Correlation Immunity and the Summation Generator

Rainer A. Rueppel
CMRR
University of California, San Diego
La Jolla, CA, 92093

Abstract:

It is known that for a memoryless mapping from $GF(2)^N$ into GF(2) the nonlinear order of the mapping and its correlation-immunity form a linear tradeoff. In this paper it is shown that the same tradeoff does no longer hold when the function is allowed to have memory. Moreover, it is shown that integer addition, when viewed over GF(2), defines an inherently nonlinear function with memory whose correlation-immunity is maximum. The summation generator which sums N binary sequences over the integers is shown as an application of integer addition in random sequence generation.

1. Introduction

Boolean functions from $GF(2)^n$ into GF(2) are commonly found in cryptographic applications. Usually they are designed to be nonlinear and to produce a balanced output, and, often one finds the additional requirement that from knowledge of the output bit it should not be possible to reliably guess one or more input bits. Consider for example DES, where the S-boxes define nonlinear mappings from $GF(2)^4$, (or $GF(2)^6$ respectively), into GF(2) chosen in such a way that little statistical dependency is created between the output bit and one or more input bits. Or consider a classical running-key generator for use in a stream cipher system. Such a running-key generator consists of N driving linear feedback shift registers (LFSRs) and some nonlinear function operating on the N output sequences in order to produce the running-key. Siegenthaler [1] has recently shown that several of the previously published running-key generators employed

H.C. Williams (Ed.): Advances in Cryptology - CRYPTO '85, LNCS 218, pp. 260-272, 1986.

nonlinear functions which created statistical dependencies between single input and output variables and therefore allowed 'divide-and-conquer' attacks using correlation techniques . These results stimulated some interest in functions which can resist the correlation attack. The concept of m-th order correlation-immunity for combining functions [2] was introduced as a measure of their resistance against such correlation attacks . (But correlation-immunity is not confined to running-key generators. In fact, if a boolean function is found to be m-th order correlation-immune, it means that there is no statistical dependency between the output variable and any subset of m input variables, provided the input variables are independent and uniformly distributed). Unfortunately, for such memoryless combining functions f there exists a tradeoff between the attainable nonlinear order and the attainable level of correlation-immunity [2]. If k and m denote the nonlinear order and the order of correlation-immunity of f, respectively, then

$$k + m \leqslant N-1 \qquad \text{for } 1 \leqslant m \leqslant N-2. \tag{1}$$

Thus, the more correlation-immunity, the smaller the nonlinear order of f and consequently the smaller the linear complexity of the running-key, and vice versa. Moreover, functions which satisfy (1) with equality are difficult to find.

In section 2 we shall show that this inconvenient tradeoff can be avoided by proper use of memory in the nonlinear combining function. In fact, one bit of memory suffices to obtain nonlinear combiners that are maximally correlation-immune and have maximum nonlinear order at the same time . In section 3 we shall demonstrate that integer (or real) addition, which is an extremely nonlinear operation when considered over GF(2), inherently defines a maximally correlation-immune combiner . Moreover, we will apply integer addition in random-sequence generation and give evidence that the resulting key stream is highly complex.

Throughout most of this paper, GF(2) is taken as the underlying field of computation; therefore, unless otherwise stated, formulas are assumed to be computed over GF(2).

2. Correlation-Immunity of Nonlinear Combiners

In order to investigate the statistical dependencies introduced by the nonlinear combiner itself (and not by the sources which feed it) we shall assume that the input sequences to the nonlinear combiner are sequences of independent and uniformly distributed binary random variables.

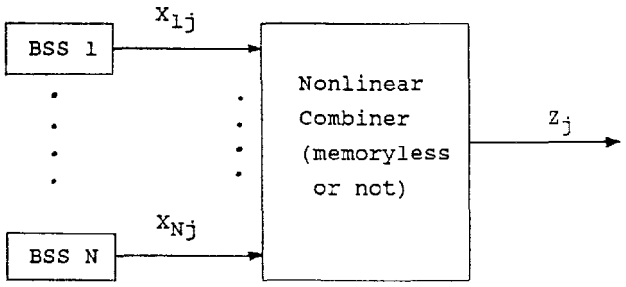

Fig. 1. Information-theoretic model used to define correlation-immunity. (BSS = Binary Symmetric Source)

Several authors ([2],[3]) investigated the correlation-immunity of nonlinear combiners, but always under the assumption that the combiner is memoryless. A memoryless nonlinear function is termed correlation-immune of order m [2] if the mutual information between the output variable and any subset of m input variables considered jointly is zero. For a memoryless combiner time is immaterial, since at any time the output only depends on the current input variables. Now allow the nonlinear combiner to contain memory which, in fact, converts it into a finite-state machine (FSM). Let S_0 denote the initial content of the combiner's memory and define $X_i^j = X_{i1}, X_{i2}, .., X_{ij}$, for $1 \leq i \leq N$. For any FSM we may write

$$Z_j = F(X_1^j, \ldots, X_N^j, S_0) \qquad (2)$$

As a natural extension of the above definition of correlation-immunity for memoryless combiners we shall say that a nonlinear combiner with memory is correlation-immune of order m if the mutual information between the output sequence and any subset of m input sequences is zero, that is, if

$$I(Z^j; X_{i_1}^j, \ldots, X_{i_m}^j) = 0 \qquad j>0 \qquad (3)$$
$$1 \leq i_1 < i_2 < \ldots < i_m \leq N$$

In this case the output sequence is statistically independent of any m input sequences considered jointly . In many cryptographic applications it is required that the output sequence should resemble as closely as possible a truly random sequence. For example, in a running-key generator, it must not be possible to reliably guess the next key bit regardless of how many prior key bits have been observed . In the information-theoretic model this corresponds to requiring that (Z_j) forms a sequence of independent and uniformly distributed random variables. Under this constraint the definition (3) is equivalent to

$$I(Z_j; X_{i_1}^j, X_{i_2}^j, \ldots, X_{i_m}^j, Z^{j-1}) = 0 \qquad j>0 \qquad (4)$$
$$1 \leq i_1 < i_2 < \ldots < i_m \leq N$$

Definition (4) is intuitively pleasing: knowing all prior output bits and knowing (or guessing) jointly any m input sequences does not provide any information whatsoever on the next output bit. To prove the equivalence of (3) and (4), let m=1 and decompose (3) in the following way,

$$I(Z^j; X^j) = I(Z^{j-1}; X^j) + I(Z_j; X^j | Z^{j-1}) = 0.$$

Mutual informations are always greater or equal to zero; hence it must hold

$$I(Z_j; X^j | Z^{j-1}) = H(Z_j | Z^{j-1}) - H(Z_j | X^j, Z^{j-1}) = 0.$$

Taking into account that $\{Z_j\}$ forms an i.i.d. sequence, we arrive at

$$I(Z_j;X^j|Z^{j-1}) = I(Z_j;X^j,Z^{j-1}) = 0$$

which establishes the equivalence. For an in-depth treatment of the different definitions of correlation-immunity we refer to [5]. Now let the function F of (2) have the form

$$Z_j = \sum_{i=1}^{N} X_{ij} + F'(X_1^{j-1},\ldots,X_N^{j-1},S_0) \qquad (5)$$

where the current input variables $X_{1j},\ldots,X_{Nj}$ are summed and added to an arbitrary function F' of all previous input variables and of the initial state S_0. Suppose we know the complete history of the nonlinear combiner F and all but one, say X_{ij}, of the current input variables. We then may rewrite (5) as

$$Z_j = X_{ij} + Y_j \qquad (6)$$

where Y_j summarizes our knowledge about the device. The fact that X_{ij} is drawn independently of Y_j from a uniform distribution implies that Z_j and Y_j are statistically independent and that Z_j is also uniformly distributed. This can also be seen from the fact that (6) corresponds to sending Y_j through a memoryless binary symmetric channel with capacity 0, thereby ensuring that Z_j is uniformly distributed and statistically independent of Y_j. Hence, <u>any nonlinear combiner of the form (5) is (N-1)st-order correlation-immune</u>, which is in fact the maximum order of immunity possible. Moreover, the function F' in (5) is not restricted in any way and may consequently be chosen to be of maximum nonlinear order. In particular, <u>one memory cell suffices in order to realize a combiner with maximum correlation-immunity and with maximum nonlinear order</u>. For this case the FSM equations may be written as

$$Z_j = \sum_{i=1}^{N} X_{ij} + S_{j-1} \tag{7a}$$

$$S_j = f(X_{1j-1},\cdots,X_{Nj-1},S_{j-1}) \tag{7b}$$

Equations (7) describe an FSM with finite memory of 1 bit. If the next state is computed irrespectively of the previous state, then the FSM (7) is said to have a finite input memory of 1 bit (it can be realized with a pure feedforward structure).
At this point it is illustrative to consider a practical example. Pless [4] proposed in 1977 a running-key generator which contains as basic building block a 2-LFSR-subgenerator. In this subgenerator a J-K Flip-Flop acts as the nonlinear element which combines the 2 LFSR sequences. A J-K Flip-Flop defines a one-bit FSM whose state just contains the previous output bit and whose output and next-state functions therefore coincide. Its behavior is completely described by

$$Z_j = X_{1j} + Z_{j-1}(1 + X_{1j} + X_{2j}) \tag{8}$$

Considering X_{1j}, X_{2j}, and Z_{j-1} as the 3 input variables to a memoryless mapping f defined by (8) we may compute the Walsh transform $S_f(w)$ [3] of f. Fig. 2 displays the result

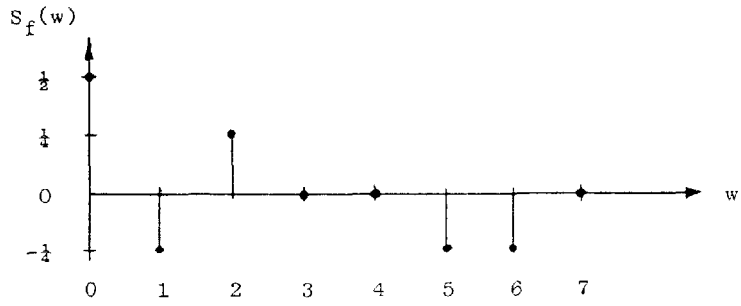

Fig. 2. Walsh transform of the boolean mapping defined by (8)

The graph of Fig. 2 may be interpreted as follows: let (w_0,w_1,w_2) denote the binary representation of w, where $0 \leq w \leq 7$. If the Walsh transform $S_f(w)$ is nonzero at some $w > 0$ then the mutual information $I(Z_j;w_0X_{1j}+w_1X_{2j}+w_2Z_{j-1})$ is greater than zero. Moreover, the value of the Walsh transform at this w gives an exact account of how much statistical dependency is introduced. For instance, the peaks in the Walsh transform at w = 1 and w = 2 in Fig. 2 tell us that the output bit Z_j is neither independent of X_{1j} nor of X_{2j}. The value -1/4 at $S_f(1)$ tells us that the probability that Z_j coincides with X_{1j} is 3/4. Equivalently, the value +1/4 at $S_f(2)$ tells us that the probability that Z_j coincides with X_{2j} is 1/4. On the other hand, since $S_f(4)$ is zero Z_j is independent of Z_{j-1}. Consequently, if a J-K Flip-Flop is fed by two binary symmetric sources it will produce a sequence of independent and uniformly distributed binary random variables (as desired), but this output sequence will exhibit a strong correlation with either input sequence. Thus, the correlation-immunity of a J-K Flip-Flop is zero.

Comparing (2) and (4) we notice that maximum correlation-immunity of F was obtained by separating the N current input variables from an arbitrary function F' of only prior input variables. In general, any desired level m of correlation-immunity can be obtained by separating m+1 input variables each taken from a different input sequence and possibly with a different time index, but disallowing their use in the arbitrary function F'.

3. The Summation Principle

Let a and b be two integers, whose binary representation is given as $a=a_{n-1}2^{n-1}+\ldots+a_1 2+a_0$ and $b=b_{n-1}2^{n-1}+\ldots+b_1 2+b_0$, respectively. Let z = a + b be the real sum of the two integers and assume that the sum is computed bit-serially in GF(2) from the binary representations of a and b. Then we may write, with increasing nonlinear order of the binary functions producing the j-th bit

$$
\begin{aligned}
z_0 &= a_0 + b_0 \\
z_1 &= a_1 + b_1 + a_0 b_0 \\
z_2 &= a_2 + b_2 + a_1 b_1 + a_1 a_0 b_0 + b_1 a_0 b_0 \\
&\cdot
\end{aligned}
\tag{9}
$$

or, we may express z_j recursively for $0 \leqslant j \leqslant n$

$$z_j = f_1(a_j, b_j, c_{j-1}) = a_j + b_j + c_{j-1} \tag{10a}$$

$$c_j = f_2(a_j, b_j, c_{j-1}) = a_j b_j + (a_j + b_j) c_{j-1} \tag{10b}$$

where c_{j-1} represents the carry-bit from the less significant bits to bit j of the sum. Fig. 3 illustrates the principle.

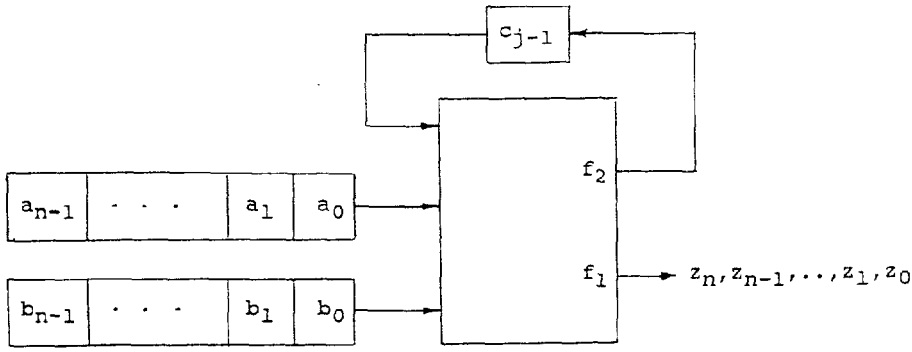

Fig. 3. Time-sharing of a 3-bit adder to produce bit-serially the real sum of two n-bit integers.

When the two input shift registers in Fig. 3 are initially loaded with the binary representation (least-significant bit first) of the two integers and when the feedback memory cell is initially zero, then after (n+1) clock cycles the (n+1) bits corresponding to the binary representation of the real sum will have appeared serially at the output. In fact, the real adder of Fig. 3 defines a finite-state machine with output and next-state functions according to (10), and, surprisingly enough, it directly realizes a correlation-immune combining function as defined in (7). Note that f_1 defines the GF(2)-sum of the input variables and thus accounts for the correlation immunity, while f_2 defines the GF(2)-sum of all second-order products of the input variables and thus implements a

memoryless nonlinear mapping. The memory-cell is used to hold the carry-bit from the (j-1)-st to the j-th position of the sum and carries all the nonlinear influence of the less significant bits. These observations suggest that real addition could be useful in running-key generation. The simplest running-key generator based on this summation principle may be obtained by adding two (or in general N) infinite integers whose binary representations are periodic sequences generated by suitable LFSRs. We shall call any such generator a summation-generator. It is apparent from the linear form of the output function (10a) that whenever at least one input sequence consists of independent and uniformly distributed random variables so will also the output sequence. Besides statistical properties of a generator one is often interested in the period of a generator and its linear complexity (that is, the length of the shortest LFSR that is able to emulate the generator for a given output sequence).

Property 1:

Let $\{a_j\}$ and $\{b_j\}$ be two binary sequences with least periods T_1 and T_2 respectively. When $\{z_j\}$ denotes the real sum of $\{a_j\}$ and $\{b_j\}$, expressed in radix-2 form, and if $\gcd(T_1,T_2)=1$, then $\{z_j\}$ has least period T_1T_2.

Proof:

Define the rational fraction s associated to a sequence $\{s_j\}$ of period T as

$$s = \frac{\sum_{j=1}^{T} s_{T-j}2^{-j}}{2^T-1} = \frac{p}{q}$$

where $\gcd(p,q)=1$. The period T may be found from q as the multiplicative order of 2 modulo q. Therefore we may write $a=p_1/q_1$ and $b=p_2/q_2$. The real sum of the sequences directly corresponds to the real sum of the rational fractions. Thus

$$a + b = \frac{p_1}{q_1} + \frac{p_2}{q_2} = \frac{p_1q_2+p_2q_1}{q_1q_2} = c + \frac{n}{q_1q_2}$$

where we identify n/q_1q_2 as the rational fraction representing the real sum sequence $\{z_j\}$ and c, which is either 0 or 1, as the carry digit from one period of the sum sequence to the next. We note that $\gcd(n,q_1q_2)=1$ because $\gcd(q_1,q_2) = \gcd(p_1,q_1) = \gcd(p_2,q_2)=1$, and that $\gcd(2,q_1q_2)=1$ since q_1 divides $2^{T_1}-1$ and q_2 divides $2^{T_2}-1$. Then the period T of the real sum sequence $\{z_j\}$ is given by the multiplicative order of 2 modulo q_1q_2. Since q_1 and q_2 are relatively prime it follows from the Chinese remainder theorem that T is equal to the product of the multiplicative orders of 2 modulo q_1 and 2 modulo q_2. Hence $T=T_1T_2$.

Property 1 may easily be generalized to the sum of N periodic sequences in radix-r representation.

Now assume that the real adder is fed by two maximum-length sequences whose minimal polynomials have relatively prime degree L_1 and L_2. This implies that their periods are relatively prime and thus, by property 1, the period T of the real sum sequence is $(2^{L_1}-1)(2^{L_2}-1)$ which value also provides an upper bound to the linear complexity of $\{z_j\}$. When the above two m-sequences are multiplied termwise then the resulting product sequence will have a minimal polynomial of degree L_1L_2 (i.e. linear complexity L_1L_2) all of whose roots are from $GF(2^{L_1L_2}) - GF(2^{L_1}) - GF(2^{L_2})$. The interesting question now is how the feedback memory of the real adder affects the linear complexity of the real sum sequence. From (9) we see that the order of the products involved in the direct description of the function producing z_j grows linearly with time. A finite-state machine is said to have finite input memory M if M is the least integer such that the output digit at time j may be expressed as a function of the input variables at times $j-M,\ldots,j-1,j$. Clearly the FSM as described by (10) has in general infinite input memory. But whenever the input sequences to the real adder produce a pair of zeros or ones, then the state of the adder FSM is set to a value independent of the preceding states and input values. In particular, when periodic input sequences are used which produce at least a common pair of zeros or ones within the period of the output sequence (which is certainly true for the above pair of m-sequences), then the input memory M will be finite with respect to the particular driving sequences. This allows to convert the feedback structure of the nonlinear combiner (10) into a feedforward structure of input memory M (corresponding to a maximum

nonlinear order of M+1 in the functional description (9)). From the feedforward function it is then possible to calculate (or at least bound) the associated linear complexity of the output sequence. In fact, one may prove that real addition of binary sequences is so nonlinear that from the available L_1 elements in $GF(2^{L_1})$ and L_2 elements in $GF(2^{L_2})$, (which are the roots of the two primitive minimal polynomials), it may generate every element in $GF(2^{L_1 L_2})$ - $GF(2^{L_1})$ -$GF(2^{L_2})$.

Property 2:

Let $\{a_j\}$ and $\{b_j\}$ be two binary m-sequences whose primitive minimal polynomials have relatively prime degrees L_1 and L_2. When $\{a_j\}$ and $\{b_j\}$ are added over the reals then the real sum sequence $\{z_j\}$ exhibits linear complexity LC close to its period length,

$$LC(\{z_j\}) \leqslant (2^{L_1}-1)(2^{L_2}-1) \qquad (11)$$

with near equality.

Instead of giving the proof which is straightforward but rather tedious, we will display some simulation results [6] which confirm that the bound (11) is extremely tight. In fact, no serious degeneracy was ever found, which suggests that integer addition is an inherently good nonlinear function.

N	L_1	L_2	L_3	T	LC
2	3	4		105	⩾100
	3	5		217	⩾208
	4	5		465	⩾455
3	2	3	5	651	⩾641

Table 1. Small-scale simulations giving evidence that the bound (11) is very tight (for explanation of the table see text below).

In table 1, the column labeled N gives the number of m-sequences

added over the reals; the columns labeled L_1, L_2, and L_3 give the degrees of the minimal polynomials of the m-sequences which were added; the column labeled T gives the period of the sum sequences; the column labeled LC displays the smallest linear complexity obtained for all possible combinations of different primitive minimal polynomials of the mentioned degrees. For instance, the first row tells us that each of the different m-sequences of degree 3 (there are 2) was separately added to each of the different m-sequences of degree 4 (there are 2), and never was a linear complexity of smaller than 100 obtained.

Although it seems counterintuitive, integer (real) addition is extremely nonlinear when viewed over GF(2). The results in this section show that given two integers whose binary representations have very low (linear) complexity then their real sum may have very high (linear) complexity. This of course depends whether use was made of the nonlinear potential of real addition. Suppose, for example, we add the two integers whose binary representations are the sequences 0101.. and 1010.. each having linear complexity 2. The result is the all-1 sequence of linear complexity 1. Note that in this case the real sum and the mod-2 sum of the 2 sequences are identical and, in fact, never a nonlinear contribution through a carry occurred.

Finally, we want to mention that a similar analysis applies to the 0/1-knapsack with N weights [7]. The i-th output bit of such a knapsack may be regarded as being produced by a boolean function from $GF(2)^N$ into GF(2) whose coefficients are determined by the weights of the knapsack. One can prove that the nonlinear order of the function producing the i-th output bit is bounded form above by $\min\{2^i,N\}$. Therefore, roughly the logN least significant bits of a knapsack are considerably less nonlinear (are considerably weaker) then the remaining output bits.

References:

[1] T. Siegenthaler," Decrypting a Class of Stream Ciphers Using Ciphertext Only", IEEE Trans. on Computers, Vol. C-33, 1984.

[2] T. Siegenthaler, "Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications", IEEE Trans. on Info. Th., Vol. IT-31,1985.

[3] Xiao Guo-zhen, J.L. Massey, "A Spectral Characterization of Correlation-Immune Combining Functions", submitted to IEEE Trans. on Info. Th.

[4] V.S. Pless, "Encryption Schemes for Computer Confidentiality", IEEE Trans. on Computers, Vol. C-26, Nov. 1977.

[5] T. Siegenthaler, "Design of Combiners to Prevent Divide and Conquer Attacks", Proceedings of Crypto 85, Santa Barbara, August 18-22, 1985.

[6] U. Maurer, R. Viscardi, "Running-Key Generators with Memory in the Nonlinear Combining Function", Diploma Project, Swiss Federal Institute of Technology Zurich, Dec. 1984.

[7] R.A. Rueppel, J.L. Massey, "The Knapsack as a Nonlinear Function", IEEE Symposium on Info. Th., Brighton, UK, June 24-28, 1985.