Non-Linear Feedback Shift Registers over Z_n*

Raúl Gonzalo, Daniela Ferrero, Miguel Soriano
Departament de Matemàtica Aplicada i Telemàtica
Universitat Politècnica de Catalunya
Campus Nord, Edifici C3. c/ Gran Capità, s/n
08034 Barcelona, Spain
e-mail: {telrgd, matdfc, soriano}@mat.upc.es

Abstract

Stream ciphering devices seem to be one of the best alternatives to provide confidentiality to high-speed transmissions.

Several indexes have been proposed for guaranteeing the security of stream ciphers, such us the linear complexity of the key stream, randomness and correlation immune attacks, but they are not sufficient to guarantee the security. There are sequences with a high linear complexity, that after changing a few bits, its complexity changes fast. In this case, a Best Affine Approximation attack (BAA) could be successful; the sequences are well approximated by others with very lower linear unpredictability. This problem is especially important when linear feedback shift registers are used. To solve this, non-linear next state functions are a possible solution. This paper shows some techniques to analyse and characterize non-linear functions with maximum period.

1 Introduction

The recent growth of multimedia services and supercomputing applications has led to a need for high speed networks which can carry all fundamental

*Published in the Proc. *Research Conference on Algebra and Discrete Mathematics of the European Mathematical Society*

1 Introduction

The recent growth of multimedia services and supercomputing applications has led to a need for high speed networks which can carry all fundamental media streams: data, voice and video. The Asynchronous Transfer Mode (ATM) is the most promising technique for high speed networks in the near future, for the public WANs, the Broadband Integrated Services Digital Network (B-ISDN) and also private local networks like ATM LANs.

Some applications and services for both scenarios require security services. First, it is necessary to control the access of users to the resources by means of smart cards, passwords, fingerprints, etc.. Mutual authentication is also required for communications over the network, between clients and servers. We have to follow an authentication protocol that uses some encryption mechanisms in order to achieve this authentication. Some applications can also negotiate session keys during or after the authentication that can later be used to cipher the communication, providing integrity and/or confidentiality.

The services of integrity and confidentiality must be applied for the bulk of the information. So, we need fast ciphering mechanisms for high-speed networks. With today's technology, hardware implementations of stream ciphers seem to be the best choice to encrypt at a rate of hundreds of Mbits/sec, and thus be compatible with high-speed networks.

In the next section, the use of structures with a non-linear feedback is justified. In the Section 3, we propose an algorithm to characterise the non-linear functions by means of which the shift registers present the maximum period. And finally, we present other results obtained when the problem is seen from the graph theoretical point of view.

2 Linear feedback vs. non-linear feedback

The typical structure of a stream cipher is the shown in figure:

The functions fs and fo are known as the next state function and output function respectively. The goal is to fs and fo which guarantees that the sequence fulfils the necessary security requirements. Most of the stream ciphers use a linear feedback in order to reach maximal length sequences (by means of a primitive polynomial), and a non-linear output mapping.

In [11], introduced the concept of correlation immunity. A Boolean combining function is m-th order correlation immune , if and only if its Walsh transform satisfies , being the Hamming weight of the binary n-tuple . As , if is m-th order correlation immune, and m is large enough, then the function is very similar to its best affine approximation (BAA). It is necessary remember that the probability that a function agrees with its BAA is.

In case of a function is m-th order correlation immune, the probability of agreement between and (its BAA) is:

As can easily be seen, this probability is very close to 1 when m is large enough. In consequence, when a linear next state function is used (linear feedback), due to the predictability of the states evolution, there is a commitment in the output sequence. If a high order of correlation immunity is desired, the output sequence can be very well approximated by a sequence linearly generated.

Nevertheless, the situation differs when a non-linear feedback function is used. In this case, the state evolution of the generator cannot be predicted. If the non-linear feedback function is approximated by its BAA, every time that a difference is done, a propagation error process is done because the actual state of the generator is different to the foreseen.

3 Non-linear feedback function analysis

In this section, we will analyse the features of the feedback functions associated to the FSR (Feedback Shift Registers) that originate maximum period sequences, called DeBruijn sequences. They have excellent statistical properties, which permits their use in a large variety of applications, from random bit generation to criptography.

The DeBruijn sequences originally arise from the teleprinter's problem.

Definition 3.1 *A DeBruijn sequence of order L is a cyclic sequence such that each subsequence of L symbols appears exactly once.*

Since there are $2L$ distinct L-tuples formed from 0 and 1, the sequence is $n = 2L$ bits long. It is not difficult to see that the alphabet size needs not be 2. It could be any number m. In that case, the maximum-length sequence

is mL symbols long. We will concentrate the study on the binary case for simplicity and practical reasons.

It is well known that the number of all the different DeBruijn sequences of order L is [1]. These sequences are usually generated by FSR. Each DeBruijn sequence has associated a different feedback function. Let L denote the length of the FSR (it will also be the order of the DeBruijn sequence generated). The sequence bits will be calculated from lower to higher index, i.e. $s_k = f(s_{k-1} \dots s_{k-L})$, where f() is the feedback function. The vector of the register will be denoted as this: $x_{L-1} \dots x_1 x_0$. The most significant bit (MSB) will be the $(L-1)$-th bit and the LSB will correspond to the 0 tap of the FSR. When the register is in the $x_{L-1} \dots x_1 x_0$ state, the next state will be the vector $x_{L-1} \dots x_1 x_0$, where $x_i = x_{i+1}$ for every i from 0 to $L-2$, and x_{L-1} is calculated with the expression given by f(). Graphically,

By representing the feedback functions in algebraic normal form some interesting properties can be deduced assuming the period to be maximum:

a) The 1 term always appears

b) The number of terms is even

c)

d) There is symmetry between the xi and the xL-i variables

e) At least there is one

Proofs

a) The 100...0 state must follow the 00...0 state. In other words,

b) Similar to (a), the 011...1 state must follow the 11...1 state, i.e.

By the same reason, since there are 2L coefficients, there are also an even number of ci = 0.

c) Each state must only have one predecessor. Golomb [GOLO 67] showed that this property (of having no prelude) is equivalent to separate the function in two parts, one linear part having only x0 and another part depending on the other variables:

These functions are known as non-singular functions.

d) An example is the better choice to illustrate this concept. Say, for example, we have a 5th order function:

and we change the each variable by this manner: x1 x4 and x2 x3, we obtain this function

If anyone of them is a DeBruijn function, then the other one will also be a DeBruijn function. (in this case, both of them are). Strictly speaking, given a maximum-period function, if we swap the indexes of the coefficients: xi xL-i we obtain another function having the same property.

There are many different ways for showing this property. The most intuitive proof is arguing over the DeBruijn graph associated to the FSR (we will later define this graph), but it can be also demonstrated algebraically. We know that the function must have the (c) form:

Imagine an infinite sequence generated by this recursion formula. Each bit in the sequence is calculated from the L previous bits:

and if the function f() is properly defined, we can obtain a DeBruijn sequence. Suppose it is a DeBruijn sequence, if we invert the time axis we obtain an another different DeBruijn sequence (for example, given 11100010, the inversion would derive to 01000111). Now, the time goes the other way round, so the next bit would be calculated as:

where the g(.) function is the same as the above equation, but here we have the time inverted, so the bits are inverted as well.

e) We call linear terms to those terms in the algebraic normal form that only contain one variable: x0, x1, ..., xL-1. The reason for not being all these terms together is for avoiding the following cycles:

We will reason by reducing to the absurd. Suppose that the function is like this:

Expressing the g() function in CNF (conjunctive normal form, commonly known as minterms) each xi terms may come only from the or states. It is easy to see that the term must appear in g(). Each negation of a variable can be transformed into the ANF form replacing by . If we want all the linear terms to be in the function, we must impose to exclude the terms.

the cycle 00...00 10...00 01...00 ... 00...10 00...01 00...00 is possible the function cannot give a maximum period sequence So, we arrive to the conclusion that there cannot be all the linear terms in the function. Every feedback function that leads to a maximum period sequence has to fit all the above necessary, but not sufficient, conditions. We have found many other regularities as, for example, that the terms always appears (for orders larger than 2); or that if one inverts the terms of the function (i.e. one term is in the new function if and only if it doesn't appear in the first function) half of

the functions are full period generators.

4 Graph insights and results

Another very useful point of view is using the graph theory. A directed graph (digraph) G is defined to be an ordered pair G = (V,E) of a set of vertex and a set or arcs. Each arc is a pair or vertexes that indicates the beginning and the ending of the arc. The DeBruijn graphs (also known as Good's diagrams) have been deeply studied.

DeBruijn graph for L = 2 DeBruijn graph for L = 3

Each node of the graph is a state of the FSR. The objective is to obtain a walk through all the vertexes without repeating them, which is called a Hamiltonian circuit. As the DeBruijn graph has the property that if it is double the resulting graph is also a DeBruijn graph of higher order, it can be shown that the problem of finding a Hamiltonian circuit in an n-order DeBruijn graph is equivalent of finding an Eulerian path (to pass all the arcs without repetition) on the (n-1)-order DeBruijn graph, which it is easier to calculate.

4.1 Application of the adjacency matrix of a digraph

Graphs can be represented by matrixes. The adjacency matrix is an matrix where each the rows and columns indicate the vertexes. If there is an arc from the node i to the node j, the (i,j) element of the matrix evaluates to 1, otherwise is 0. The adjacency matrixes of DeBruijn graphs or order 2 and 3 have the form:

Adjacency matrix for L = 2

Adjacency matrix for L = 3

When a feedback function is defined, the elements on the DeBruijn graph change so that every vertex has only one predecessor and one successor. The adjacency matrix of the Hamiltonian circuit by a function is an permutation matrix, in which each row or column has only one element equal to 1. In order to calculate these functions, we propose an iterate method. In this method, all the ones are changed by variables, imposing that every row or column must have only one '1'. For L = 3, this matrix is:

If we want a Hamiltonian circuit, there cannot be 1-length cycles, i.e., all the arcs going from one vertex to the same vertex, the (i,i) elements on the matrix, must be 0. Imposing this restriction, we have a = 0 and d = 0. In fact, we can continue extending the fact that there cannot be 2-length cycles, 3-length cycles,... , (2L-1)-length cycles by imposing that all the (i,i) elements of the matrices Mi are 0, for i¡2L.

Analysing the results, it has seen the following properties (mostly by the fact of been a permutation matrix):

the eigenvalues are the complex n-roots of the unity

4.2 Application of the incidence matrix of a digraph

A sequence of maximum period length over a register with size n is determined by a function , with . Reciprocally, with such f() we can obtain a maximum period length sequence, by adding the following conditions:

1) f() bijective 2) ,

That is, with the first condition we assure that with f is possible to reach every sequence in , so the function defined by represents a permutation over . With the second condition, we restrict f() to be a cycle; that is, its decomposition in product of disjoint cycles must have a unique term. Now, for each f() we define a digraph G with:

vertices: adjacencies:

With this definition, if f() is bijective, the disjoint cycles in the decomposition of h() are the connex components of G. Thus, the above conditions can be rewritten as:

1) f() bijective 2) G connex

Obviously, the first condition is easier to establish than the second one. To assure that G is connex we recall the following property of the incidence matrix of a digraph:

Property: If G is a digraph with order n, r connex components, and incidence matrix I, then . Applying this result to our problem we conclude:

A function has the following two properties:

1) f() bijective. 2) The rank of the incidence matrix of the digraph G associated to f() is n-1. (if and only if, the formula induces sequence of maximum period length)

With this, if we know a function f(), we have a direct and easy condition to determine if it is possible to obtain a sequence of maximum period length

from it. Also, we apply the result in order to generate a new f(): In fact, if f() is unknown, we can use the above result with a symbolic incidence matrix. That is, with a matrix defined as a function of f() evaluated in each point. Then, the condition over its rank gave set of equations in these terms, whose solutions are all the possibilities for such f().

References

[1] N.G. de Bruijn, A combinatorial problem , *Proc. Nederlanske Akademie van Wetenschappen*, **49** (1946) 758–764.

[2] A.H. Chan, R.A. Games, and E.L. Key, On the complexities of DeBruijn sequences, *J. Comb. Theory*, **33** (1982) 233–246.

[3] A.H. Chan and R.A. Games, On the Quadratic Spans of DeBruijn Sequences, *IEEE Trans. Inform. Theory*, **36-4** (1990) 822–829.

[4] C. Ding, G. Xiao and W. Shan, The Stability Theory of Stream Ciphers, *Lecture Notes in Computer Science*, **561** (1991) 822–829.

[5] C. Ding, G. Xiao and W. Shan, On the Distribution of DeBruijn Sequences of Given Complexity, *IEEE Trans. Inform. Theory*, **IT-30-4** (1984) 611–614.

[6] S. Golomb. *Shift Register Sequences*. Plenum Press (1967).

[7] J.A. Jansen and D.E. Boekee, An Efficient Algorithm for the Generation of DeBruijn Cycles, *IEEE Trans. Inform. Theory*, **37-5** (1991) 1475–1478.

[8] P. Jeavons and D.A. Cohen, Generating Binary Sequences for Stochastic Computing, *IEEE Trans. Inform. Theory*, **40-3** (1994) 716–720.

[9] R. A. Rueppel. *Analysis and Design of Stream Ciphers*. Springer-Verlag (1986).

[10] B. Schneier. *Applied Cryptography*. John Wiley and Sons (1996).

[11] T.Siegenthaler, Correlation-immunity of non-linear combining functions for cryptographic applications, *IEEE Trans. Inform. Theory*, **IT-30** (1984) 776–780.