The Differential Cryptanalysis and Design of Natural Stream Ciphers

Cunsheng Ding

Buchenring 15B
D-76297 Stutensee-Buechig, Germany

Abstract. This paper introduces the differential cryptanalysis of additive stream ciphers, and develops its theoretical basis. The relationships between differential and other types of stream cipher analysis are presented. The conservation laws of patterns and of mutual information are derived. The cryptographic significance of pattern distribution of key-stream sequences is shown. The cryptographic transformation densities are introduced, and their relations with other cryptographic factors are summarized. This work is illustrated by reference to the design and security of additive natural stream ciphers, which are nonlinear filtered sequences driven by a counter rather than by a shift register.

1 Introduction

Stream ciphers have a long history and still play an important role in securing communications. Most of the literature on stream ciphers is about the design and analysis of synchronous stream ciphers, and especially additive synchronous stream ciphers, because of their relatively tractable structure.

The main design problem of additive synchronous stream ciphers is producing a secure key stream generator. So far many kinds of generator have been proposed: nonlinearly-filtered LFSR generators [18], nonlinearly-combined LFSR generators [13, 25], multiplexer generators [17], threshhold generators [10], inner product generators [20], BBS generators [9], knapsack generators [27], Shamir's generators [28], counter generators [11], clock-controlled LFSR generators (survey in [16]), and the shrinking generator [6], to name only a few. Though there are some common security measures for every sequence generator (such as nonlinearity, linear complexity, sphere complexity [12] and 2-adic complexity [19]), every system has its own particular security problems.

Though it may be generally said that cryptographic gains and losses usually go together, there are differences between cipher systems. Some are easy to implement, but may have tradeoffs between known security parameters; some are relatively difficult to implement, but their security may be easy to control; others may have both an easy implementation and ideal security, but be slow. Of course, fewer tradeoffs make for easier design. In designing secure cipher systems the most important problems are:

1. how can we build systems which have as few security tradeoffs as possible?
2. what are the tradeoffs or conflicts in a given system?
3. how do we manage tradeoffs and conflicts?
4. how do we coordinate security and performance?

The additive natural stream cipher presented in this paper will show that it is possible to build some stream ciphers with many security aspects in harmony. Some provably-secure ciphers have been presented in [20] [21], but they seem to be difficult to implement. Well-designed additive stream ciphers give examples of practical ciphers whose security can be proven to a certain extent.

There are many kinds of attack on stream ciphers: divide-and-conquer attacks [29], correlation attacks [14, 22, 29], best affine approximation attacks [12], Zeng-Yang-Rao's consistency attack [31], Anderson's meet-in-the-middle consistency attack [1, 2], and the attacks on clock-controlled generators [15, 23], the derived-sequence attacks [3], to mention only a few. Most of these attacks are key-recovering attacks: they use known keystream to get information about the key. But the techniques involved vary from system to system.

Much work has been done on the differential cryptanalysis for DES-like block ciphers since this technique was introduced by Biham and Shamir [7]. To make an iterated block cipher immune to this analysis, it is necessary to let the round function have good nonlinearity [20]; differential analysis is mainly based on the local linearity analysis of the round function. Differential analysis of hash functions was also carried out by Biham [8], Preneel, Govaerts and Vandewalle [26]. In this paper we extend differential ideas to additive natural stream ciphers: these are like nonlinearly filtered shift register systems, except that the shift register is replaced by a simple counter with arbitrary period N.

2 Additive Natural Stream Ciphers:

The additive natural stream cipher is depicted in Figure 1, and is based on a filtered counter with period N or natural sequence generator (briefly, NSG), where $(\sum)_N$ denotes the integer modulo N addition, "+" denotes the binary operation of an abelian group $(G, +)$ and $f(x)$ a function from Z_N to G. The key is the counter's initial state. Thus, this kind of stream cipher is different from the counter stream ciphers proposed by Diffie and Hellman [11], in that the key consists only of the initial state.

Let a periodic sequence generator have output sequence s^∞ of period N. Now we define a function on the residue ring Z_N by

$$f(i) = s_i, \ i \in Z_N.$$

We see that every sequence generator which produces periodic sequences can be realized by the generator of figure 1 - this is why we call it *natural*, and

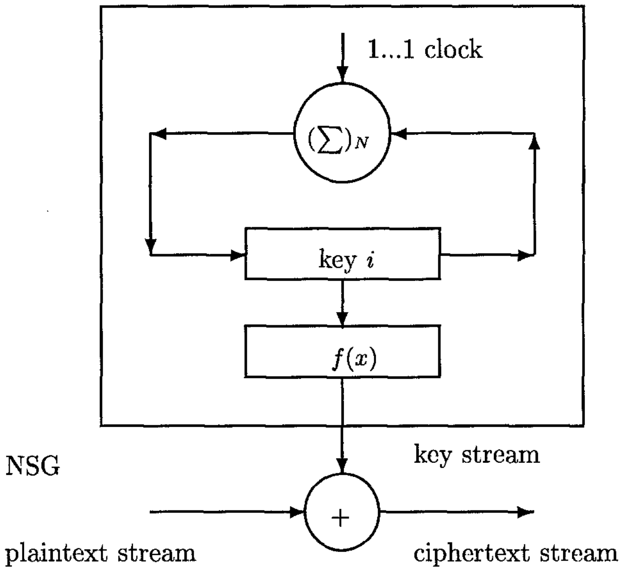

Fig. 1. The additive natural stream cipher system

one reason for its cryptographic significance. Another interesting fact about this generator is that many of its security aspects are easy to coordinate: this will be seen later.

3 The Differential Analysis of Additive Stream Ciphers

For the generator of Figure 1, assume that $f(x)$, N and a piece of key stream $h^t = h_0 h_1 \cdots h_{t-1}$ are known to a cryptanalyst, who wishes to recover the key at the time when h_0 was produced.

3.1 D-Crptanalysis:

Let $C(f)_i = \{x : x \in Z_N,\ f(x) = i\}$ for $i = 0, 1$. Then the procedure of this attack can be described as follows:

Step 1: Find parameters $(i, j; w)'s$ with $(i, j) \in G \times G$, $w \in Z_N$ such that

$$d_f(i, j; w) = |C(f)_i \cap (C(f)_j - w)| \tag{1}$$

is as small as possible, and determine the corresponding sets

$$D_f(i, j; w) = C(f)_i \cap (C(f)_j - w). \tag{2}$$

Step 2: Look at the known sequence h^t and find patterns $i ** \cdots * j$ of length $w+1$ in the known key-stream sequence. If such pattern is found and i is in the k'th position of the known peace of sequence, then

$$k \in D_f(i,j;w) - k'. \tag{3}$$

If $d_f(i,j;w)$ is small, then search the set for k; otherwise, choose another $(i',j';w')$ and find the corresponding $D_f(i',j';w')$ $-k''$. Then

$$k \in (D_f(i,j;w) - k') \cap (D_f(i',j';w') - k''). \tag{4}$$

Continue in this way until the number of the set which contains k is small enough.

3.2 The Form of the Attack:

Let's see why this can be considered to be a differential attack. In Step 1 above, the parameter $d_f(i,j;w)$ is in fact the number of ways in which the element w of Z_N can be written as the difference of the elements of $C(f)_j$ and $C(f)_i$, i.e., it is the number of solutions of the difference equation

$$w = x_j - x_i, \;\; x_j \in C(f)_j, \; x_i \in C(f)_i. \tag{5}$$

That is why (1) and (2) are called difference parameters. The main part of the attack is based on analysing the difference parameters: the idea is to find small w's or large w's which can be written as few difference of some C_i and C_j as possible.

3.3 The Theoretical Basis of the Attack:

Theoretically every bit of a keystream can give information about a generator's initial state and the key. Thus a basic requirement for stream ciphers is that every bit of keystream gives approximately the same amount of information. In our case, this yields balance requirements for the filter function $f(x)$. This *single bit analysis* is apparently suitable to all synchronous stream ciphers. If $n = \log_2 N$, we can write

$$I(k;h_0 = 0) = n - \log_2 |C(f)_0| \text{ bits}, \tag{6}$$
$$I(k;h_0 = 1) = n - \log_2 |C(f)_1| \text{ bits}. \tag{7}$$

Noticing that $|C(f)_0| + |C(f)_1| = N$, we get

$$2^{n-I(k;h_0=0)} + 2^{n-I(k;h_0=1)} = N. \tag{8}$$

This is the theoretical basis for keeping the mutual information stability of a keystream as flat as possible.

If we now consider two bits h_i and h_j separately or arbitrarily, we may not obtain $I(k; h_i) + I(k; h_j)$ bits of information about the key. If the cipher is not properly designed, some combinations of bits may give much more information about the key than others. We call such combinations with their length $(h_i, h_j, |i-j|)$'s *bad patterns.* The idea behind the differential attack is to look for bad patterns, and in particular for triples $(i, j; w)$ which gives as much information about the key as possible.

One may argue that we should design our cipher so that the mutual information $I(k; (i, j; w))$ is as small as possible for all $(i, j; w) \in Z_2 \times Z_2 \times Z_N$, but in fact we cannot achieve this: one pattern $(i, j; w) \in Z_2 \times Z_2 \times Z_N$ gives us

$$I = n - \log_2 d_f(i,j;w) = n - \log_2 |C(f)_i \cap (C(f)_j - w)| \text{ bits} \tag{9}$$

of information about the key. Now consider the following theorems:

Theorem 1. *(Conservation Law for Difference Parameters): With symbols as above, we have*

$$\sum_j d_f(i,j;w) = |C(f)_i|, \; i \in Z_2, \; w \in Z_N; \tag{10}$$

$$\sum_i d_f(i,j;w) = |C(f)_j|; \; j \in Z_2, \; w \in Z_N; \tag{11}$$

$$\sum_{(i,j)\in Z_2 \times Z_2} d_f(i,j;w) = N, \; w \in Z_N. \tag{12}$$

These are the laws of conservation between difference parameters which appear in three forms. It follows that:

Theorem 2. *(Conservation Law of Mutual Information:) With symbols as above, we have*

$$\sum_j 2^{n-I(k;(i,j;w))} = |C(f)_i|, \; i \in Z_2, \; w \in Z_N; \tag{13}$$

$$\sum_i 2^{n-I(k;(i,j;w))} = |C(f)_j|; \; j \in Z_2, \; w \in Z_N; \tag{14}$$

$$\sum_{(i,j)\in Z_2 \times Z_2} 2^{n-I(k;(i,j;w))} = N, \; w \in Z_N; \tag{15}$$

$$\sum_{(i,j;w)} 2^{n-I(k;(i,j;w))} = N^2. \tag{16}$$

It is not difficult to prove the above theorems, which provide the theoretical basis for analysing mutual information stability between two-bit patterns and the key. Generalization of the above theorems to the case of an arbitrary finite G is also easy.

In particular, similar results hold for three-bit patterns, and attacks based on three-bit pattern analysis can also be developed.

Now one question arises: does mutual information stability between the key and both one-bit and two-bit patterns give us any results for patterns of more bits? The answer is yes - but the proof is not easy.

3.4 The Attack and Pattern Distribution:

Many tests and measures of sequence randomness have been proposed in the literature, including Golomb's three postulates. One may ask why these postulates are important; this is shown clearly by differential attacks. Indeed, a stronger randomness postulate is indicated:

> All patterns $(i_1, i_2, \cdots, i_t)$ with fixed distances between i_u and i_{u+1} for $u = 1, ..., t-1$, should appear with probability about 2^{-t} in a key stream sequence.

If some keystream sequence has bad patterns with significantly *lower* than random probability, then searching for these patterns may enable us to break it more quickly than exhaustive search.

3.5 Practical Attacks:

As shown, the key part of this attack is the first step, i.e., finding some of the parameters of (1) and (2). In practice we do not need to determine all of these, as a selection of them may be enough to determine the key. To this end, we may first try to determine the $C(f)_i$ for some i, and then use (1) and (2) to determine the $d_f(i, j; w)$ and $D_f(i, j; w)$. Another method is to introduce the function

$$g_{i,j;w}(x) = (f(x) + 1 + i)(f(x + w) + 1 + j),$$

where $f(x)$ is a binary function. It is easy to know that $C(g_{i,j;w})_1 = D_f(i, j; w)$. This means that the determination of the differential parameters of (1) and (2) is equivalent to that of characteristic set of $g_{i,j;w}(x)$. Since the algorithm is known, there are two ways to do this: computation and theoretical analysis.

The complexity of the computational method will depends partially on N and $f(x)$. If N is not large enough, it will be possible to break the system. For example, if $N = 19$, $f(x) = (x^2 \bmod 19) \bmod 2$, we have

$$\begin{aligned}
&C(f)_0 = \{0, 2, 17, 4, 15, 5, 14\} \\
&D_f(0,0;1) = \{4, 14\},\ D_f(0,0;2) = \{17, 0, 15, 2\}, \\
&D_f(0,0;3) = \{14, 2\},\ D_f(0,0;4) = \{0, 17, 15\}, \\
&D_f(0,0;5) = \{0, 14\},\ D_f(0,0;7) = \{14, 17\}.
\end{aligned}$$

Thus, one pattern 00 in the output sequences makes the set containing the key to be $\{4, 14\}$. Similarly, $0 * *0$ gives $\{14,\ 2\}$, $0 * * * *0$ gives $\{0, 14\}$, etc.. Furthermore, two patterns 00 and $0 * *0$ determine the key 14. Thus, if we can compute difference parameters exhaustively, then we can break such a cipher unless $f(x)$ is extremely well chosen. However, the computational complexity of this task needs to be further investigated.

The other approach, theoretical analysis, means determining the differential parameters algebraically. A poor choice of $f(x)$ may make it easy to find $C(f)_i$'s and thus the differential parameters. Let us illustrate this by the following example.

We choose for the additive NSC of Figure 1 $N = 4t + 1$ a prime with t also prime or $N = 4t - 1$ with $2t - 1$ also prime, and filter function $f(x) = x \bmod 2$. It is not difficult to show that there is no trivial affine function from $(Z_N, +)$ to $(Z_2, +)$ when N is odd. It follows that $f(x)$ is nonlinear with respect to the additions of Z_N and Z_2. It can be proven that $L(s^\infty) = N - 1$ and $SC_k(s^\infty) \geq N - 1$ if $0 \leq k \leq (N - 3)/2$, where $L(s)$ denotes the linear complexity of s, and $SC_k(s)$ is the sphere complexity of s [12]. By analysis we have $C_0 = \{0, 2, 4, \cdots, N-3, N-1\}$, $C_1 = \{1, 3, 5, \cdots, N-4, N-2\}$ and the differential results are as follows:

$$\begin{aligned}
&C_0 \cap (C_0 - 1) = \{N - 1\} \\
&C_0 \cap (C_0 - 3) = \{N - 1, N - 3\} \\
&C_0 \cap (C_0 - 5) = \{N - 1, N - 3, N - 5\}
\end{aligned}$$

$$C_0 \cap (C_0 - (N - 2)) = \{N - 1, N - 3, N - 5, \cdots, 2\}.$$

This means that if we have two bits 00 in the known keystream, then the key that generate the first 0 was $k = N - 1$. We have

$$\begin{aligned}
&00 \to k = N - 1 \\
&0 * *0 \to k = N - 1, \text{ or } N - 3 \\
&0 * * * *0 \to k = N - 1, N - 3, \text{or } N - 5.
\end{aligned}$$

Similar arguments show that

$$C_0 \cap (C_1 - 2) = \{N - 1\}$$

$$C_0 \cap (C_1 - 4) = \{N-1, N-3\}$$
$$C_0 \cap (C_1 - 6) = \{N-1, N-3, N-5\}$$

$$C_0 \cap (C_1 - (N-1)) = \{N-1, N-3, N-5, \cdots, 2\}.$$

$$0 * 1 \to k = N-1$$
$$0 * * * 1 \to k = N-1, \text{ or } N-3$$
$$0 * * * * * 1 \to k = N-1, N-3, \text{or } N-5.$$

$$C_1 \cap (C_0 - 2) = \{N-2\}$$
$$C_1 \cap (C_0 - 4) = \{N-2, N-4\}$$
$$C_1 \cap (C_0 - 6) = \{N-2, N-4, N-6\}$$
$$\vdots$$
$$C_1 \cap (C_0 - (N-1)) = \{N-2, N-4, N-6, \cdots, 1\}.$$

$$1 * 0 \to k = N-2$$
$$1 * * * 0 \to k = N-2, \text{ or } N-4$$
$$1 * * * * * 0 \to k = N-2, N-4, \text{or } N-6.$$

$$C_1 \cap (C_1 - 2) = \{N-1\}$$
$$C_1 \cap (C_1 - 4) = \{N-1, N-3\}$$
$$C_1 \cap (C_1 - 6) = \{N-1, N-3, N-5\}$$
$$\vdots$$
$$C_1 \cap (C_1 - (N-1)) = \{N-1, N-3, N-5, \cdots, 2\}.$$

$$1 * 1 \to k = N-1$$
$$1 * * * 1 \to k = N-1, \text{ or } N-3$$
$$1 * * * * * 1 \to k = N-1, N-3, \text{or } N-5.$$

The above analysis shows that the bad patterns in the keystream are:

$0 * \cdots * 0$ of length $2l$ with l small enough

$0 * \cdots * 1$, $1 * \cdots * 0$ and $1 * \cdots * 1$ of length $2l+1$ with l small enough.

Thus keys which are very near to $N-1$ (k or $N-k$ for small k) are very weak under this analysis. Those keys which are far away 0 in both directions are of course strong keys.

Of course, the output sequence of the NSG of Figure 1 is cryptographically very bad. This is because the nonlinearity (and the difference property) of $f(x)$ is poor. The purpose of the example is to show that if the characteristic sets $C(f)_i$ can be determined algebraically, then the cipher can often be broken, provided that the $f(x)$ has bad difference property. Thus there are two rules a stream cipher designer must follow:

1. make sure that the characteristic sets of your $f(x)$ can not be determined (whether computationally or algebraically); or
2. choose f so that its characteristic sets have good difference properties.

If a designer chooses the second option, then the cipher may be secure against this attack even if the characteristic sets are given to an attacker.

4 Differential and Other Analyses

The analysis of (1) is called the *differential analysis* of the additive natural stream ciphers. It is equivalent to the following:

1. nonlinearity analysis of $f(x)$ with respect to Z_N and G;
2. autocorrelation analysis of the key stream sequences;
3. analysis of the mutual information between the key and the key stream sequence;
4. analysis of the transformation density of the ciphers.

Its equivalence to the analysis of the mutual information has already been implied in the above discussion. The proofs of other equivalences are too long to be presented here. In what follows, we would like to introduce the notion of cryptographic transformation density and its relation to the other analytic techniques.

Let M be the plaintext space, C the ciphertext space, K the key space and T_K the set of encryption or decryption transformations specified by the keys. Then the transformation densities are defined by

$$D(T,K) = 1 - \sum_{k,k'} \frac{p(t_k,t_{k'})}{\binom{|K|}{2}}$$

$$D_0(T,K) = 1 - \max_{k,k'} p(t_k,t_{k'})/|K|,$$

where $p(t_k,t_{k'})$ denotes the probability of agreement between the two encryption or decryption transformations specified by the two keys, which is usually replaced by $d(t_k,d_{k'})/|M|$ for simplicity. The transformation densities were inspired by the following three questions about cryptography.

> *Question 1:* To break a cipher or to decipher a piece of ciphertext, do we have to recover the original key?

Question 2: Are the encryption decryption transformations specified by the keys really "different" from one another?

Question 3: When the answer to Question 2 is "yes", for a given key k, is there any key $k' \in K$ such that the probability of agreement $p(t_k, t_{k'})$ or the distance $d(t_k, t_{k'})$ is small enough? If there are, which are they and how many?

The importance of the questions are clear, as attacks may involve trying partial keys. That they are practical, follows from the fact that the M-209 cipher machine had large equivalence classes of keys. However, it seems that for most proposed ciphers the above three question have not been answered.

The transformation density (briefly, T-density) is related to partial-key attacks, key density, key size, message density, message and cryptogam residue classes, perfect secrecy, autocorrelation and crosscorrelation functions of sequences, difference sets, difference property of partitions, nonlinearity of cryptographic functions, affine approximation of functions, mutual information stability and source coding.

5 Differential Analysis and Security

For the additively natural stream ciphers of Figure 1, the following attacks are related:

1. differential attacks;
2. key determining attacks based on decision trees;
3. partial-key attacks;
4. linear approximation attacks with respect to additions of Z_N and G;
5. key (or key stream) correlation attacks.

The idea of the key determining attacks based on decision trees is to make use of the known keystream segment sequentially. For example, let $h_0 h_1 \cdots h_{t-1}$ be the known keystream segment. Using h_0 and the cryptographic algorithm we can decide whether $k \in C(f)_0$ or $\in C(f)_1$. If $k \in C(f)_0$, then we continue withh_1 and partition $C(f)_0$ into two further sets, and so on. Continuing in this way we can zero in on the key

$$U_0 \supseteq U_1 \supseteq \cdots \supseteq U_l = \{k\}$$

This kind of attack makes use of the known sequence sequentially, but not optimally. That is one of the ways in which it differs from differential attack.

The idea of the partial-key attack is to partition the key space into small subspaces such that the distance between two decryption transformations specified by any two keys in a subspace is smaller than a given constant c. If for a properly chosen c there is a usable partition such that the number of subsets is not large, then we can choose only one key in each subset to decipher a peace of ciphertext until a correct message is obtained. The probability of correct decipherment is determined by the constant. These are therefore ciphertext-only attacks.

Linear approximation attacks refer to all the attacks based on the linear approximation of the crytographic function $f(x)$ with respect to the additions of Z_N and G. Key correlation or key-stream correlation attacks refer to all the attacks based on the cross-correlation analysis of the output sequences produced by two keys.

It can be proven that for the cipher system of Figure 1, if N is chosen large enough and the filter function $f(x)$ is chosen such that the difference parameter of (1) is stable (it is approximately the same for all (i, j)), then the cipher is secure against all the attacks mentioned above. Furthermore, if the partner pair (N, $(G, +)$) is properly chosen, the linear complexity and linear-complexity stability of the key-stream sequence can be controlled without causing tradeoffs between the linear complexity aspect and the nonlinearity (difference property) aspect. Thus, many security aspects of the additive natural system are in harmony.

6 The Design of Natural Sequence Generators

The discussions in the previous sections show that the main problems in designing the NSG of Figure 1 consist of

- choosing the partner pair $(Z_N;\ (G, +))$; and
- designing the cryptographic function $f(x)$

such that

1. the partner pair $(Z_N;\ (G, +))$ works in "harmony";
2. the output sequences have large linear complexity, good linear complexity stability, good autocorrelation, and good pattern distribution;
3. the cryptographic function $f(x)$ has good nonlinearity with respect to $(Z_N, +)$ and $(G, +)$ and also good difference property with respect to $(Z_N, +)$.
4. the additive natural stream ciphers have good cryptographic transformation densities and are secure against all the five attacks mentioned in Section 5.

For binary NSGs, i.e., $(G, +) = (Z_2, +)$, we should choose first the cryptographic function $f(x)$. Possible choices are the characteristic functions of the cyclic difference sets of $(Z_N, +)$, i.e., the characteristic sets $C(f)_0$ and $C(f)_1$ should be the (N, k, λ) residue difference sets such that $k/N \approx 0.5$. Good choices for the cryptographic function $f(x)$ are also those such that the partition $\{C(f)_0, C(f)_1\}$ of Z_N has good difference property, i.e., the parameter in (1) is fairly stable.

One of the best candidates for $f(x)$ is the function

$$f(x) = (x^{(N-1)/2} \bmod N) \bmod 2, \tag{17}$$

where N is chosen to be a large prime. If the prime N is of the form $4t-1$, then $C(f)_1$ is the famous $(N,(N-1)/2,(N-3)/4)$ quadratic residue difference set. If N is of the form $4t+1$, then $C(f)_0$ and $C(f)_1$ are not residue difference sets, but they are almost-difference sets, by which we mean that the difference property of this partition of Z_N is almost the same as the case of N being of the form $4t-1$. We call the corresponding NSGs *DSC (difference-set characterized) generators* and *ADSC (almost difference-set characterized) generators.* In fact the nonlinearity of the function of (17) with respect to $(Z_N,+)$ and $(Z_2,+)$, which is optimal, is determined by the cyclotomic numbers of order 2 [30]. The proof of this is to long to be presented here.

To control the linear complexity and linear-complexity stability of the DSC generator, N should be chosen to be $N=4t-1$ with N and $2t-1$ both prime. This will guarantee

$$L(s^\infty) \geq N-1, \tag{18}$$
$$SC_k(s^\infty) \geq N-1, \text{ if } 0 \leq k < \min\{WH(s^N), N-WH(s^N)\}, \tag{19}$$

where $WH(s^N)$ denotes the Hamming weight of one period segment of the periodic sequence s^∞. To control these for the ADSC generator, $N=4t+1$ with both N and t being prime is chosen. This gives the same results of (18) and (19). In fact, in both cases, we actually get Legendre sequences. It should be pointed out that there are both good and bad cryptographic Legengdre sequences. Provided we choose the prime N properly, the corresponding Legendre sequences are among the best cryptographic sequences.

Another good binary NSG is the twin-primes generator, in which we choose the cryptographic function $f(x)$ to be the characteristic function of the famous twin-primes difference set [4]. In fact this function can be expressed by

$$f(x) = \begin{cases} 1, \; x = 0, p, p+2, 2(p+2), \cdots, (p-1)(p+2); \\ 0, \; x = p, 2p, \cdots, (p+1)p; \\ (1+(\frac{x}{p})(\frac{x}{p+2}))/2, \; \gcd(j, p(p+2)) = 1, \end{cases} \tag{20}$$

where (a/p) denotes the Legendre symbol. Here $N=p(p+2)$, p and $p+2$ both are primes. If p is chosen of the form $4t+1$, then the linear complexity of the sequence is $N=p(p+2)$. To control the linear complexity stability we should choose $p=4t+1$ where both t and $2t-1$ consist only of large prime factors. All of the above generators are relatively easy to implement using fast exponentiation algorithms.

Let s^∞ be a binary sequence of period N. setting

$$I_1 = \{i : \; s_i = 1, \; 0 \leq i \leq N-1\},$$
$$I_0 = \{i : \; s_i = 0, \; 0 \leq i \leq N-1\},$$

we know that I_0 and I_1 are a partition of Z_N. We call I_0 and I_1 the *characteristic sets* of s^∞, and conversely s^∞ the *characteristic sequence* of the partition $\{I_0, I_1\}$. Let

$$S^N(x) = s_0 + s_1 + \cdots + s_{N-1}x^{N-1}.$$

Then we have the following theorem about the linear complexity of the characteristic sequence of a difference partition:

Theorem 3. *Let I_0, I_1 be a partition of Z_N with I_1 being a (N, k, λ) difference set of Z_N and s^∞ be the characteristic sequence of the partition. Then*

1. *If k is even and λ odd, then $L(s^\infty) = N - 1$.*
2. *If k is odd and λ even, then $L(s^\infty) = N$.*
3. *If k and λ both are even, then*

$$L(s^\infty) = deg[\frac{\gcd(S^N(x^{-1})x^N, x^N + 1)}{\gcd(\gcd(S^N(x), x^N + 1), \gcd(S^N(x^{-1})x^N, x^N + 1))}].$$

4. *If k and λ both are odd, then*

$$L(s^\infty) = deg[\frac{\gcd(S^N(x^{-1})x^N, x^N + 1)(x + 1)}{\gcd(\gcd(S^N(x), x^N + 1), \gcd(S^N(x^{-1})x^N, x^N + 1))}].$$

This theorem shows how to control the linear complexity of output sequences of a natural sequence generator when we want to use the characteristic functions of difference sets as the cryptographic functions for this kind of generators. This can be done by choosing these difference sets with $n = k - \lambda$ being odd.

Let I_0 and I_1 be a partition of Z_N with I_1 being an (N, k, λ) difference set of Z_N. The function defined by

$$f(x) = i, \text{ for all } i \in I_i, \ i = 0, 1,$$

is called the characteristic function of the partition. Concerning the nonlinearity of the characteristic function of an (N, k, λ) difference set we have the following conclusion:

Theorem 4. *Let I_1 be an (N, K, λ) difference set, $f(x)$ the characteristic function of the partition $\{I_0, I_1\}$. Then for any $\alpha \neq o$, it holds*

$$p(f(x) - f(y) = \beta | x - y = \alpha) = \begin{cases} [N - (k - \lambda)]/N, & \beta = 0, \\ 2(k - \lambda)/N, & \beta = 1. \end{cases}$$

This result means that only the characteristic functions of those residue difference sets of parameters (N, k, λ) with $k \approx N/2$ and $k - \lambda = N/4$ have good nonlinearity with respect to $(Z_N, +)$ and $(Z_2, +)$. The nonlinearity of the cryptographic functions of the DSC generator, the ADSC generator and the twin-primes generator can easily be written down using the above theorem.

Suppose we choose a large prime $p = de + 1$, $(G, +) = (Z_d, +)$ and filter function $f(x) = h(g(x))$ with

$$g(x) = \begin{cases} c, & x = 0; \\ x^{(p-1)/d} \bmod p, & x \neq 0, \end{cases} \tag{21}$$

where $c = a^{(p-1)/d} \bmod p \neq 0$ is a constant, while the function $g(x)$ is defined by $g(x) = \log_\theta x$, where θ is a primitive root modulo p. It can be shown that the nonlinearity of the $f(x)$ with respect to $(Z_p, +)$ and $(Z_d, +)$ and the difference property of the cryptographic function with respect to $(Z_p, +)$ is determined by the cyclotomic numbers of order d, which is usually ideal. The linear complexity and the linear-complexity stability of the output sequences need to be controlled, but this can be done in many cases.

Interestingly, the properties of this kind of generator are related to many (if not most) of the outstanding problems in number theory; so it is not surprising that we can only make part of the design mathematically tractable.

Acknowlegements: The author would like to thank Ross Anderson for making many good suggestions for the paper.

References

[1] R. J. Anderson. *Solving a class of stream ciphers*, Cryptologia 14, no. 3, 1990, pp. 285-288.

[2] R. J. Anderson. *Fast Attack on Certain Stream Ciphers,* Electronics Letters, 22nd July 1993, Vol. 29, No. 15, pp. 1322-1323.

[3] R. J. Anderson. *Derived sequence attacks on stream ciphers,* presented at the Rump Session of Crypto'93.

[4] L. D. Baumert. *Cyclic Difference Sets,* Lecture Notes on Mathematics, Vol. 182, Springer-Verlag, 1971.

[5] H. Beker, F. Piper. *Cipher Systems: The Protection of Communications,* Northwood Books, London, 1982.

[6] D. Coppersmith, H. Krawczyk, Y. Mansour. *The Shrinking Generator,* Preproceedings of Crypto'93.

[7] E. Biham, A. Shamir. *Differential cryptanalysis of DES-like cryptosystems,* Advances in Cryptology, Proc. of Crypto '90, LNCS, Springer-Verlag, 1990.

[8] E. Biham. *On the applicability of differential cryptanalysis to hash functions,* E.I.S.S. Workshop on Cryptographic Hash Functions, Oberwolfach (D), March 25-27, 1992.

[9] L. Blum, M. Blum, M. Shub. *A simple unpredictable pseudorandom number operator,* SIAM J. Comput. 15, pp. 364-383.

[10] J. O. Brüer. *On pseudorandom sequences as crypto generators,* Proc. of Int. Zürich Sem. on Digital Communic., Zürich, Swithland, 1984.

[11] W. Diffie, M. Hellman. *Privacy and authentication: An introduction to cryptography,* Proc. IEEE, vol. 67(3), Mar. 1979, pp. 397-427.

[12] C. Ding, G. Xiao, W. Shan. *The Stability Theory of Stream Ciphers.* LNCS, vol. 561, Springer-Verlag, 1991.

[13] P. R. Geffe. *How to protect data with ciphers that are really hard to break,* Electronics, Jan. 4, 1973.

[14] J. Golič, M. Mihaljevič. *A generalized correlation attack on a class of stream ciphers based on the Levenshtein distance,* J. Cryptology, Vol. 3(3), pp. 201-212, 1991.

[15] D. Gollmann, W. G. Chambers. *A Cryptanalysis of* $Step_{k,m}$*-Cascadeds,* Proc. Eurocrypt'89, J. Quisquater, J. Vandewalle (eds.), Springer-Verlag, LNCS 434, pp. 680-687, 1990.

[16] D. Gollmann, W. G. Chambers. *clock-controlled shift registers: A review,* IEEE J. on Selected Areas in Communications, vol. 7, no. 4, May 1989, pp. 525-533.

[17] S. M. Jennings. *Multiplexed sequences: Some properties of the minimal polynomial,* LNCS, vol. 149, Springer-Verlag, 1983, pp. 189-206.

[18] E. L. Key. *An analysis of the structure and complexity of nonlinear binary sequences generators,* IEEE Trans. Inform. Theory, vol. IT-22, no. 6, Nov. 1976, pp. 732-763.

[19] A. Klapper, M. Goresky. *2-adic shift registers,* Proc. of the 1993 Cambridge Algorithm Workshop, December 9-11 1993 (this volume)

[20] J. L. Massey, I. Ingemarsson. *The Rip van Winkel Cipher: A simple and provably computationally secure cipher with a finite key,* in IEEE Int. Symp. on Inform. Theory, (Brighton, England), Abstr. June 24-28, 1985, pp. 146.

[21] U. Maurer. *A provably-secure strongly randomized cipher,* in Advances in Cryptology, Eurocrypt'90, I. Damgård, Ed., LNCS, vol. 473, Springer-Verlag, 1991, pp. 361-373.

[22] W. Meier, O. Staffelbach. *Fast correlation attacks on certain stream ciphers,* J. Cryptology, Vol. 1(3), pp. 159-176, 1989.

[23] R. Mennicocci. *Cryptanalysis of a two-stage Gollmann cascade generator,* Proc. of SPRC'93, W. Wolfowicz (ed.), pp. 62-69, 1993.

[24] K. Nyberg, L. R. Knudsen. *Provable security against differential cryptanalysis,* Advances in Cryptology: Eurocrypt'92.

[25] V. S. Pless. *Encryption schemes for computer confidentiallity,* IEEE Trans. Comput., vol. C-26, Nov. 1977, pp. 756-763.

[26] B. Preneel, R. Govaerts, J. Vandewalle. *Differential Cryptanalysis of Hash Functions Based on Block Ciphers,* Proc. of the 1st ACM Conference on Computer & Communications Security, Fairfax VA, Nov 1993, published by the ACM pp. 183-188.

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

[28] A. Shamir. *On the generation of cryptographically strong pseudo-random sequences,* 8th Int. Colloquium on Automata, Languages and Programming, LNCS vol. 62, Springer-Verlag, 1981.

[29] T. Siegenthaler. *Decrypting a Class of Stream Ciphers Using Ciphertext only,* IEEE Trans. Computers, Vol. C-34, No. 1, Jan. 1985, pp. 81-85.

[30] T. Storer. *Cyclotomy and Difference Sets,* Markham Publishing Company, Chicago, 1967.

[31] K. C. Zeng, C. H. Yang, T. R. N. Rao. *On the linear consistancy test (LCT) in cryptanalysis and its applications,* Advances in Cryptology, Crypto'89, Springer-Verlag, LNCS 435, pp. 164-174.