ for $k \leq t \leq D-1$ is straight forward.

The expected number of check equations $\mu(t, w)$ in $E(t, w)$ is given by $2^{B-k}\binom{N-k}{w}$ for $B \leq t \leq k-1$ and $2^{B-k}\binom{N-D}{w}$ for $k \leq t \leq D-1$ ([MFI01b]). Note that the values $\mu(t, w)$ does not depend on a particular connection polynomial c.

The number of check equations $\mu(t, w)$ can be increased significantly by a factor of 2^{t-B} if we evaluate the equations in step 4 in the order $B, B+1, \ldots, D-1$. The estimated values $L'''(t')$ can then included in the evaluation of $L''(t)$ for $t' \in \{B, B+1, \ldots, t-1\}$, if $|L'''(t')|$ exceeds a threshold L'''_{min}. The components of the vector $\underline{a}_t$ are then given by

$$a_{t,i} := \begin{cases} *, \text{if } 0 \leq i \leq t-1, \\ 1, \text{if } i = t \text{ and } t \neq k, \\ 0, \text{else.} \end{cases}$$

6 Conclusions

With the hybrid correlation attack against the NLFG we have proposed a general framework for embedding any fast correlation attack against LFSR-based keystream generators. The hybrid correlation attack reduces the real error probability introduced by the nonlinearity of the filter function f so that fast correlation attacks will be more successful and efficient.

Note that the application of the hybrid correlation and concentration attack is only possible if there are conditional correlations for conditions B_1 or B_4, which are greater than zero. With Theorem 1, a class of filter functions, is classified for which these conditional correlation coefficients are zero for any $m \geq 1$ and arbitrary $T^{(m)}$ and Γ. With the linear transformation attack (end of section 2) the attacker is able to find an equivalent NLFG for which the filter function doesn't have the structure as required in Theorem 1 and conditional correlation coefficients for B_1 and B_4 greater than zero may exist. But in this case, M' which determines the computational effort for computing the coefficients will in most cases be much larger than the original value.

The concentration attack uses the first and second phase of the hybrid correlation attack and then concentrates the computed log-likelihood values to $D-B$ unknowns with the help of a partial search over 2^B possible states.

A convergence analysis for the two new attacks and comparisons with other attacks against the NLFG are still missing. We think that these tasks can only be solved if more insight in the cumulated frequency values of the conditional correlation coefficients is gained.

7 Acknowledgements

I am grateful to my former colleagues at the University of Hagen and at T-Systems for support and helpful comments. Also thanks to anonymous referees for hints to improve the quality of the paper.

References

[And95] Ross J. Anderson. Searching for the optimum correlation attack. In Bart Preneel, editor, *Fast Software Encryption (FSE 1994), LNCS 1008*, pages 137–143. Springer-Verlag, 1995.

[Bab01] Steve Babbage. Cryptanalysis of LILI-128. Technical report, January 2001. `https://www.cosic.esat.kuleuven.ac.be/nessie/reports/`.

[BP00] S.S. Bedi and N.R. Pillai. Cryptanalysis of the nonlinear feedforward generator. In Bimal Roy and Eiji Okamoto, editors, *Progress in Cryptology, INDOCRYPT 2000, LNCS 1977*, pages 188–194. Springer-Verlag, 2000.

[BS00] Alex Biryukov and Adi Shamir. Cryptanalytic time/memory/data tradeoffs for stream ciphers. In Tsutomu Matsumoto, editor, *Advances in Cryptology, ASIACRYPT'00, Lecture Notes in Computer Science 1976*, pages 1–13. Springer-Verlag, 2000.

[CJM02] Phillippe Chose, Antoine Joux, and Michel Mitton. Fast correlation attacks: An algorithmic point of view. In Lars Knudsen, editor, *Advances in Cryptology, EUROCRYPT'02, LNCS 2332*, pages 209–221. Springer-Verlag, 2002.

[CJS01] Vladimor Chepyzhov, Thomas Johansson, and Bernard Smeets. A simple algorithm for fast correlation attacks on stream ciphers. In Bruce Schneier, editor, *Fast Software Encryption (FSE 2000), Proceedings, Lecture Notes in Computer Science 1978*, pages 181–195. Springer-Verlag, 2001.

[CKPS00] Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In Bart Preneel, editor, *Advances in Cryptology, EUROCRYPT'00, Lecture Notes in Computer Science 1807*, pages 392–407. Springer-Verlag, 2000.

[Cou02] Nicolas T. Courtois. Higher order correlation attacks, XL algorithm and cryptanalysis of Toyocrypt. Technical report, Cryptology ePrint Archive of the IACR, 2002. TR-2002-087, `http://eprint.iacr.org`.

[CP03] Nicolas T. Courtois and J. Patarin. About the XL algorithm over GF(2). In M. Joye, editor, *Progress in Cryptology - CT-RSA 2003, Lecture Notes in Computer Science 2612*, pages 140–156. Springer-Verlag, 2003.

[CT00] Anne Canteaut and Michael Trabbia. Improved fast correlation attacks using parity-check equations of weight 4 and 5. In Bart Preneel, editor, *Advances in Cryptology, EUROCRYPT'00, LNCS 1807*, pages 573–588. Springer-Verlag, 2000.

[CW90] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic programming. *Journal of Symbolic Computation*, 9:251–280, 1990.

[Fil00] Eric Filiol. Decimation attack on stream ciphers. In Bimal Roy and Eiji Okamoto, editors, *Progress in Cryptology, INDOCRYPT 2000, LNCS 1977*, pages 31–42. Springer-Verlag, 2000.

[For90] Réjane Forré. A fast correlation attack on nonlinearity feedforward filtered shift-register sequences. In Jean-Jacques Quisquater and Joos Vandewalle, editors, *Advances in Cryptology, EUROCRYPT'89, LNCS 434*, pages 586–595. Springer-Verlag, 1990.

[FSCG95] Amparo Fúster-Sabater and Pino Caballero-Gil. On the linear complexity of nonlinearly filtered pn-sequences. In Josef Pieprzyk and Reihaneh Safavi-Naini, editors, *Advances in Cryptology, ASIACRYPT'94, LNCS 917*, pages 80–90. Springer-Verlag, 1995.

[GBM02] Jovan Dj. Golić, Vittorio Bagini, and Guglielmo Morgari. Linear cryptanalysis of bluetooth stream cipher. In Lars Knudsen, editor, *Advances in Cryptology, EUROCRYPT'2002, Lecture Notes in Computer Science 2332*, pages 238–257. Springer-Verlag, 2002.

[GCD99] Jovan Dj. Golić, Andrew Clark, and Ed Dawson. Inversion attack and branching. In Josef Pieprzyk, Reihaneh Safavi-Naini, and Jennifer Seberry, editors, *Information Security and Privacy, Fourth Australasian Conference, ACISP'99, LNCS 1587*, pages 88–102. Springer-Verlag, 1999.

[GCD00] Jovan Dj. Golić, Andrew Clark, and Ed Dawson. Generalized inversion attack on nonlinear filter generators. *IEEE Transactions on Computers*, 49(10):1100–1109, October 2000.

[GG02] A. Górska and K. Górski. Improved inversion attacks on nonlinear filter generators. *Information Processing Letters*, 38(16):870–871, August 2002.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Interactability.* W.H. Freeman and Company, 1979.

[Gol96] Jovan Dj. Golić. On the security of nonlinear filter generators. In Dieter Gollmann, editor, *Fast Software Encryption (FSE 1996), LNCS 1039*, pages 173–187. Springer-Verlag, 1996.

[GR94] Richard A. Games and Joseph J. Rushanan. Blind synchronization of m-sequences with even span. In Tor Helleseth, editor, *Advances in Cryptology, EUROCRYPT'93, Lecture Notes in Computer Science 765*, pages 168–180. Springer-Verlag, 1994.

[GSSD97] Jovan Dj. Golić, Mahmoud Salmasizadeh, Leonie Ruth Simpson, and Ed Dawson. Fast correlation attacks on nonlinear filter generators. *Information Processing Letters*, 64(1):37–42, October 1997.

[HOP96] Joachim Hagenauer, Elke Offer, and Lutz Papke. Iterative decoding of binary block and convolutional codes. *IEEE Transactions on Information Theory*, 42(2):429–445, March 1996.

[JJ00] Fredrik Jönsson and Thomas Johansson. Theoretical analysis of a correlation attack based on convolutional codes. In Ezio Biglieri and Sergio Verdú, editors, *IEEE International Symposium on Information Theory 2000*, page 212, 2000.

[JJ02] Fredrik Jönsson and Thomas Johansson. A fast correlation attack on LILI-128. *Information Processing Letters*, 81(3):127–132, February 2002.

[Key76] Edwin L. Key. An analysis of the structure and complexity of nonlinear binary sequence generators. *IEEE Transactions on Information Theory*, 22(6):732–736, November 1976.

[Kra01] Matthias Krause. BDD-based cryptanalysis of keystream generators. Technical report, Cryptology ePrint Archive of the IACR, 2001. TR-2001-092, `http://eprint.iacr.org`.

[Kra02] Matthias Krause. BDD-based cryptanalysis of keystream generators. In Lars Knudsen, editor, *Advances in Cryptology, EUROCRYPT'2002, Lecture Notes in Computer Science 2332*, pages 222–237. Springer-Verlag, 2002.

[LBGZ01] Sabine Leveiller, Joseph Boutros, Philippe Guillot, and Gilles Zémor. Cryptanalysis of nonlinear filter generators with $\{0, 1\}$-metric Viterbi decoding. In Bahram Honary, editor, *Cryptography and Coding VIII, LNCS 2260*, pages 402–414. Springer-Verlag, 2001.

[LCPP96] Sangjin Lee, Seongtaek Chee, Sangjoon Park, and Sungmo Park. Conditional correlation attack on nonlinear filter generators. In Kwangjo Kim and Tsutomu Matsumoto, editors, *Advances in Cryptology, ASIACRYPT'96, LNCS 1163*, pages 360–367. Springer-Verlag, 1996.

[Löh00] Bernhard Löhlein. Analysis and modifications of the conditional correlation attack. In Joachim Hagenauer, Johannes Huber, and Peter Vary, editors, *Third ITG Conference Source and Channel Coding, ITG-Fachbericht 159*, pages 37–43. VDE Verlag, January 2000.

[Löh01] Bernhard Löhlein. *Design and analysis of cryptographic secure keystream generators for stream cipher encryption (in German).* PhD thesis, Faculty of Electrical and Information Engineering, University of Hagen, Germany, December 2001. Berichte aus der Kommunikationstechnik, Band 10, Shaker Verlag.

[MFI01a] Miodrag J. Mihaljeviċ, Marc P. C. Fossorier, and Hideki Imai. Fast correlation attack algorithm with list decoding and an application. In Hideki Imai, editor, *Fast Software Encryption (FSE 2001), LNCS 2355*, pages 196–210. Springer-Verlag, 2001.

[MFI01b] Miodrag J. Mihaljeviċ, Marc P. C. Fossorier, and Hideki Imai. On decoding techniques for cryptanalysis of certain encryption algorithms. *IEICE Transactions on Fundamentals*, E84-A(4):919–930, April 2001.

[MS89] Willi Meier and Othmar J. Staffelbach. Fast correlation attacks on stream ciphers. *Journal of Cryptology*, 1(3):159–176, 1989.

[Rue86] Rainer A. Rueppel. *Analysis and Design of Stream Ciphers.* Springer-Verlag, 1986.

[Sch99] Markus Schneider. *Methods of generating binary pseudo-random sequences for stream cipher encryption (in German).* PhD thesis, Faculty of Electrical Engineering, University of Hagen, Germany, September 1999. Berichte aus der Kommunikationstechnik, Band 4, Shaker Verlag.

[Sie85] Thomas Siegenthaler. Decrypthing a class of stream chiphers using chipertext only. *IEEE Transactions on Computers*, 34(1):81–85, January 1985.

[Str69] V. Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13:354–356, 1969.

[SZZ93] Jennifer Seberry, Xian-Mo Zhang, and Yuliang Zheng. Nonlinearly balanced boolean functions and their propagation characteristics. In Douglas R. Stinson, editor, *Advances in Cryptology, CRYPTO'93, LNCS 773*, pages 49–60. Springer-Verlag, 1993.

[Wag02] David Wagner. A generalized birthday problem (extended abstract). In Moti Yung, editor, *Advances in Cryptology, CRYPTO'2002, LNCS 2442*, pages 288–303. Springer-Verlag, 2002. Full version under `http://www.cs.berkeley.edu/~daw/papers/genbday.html`.

A Examples for notations and definitions

In this section we provide some examples on the notations and definitions introduced in section 3.

Consider a NLFG with the filter function $f : \mathrm{GF}(2)^3 \to \mathrm{GF}(2)$, $f(x_1, x_2, x_3) = x_1x_2 + x_2x_3 + x_3$, $\Gamma = (0, 3, 10)$ and an arbitrary LFSR of length $k = 40$. f has $n = 3$ inputs, is balanced, has an algebraic degree of 2, and a nonlinearity of $N_f = 2$ ($p_{e,f} = 0.25$).

Let $m = 2$ and the output time pattern $T^{(2)} = (t_1, t_2) = (0, 3)$, and so $T^{(1)} = (0)$. This yields $\Gamma(t_1) = (0, 3, 10)$, $\Gamma(t_2) = (3, 6, 13)$, $\Gamma(T^{(1)}) = (0, 3, 10)$ and $\Gamma(T^{(2)}) = (0, 3, 10, 6, 13)$. Note that new numbers in the step from $\Gamma(T^{(1)})$ to $\Gamma(T^{(2)})$ are appended at the tail of $\Gamma(T^{(1)})$. The cardinality M' of $\Gamma(T^{(2)})$ is 5.

$\Gamma(T^{(2)})$ includes the indices of the inputs from the sequence $\tilde{s}$ to the filter function f to produce the keystream output at time $t_1 = 0$ and $t_2 = 3$.

For the above configuration we compute the probability $p_1(f, \Gamma, T^{(2)}, 10, (0, 1)^T)$ and the corresponding conditional correlation coefficient $\lambda_1(f, \Gamma, T^{(2)}, 10, (0, 1)^T)$ for condition B_1. For the computation of these values the sequence $\tilde{s}$ is treated as a random sequence with $P(s_t = 0) = P(s_t = 1) = 0.5$ for any $t \geq 0$.

The set S of inputs to the filter function f, which procduce the output $\underline{y}(T^{(2)}) = (0, 1)^T$ at time $t_1 = 0$ and $t_2 = 3$, is

$$\begin{aligned} S = \{&(0,0,0,0,1)^T, (0,1,0,0,1)^T, (0,1,0,1,0)^T, (0,1,0,1,1)^T, \\ &(0,1,1,0,1)^T, (0,1,1,1,0)^T, (0,1,1,1,1)^T, (1,0,0,0,1)^T\}. \end{aligned}$$

The subset S' of S, for which the condition $s_{10} = 0$ holds, is given by

$$S' = \{(0,0,0,0,1)^T, (0,1,0,0,1)^T, (0,1,0,1,0)^T, (0,1,0,1,1)^T, (1,0,0,0,1)^T\}.$$

Thus, we have $p_1(f, \Gamma, T^{(2)}, 10, (0, 1)^T) = 5/8 = 0.625$ and $\lambda_1(f, \Gamma, T^{(2)}, 10, (0, 1)^T) = |0.625 - 0.5| = 0.125$.

B A toy example for the application of the hybrid correlation attack

In this example we show how the first phase of the hybrid correlation attack (see section 4) is applied.

We consider the filter function $f : \mathrm{GF}(2)^9 \to \mathrm{GF}(2)$,

$$f(x_1, \ldots, x_9) = x_1x_2 + x_3x_4 + x_5x_6 + x_7x_8 + x_9$$

with $N_f = 240$ and $p_{e,f} \approx 0.469$. Also let $\Gamma = (0, 1, \ldots, 8)$, the connection polynomial $c \in \mathrm{GF}(2)[x]$ be primitive and of degree $k = 40$. In Fig. 4 the normalised cumulated frequency values $h'_1(p) = 2^{-m} h'_1(f, \Gamma, T^{(m)}, p)$ of the conditional correlation coefficients for condition B_1 are displayed for different values of m, namely for $m = 3, 5$ and 10 and $T^{(m)} = (0, 1, \ldots, m - 1)$.

For 100 different initialisations of the NLFG we have generated keystream sequences of length $N = 10000$. The mean value of the error probability was $p_{e,f,real} = P(e_t = 1) \approx 0.455$ and was every time greater than 0.45. On $\tilde{z}$ we applied phase 1 of the hybrid correlation attack with condition B_1 for $m = 3, 5$ and 10. The average improvements $\Delta p_{e,f,real}$ and the rest error probability $p_{e,1} + p_{e,2} = P(e_{1,t} = 1) + P(e_{2,t} = 1)$ are displayed in Table 1. Even for $m = 3$ the rest error probability achieves $p_{e,1} + p_{e,2} = 0.438416$ so that a fast correlation attack in the third phase of the hybrid correlation attack on the sequence $\tilde{r}$ would be successful.

In this example, the application of phase 2 of the hybrid correlation is also possible and brings further error correction.

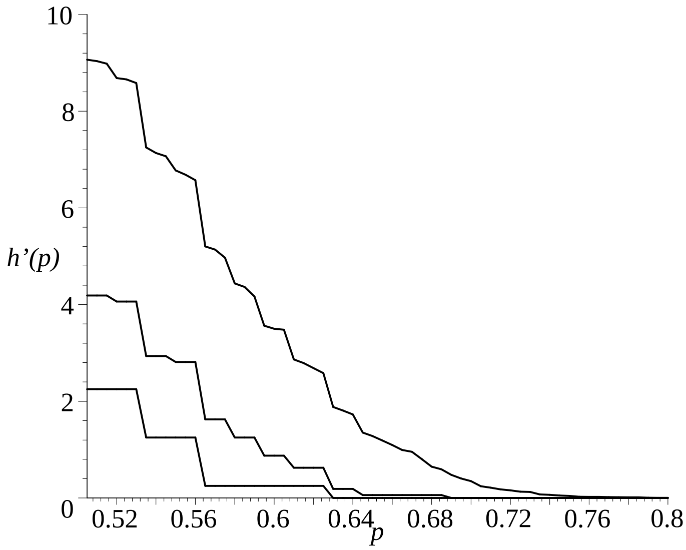

Fig. 4. The normalised cumulated frequency values $h'(p) = 2^{-m} h'_1(f, \Gamma, T^{(m)}, p)$ of the conditional correlation coefficients for $m = 3$ (left), 5 (middle) and 10 (right) for condition B_1.

m	$\Delta p_{e,f,real}$	$p_{e,1} + p_{e,2}$
3	0.017207	0.438416
5	0.036875	0.417820
10	0.058812	0.395700

Table 1. Improvements and rest error probability after applying the first phase of the hybrid correlation attack.

C Introduction to log-likelihood algebra

In this section we give an introduction to the log-likelihood algebra, which was introduced in [HOP96] and is used in the hybrid correlation and the concentration attack in this paper.

Consider a binary random variable $X \in \mathrm{GF}(2)$ with probability $0 \le P(X) \le 1$, i.e. $P(X = 0) + P(X = 1) = 1$.

The log-likelihood $L(X)$ for the random variable X is then defined as

$$L(X) := \ln\left(\frac{P(X=0)}{P(X=1)}\right) = \ln\left(\frac{P(X=0)}{1-P(X=0)}\right).$$

$L(X)$ is strong monotone and takes values from $-\infty$ to $+\infty$ for $0 \le P(X = 0) \le 1$. $L(X) = 0$ if $P(X = 0) = 0.5$. If $L(X) > 0$ then the event $X = 0$ is more likely than $X = 1$. Such $L(X)$ is a measure for the reliability of the event $X = 0$.

If we consider two statistic independent, binary random variables $X_1 \in \mathrm{GF}(2)$ and $X_2 \in \mathrm{GF}(2)$ with probability $P(X_1)$ and $P(X_2)$.

Then we have for the probability of the summation of the two random variables X_1 and X_2

$$\begin{aligned}
P(X_1 + X_2 = 0) &= P(X_1 = 0) \cdot P(X_2 = 0) + P(X_1 = 1) \cdot P(X_2 = 1) \\
&= P(X_1 = 0) \cdot P(X_2 = 0) + (1 - P(X_1 = 0)) \cdot (1 - P(X_2 = 0)) \\
&= 1 + 2P(X_1 = 0) \cdot P(X_2 = 0) - P(X_1 = 0) - P(X_2 = 0).
\end{aligned}$$

With

$$P(X=0) = \frac{e^{L(X)}}{1+e^{L(X)}}$$

we receive the log-likelihood relation of the summation as

$$\begin{aligned}
L(X_1 + X_2 = 0) &= \ln\left(\frac{1 + 2P(X_1=0)P(X_2=0) - P(X_1=0) - P(X_2=0)}{P(X_1=0) + P(X_2=0) - 2P(X_1=0)P(X_2=0)}\right) \\
&= \ln\left(\frac{1 + e^{L(X_1)}e^{L(X_2)}}{e^{L(X_1)} + e^{L(X_2)}}\right) \\
&\approx \mathrm{sgn}(L(X_1)) \cdot \mathrm{sgn}(L(X_2)) \cdot \min(|L(X_1)|, |L(X_2)|).
\end{aligned}$$

This relation can be generalised to the summation of n statistic independent, binary random variables $X_1, \ldots, X_n \in \mathrm{GF}(2)$ with

$$\tanh\left(\frac{x}{2}\right) = \frac{e^x - 1}{e^x + 1}$$

to

$$\begin{aligned}
L\left(\sum_{i=1}^{n} X_i = 0\right) &= \ln\left(\frac{1 + \prod_{i=1}^{n} \tanh\left(\frac{L(X_i)}{2}\right)}{1 - \prod_{i=1}^{n} \tanh\left(\frac{L(X_i)}{2}\right)}\right) \\
&\approx \prod_{i=1}^{n} \mathrm{sgn}(L(X_i)) \cdot \min_{1 \le i \le n}(|L(X_i)|).
\end{aligned}$$

Such the reliability of the summation of n statistic independent, binary variables in GF(2) can be approximated with the minimum value over all involved log-likelihood values.