been trivial if $\log_t h$ was known to A. But notice that h and t are chosen by B, thus A has to compute a discrete logarithm. This is infeasible by our assumption that exponentiation mod p is a one-way function.

Another possibility for A to cheat is to choose x of the form

$$x = \frac{p-1}{m} \cdot j, \qquad j = 0,1,\cdots,m-1, \tag{1}$$

where m is a product of some *small* prime factors of $p-1$. A then can find an $x' = ((p-1)/m)\cdot j'$ such that $h^x = t^{x'} \pmod p$, by merely searching over the m possible values for j'. This attempted swindle by A is a violation of the requirement at step 3) that x and $p-1$ be coprime. It will be easily detected by B at step 5) of the protocol when x is revealed. Since the factorization of $p-1$ is known, B can do even better. As soon as B receives h^x (or t^x), he can check that it is a primitive element, an equivalent to the coprimality requirement.

If we did not require that h and t both be primitive elements, then B could cheat as follows. He could choose $h = t^2 \pmod p$ (p is an odd prime) with t primitive. Then the odd powers of t would never occur if h was selected in step 3) of the protocol. Thus, whenever B found that y was an odd power of t, he would surely win. (Checking whether a number is an even or odd power of a primitive element is easy, as shown in [5].)

III. Comments

Suppose the two parties wish to play more than one game. They can use the same functions f_h and f_t with a different argument x chosen by A. The accumulating knowledge of pairs $(x, f_h(x))$ does not make the solution of $y = f_h(x)$ (for other values of y) feasible. (Otherwise, A can produce such pairs before the game starts and will be able to invert f_h, a contradiction to our first assumption on f_h and f_t.) Contrast this with the "oblivious transfer," where with probability 1/2 a game is ended when A's number is factored. Thus, on the average, new values should be computed every other game.

Our protocol also differs from the "oblivious transfer" by the fact that no party can force himself to lose or reduce his chances to win once f_h and f_t are determined.

Acknowledgment

The authors are thankful to Martin E. Hellman for helpful discussions.

References

[1] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol. IT-22, pp. 644–654, Nov. 1976.
[2] A. Shamir, R. Rivest, and L. Adleman, "Mental poker," MIT/LCS/TM-125, Feb. 1979.
[3] M. O. Rabin, "How to exchange secrets by oblivious transfer," preprint.
[4] M. Blum, "Coin flipping by telephone," presented at IEEE Workshop on Communications Security (Crypto-81), Santa Barbara, CA, Aug. 23–26, 1981.
[5] S. Pohlig and M. Hellman, "An improved algorithm for computing logarithms over GF(p) and its cryptographic significance," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 106–110, Jan. 1978.
[6] National Bureau of Standards, "Notice of a proposed federal information processing data encryption standard," *Fed. Register*, vol. 40, no. 12134, March 17, 1975.

Correlation-Immunity of Nonlinear Combining Functions for Cryptographic Applications

T. SIEGENTHALER

Received November 28, 1983; revised February 23, 1984
The author is with the Institute for Communication Technology, ETH-Zentrum, 8092 Zurich, Switzerland.

***Abstract*—Pseudonoise generators for cryptographic applications consisting of several linear feedback shift registers with a nonlinear combining function have been proposed as running key generators in stream ciphers. These running key generators can sometimes be broken by (ciphertext-only) correlation attacks on individual subsequences. A new class of combining functions is presented, which provides better security against such attacks. The security is quantified by the smallest number $m+1$ of subsequences that must be simultaneously considered in a correlation attack. A necessary condition for such mth-order correlation-immunity is proved. A recursive construction is given that permits the construction of an mth-order immune combining function for n subsequences for any m and n with $1 \le m < n$. Finally, the trade-off between the length of the linear equivalent of the nonlinear generator and the order m of its immunity against correlation attacks is considered.**

I. Introduction

In conventional cryptography, pseudonoise generator structures like those given in Fig. 1 have been proposed [3], [4] as running key generators in stream ciphers. The n subgenerators S_i, $i = 1,2,\cdots,n$, are usually realized by linear feedback shift registers, and f is a memoryless (nonlinear) combining function. The key K_i determines the initialization of the subgenerator S_i. We assume that the secret key of the generator consists of the n keys, $K_1, K_2, \cdots, K_n$, of the subgenerators. Everything about the generator except these keys is known to the cryptanalyst. The number M of different keys for G is

$$M = \prod_{i=1}^{n} M_i,$$

where M_i is the number of different subkeys K_i for the subgenerator S_i.

The purpose of the nonlinear combining function f in Fig. 1 is to make the keystream difficult for the cryptanalyst to predict. Ideally, the cryptanalyst would be forced to try an average of half of the M possible keys before finding the actual key that produces the observed keystream sequence. However, if the function f is not properly chosen, a cryptanalyst may make a selective attack on each subkey K_i; this can be performed by correlating the ciphertext with the sequence generated by subgenerator S_i for each choice of K_i [1], [2]. The subkey K_i of S_i will be found after at most M_i trials. For the case where the sequence generated by each subgenerator S_i is correlated with the ciphertext, at most

$$M' = \sum_{i=1}^{n} M_i \ll M$$

keys have to be tested to obtain the whole key.

In some cases there may be no correlation between any subgenerator sequence and the keystream sequence, but there may be a statistical dependence between pairs (or triples, etc.) of subgenerator sequences and the keystream sequence. In this case a more laborious correlation attack, in which pairs (or triples, etc.) of subkeys are tested, can still be made. In general, to make the generator structure of Fig. 1 resistant to a correlation attack, one should ensure that there is no statistical dependence between any small subset of the n subgenerator sequences and the keystream sequence. In the next section we give a precise definition of such immunity to a correlation attack. We then develop a necessary condition on the function f to provide such immunity, and we show how to construct such correlation-immune functions for any number of subsequences.

II. Correlation-Immune Functions

To make our problem mathematically tractable (but still practically useful), we assume hereafter that the n subgenerators in

0018-9448/84/0900-0776$01.00

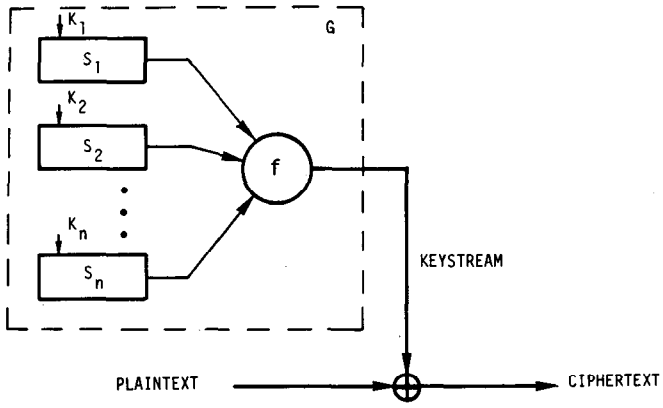

Fig. 1. Cryptographic generator (G) consisting of n subgenerators in stream cipher.

Fig. 1 are independent binary symmetric sources, i.e., that the sequence $X_{i1}, X_{i2}, X_{i3}, \cdots$ produced by subgenerator S_i is a sequence of independent identically distributed (i.i.d.) binary random variables equally likely to be zeros or ones and independent of the sequences produced by the other subgenerators. The keystream sequence $Z_1, Z_2, Z_3, \cdots$ is then determined as

$$Z_j = f(X_{1j}, X_{2j}, \cdots, X_{nj}),$$

where f is the memoryless (nonlinear) combining function of the generator. It follows from our assumptions about the subgenerator sequences that the keystream sequence $Z_1, Z_2, Z_3, \cdots$ is also i.i.d. but not necessarily balanced in the sense that $P(Z_j = 1) = 1/2$.

Let $\boldsymbol{X}_j = (X_{1j}, X_{2j}, \cdots, X_{nj})$ be the n-tuple of subgenerator output digits at time j. We shall say that the combining function f is mth-order correlation-immume if every m-tuple obtained by choosing m components from $\boldsymbol{X}_j$ is statistically independent of Z_j for all $j = 1, 2, 3, \cdots$. By the time invariance of the system and by the i.i.d. nature of all sequences, this is equivalent to stating that every subset of m random variables chosen from $X_1, X_2, \cdots, X_n$ is statistically independent of

$$Z = f(X_1, X_2, \cdots, X_n) \tag{1}$$

when $X_1, X_2, \cdots, X_n$ are balanced i.i.d. binary random variables. Thus f is mth-order correlation-immune if and only if for each choice of indices $i_1, i_2, \cdots, i_m$ with $1 \le i_1 < i_2 < \cdots < i_m \le n$, the random variable Z of (1) is statistically independent of the random vector $(X_{i_1}, X_{i_2}, \cdots, X_{i_m})$. This condition is, of course, equivalent to the condition

$$I(X_{i_1}, X_{i_2}, \cdots, X_{i_m}; Z) = 0 \tag{2}$$

on mutual information, and in fact, it was the intuitive content of (2) that led us to our definition of correlation-immunity.

III. A Necessary Condition for Correlation Immunity

Any binary-valued function f of n binary random variables can be written in its "algebraic normal form", i.e., as the GF(2) sum of products

$$\begin{aligned} f(x_1, x_2, \cdots, x_n) &= a_0 + a_1 x_1 + a_2 x_2 + \cdots + a_n x_n + a_{12} x_1 x_2 + a_{13} x_1 x_3 \\ &\quad + \cdots + a_{12\cdots n} x_1 x_2 \cdots x_n. \end{aligned} \tag{3}$$

Moreover, each binary coefficient, say $a_{12\cdots k}$, is determined by the inversion formula

$$a_{12\cdots k} = \sum_{\boldsymbol{x} \in S_{12\cdots k}} f(x_1, x_2, \cdots, x_n), \tag{4}$$

where $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ and

$$S_{12\cdots k} = \begin{cases} \{\boldsymbol{x}: x_{k+1} = x_{k+2} = \cdots = x_n = 0\}, \\ \qquad \text{for } k \in \{1, 2, \cdots, n-1\} \\ \{\boldsymbol{x}\}, \qquad \text{for } k = n \end{cases}. \tag{5}$$

Equation (4) follows from the fact that the product $x_{i_1} x_{i_2} \cdots x_{i_j}$, where $1 \le i_1 < i_2 < \cdots < i_j \le n$, vanishes for all $\boldsymbol{x}$ in $S_{12\cdots k}$ unless $i_j \le k$, and, for $i_j \le k$, fails to vanish for exactly 2^{k-j} values of $\boldsymbol{x}$ in $S_{12\cdots k}$; thus, the product $x_{i_1} x_{i_2} \cdots x_{i_j}$ summed over $\boldsymbol{x}$ in $S_{12\cdots k}$ equals one if and only if $x_{i_1} x_{i_2} \cdots x_{i_j} = x_1 x_2 \cdots x_k$.

We will say that a certain product term, e.g., $x_1 x_3 x_4$, is present in the algebraic normal form (3) of f if the corresponding binary coefficient, in our example a_{134}, has value one. We now show that the presence of certain product terms in the algebraic normal form of the combining function f in Fig. 1 is incompatible with correlation immunity.

Theorem 1: If $f(x_1, x_2, \cdots, x_n)$ is mth-order correlation immune, where $1 \le m < n$, then no product of $n - m + 1$ or more variables can be present in the algebraic normal form (3) of f. Moreover, if

$$P[f(X_1, X_2, \cdots, X_n) = 1] = P[f(X_1, X_2, \cdots, X_n) = 0]$$

when $X_1, X_2, \cdots, X_n$ are balanced i.i.d. binary random variables, then no product of $n - m$ variables can be present in the algebraic normal form of f unless $m = n - 1$.

Proof: Let $Z = f(\boldsymbol{X})$ as in (1) and let $X_1, X_2, \cdots, X_n$ be balanced i.i.d. binary random variables. Let $S_{12\cdots k}$ be defined by (5) and let

$$N_{12\cdots k} = \#\{\boldsymbol{x}: \boldsymbol{x} \in S_{12\cdots k} \text{ and } f(\boldsymbol{x}) = 1\}, \tag{6}$$

where $\#(\cdot)$ denotes the cardinality of the enclosed set. Then it follows from (5) and (6) that

$$P(Z = 1 | X_{k+1} = X_{k+2} = \cdots = X_n = 0) = \frac{N_{12\cdots k}}{2^k}, \quad \text{for } k = 1, 2, \cdots, n-1, \tag{7a}$$

$$P(Z = 1) = \frac{N_{12\cdots n}}{2^n}. \tag{7b}$$

Suppose that f is mth-order immune. Then, because the probability on the left of (7a) is conditioned on the values of $n - k$ of the random variables $X_1, X_2, \cdots, X_n$, it follows that

$$P(Z = 1 | X_{k+1} = X_{k+2} = \cdots = X_n = 0) = P(Z = 1), \quad \text{for } n - m \le k \le n - 1 \tag{8}$$

and hence from (7a) and (7b) that

$$\frac{N_{12\cdots n}}{2^n} = \cdots = \frac{N_{12\cdots k}}{2^k} = \cdots = \frac{N_{12\cdots (n-m)}}{2^{n-m}}, \quad \text{for } n - m \le k \le n$$

and therefore

$$N_{12\cdots k} = 2^{k-(n-m)} \cdot N_{12\cdots (n-m)}, \qquad \text{for } n - m \le k \le n. \tag{9}$$

However (9) shows that $N_{12\cdots k}$ is even for $n - m + 1 \le k \le n$; thus, (4) and (6) imply that

$$a_{12\cdots k} = 0, \qquad \text{for } n - m + 1 \le k \le n.$$

But the above argument clearly applies to any k components of

$\boldsymbol{x}$, not only to the first k, so that

$$a_{i_1 i_2 \cdots i_k} = 0, \qquad \text{for } n-m+1 \le k \le n \tag{10}$$

for any $1 \le i_1 < i_2 < \cdots < i_k \le n$, as was to be shown. It remains to show that (10) also holds for $k = n-m$ when $P(Z=1) = P(Z=0)$. But in this case $P(Z=1) = 1/2$ so that (7a) and (8) give

$$N_{12\cdots(n-m)} = 2^{n-m-1},$$

which shows that $N_{12\cdots(n-m)}$ must be even unless $m = n-1$. Thus, for $m \neq n-1$, (4) and (6) imply that

$$a_{12\cdots(n-m)} = 0.$$

Again the argument clearly applies to any k components of $\boldsymbol{x}$, so that $m \neq n-1$ implies

$$a_{i_1 i_2 \cdots i_{n-m}} = 0$$

for all $1 \le i_1 < i_2 < \cdots < i_{n-m} \le n$, as was to be shown.

Remark: The above proof shows that, for $m = n-1$, all products of order $n-m=1$ must appear in f. Thus, the only possible $(n-1)$st order correlation-immune functions are

$$f(\boldsymbol{X}) = X_1 + X_2 + \cdots + X_n + c,$$

where $c=1$ or $c=0$. As will be seen later, these functions are indeed $(n-1)$st order correlation-immune.

Example: It follows from Theorem 1 that the function

$$f(x_1, x_2, \cdots, x_5) = x_1 + x_2 + x_4 + x_3x_5 + x_4x_5,$$

which gives $P[f(X_1, X_2, \cdots, X_5) = 1] = 1/2$, cannot be third-order correlation immune because $m=3$ gives $n-m = 5-3 = 2$, which is incompatible with the existence of second-order products in the algebraic normal form of f. We shall see in the next section that this function is, however, second-order correlation immune.

IV. Construction of Correlation-Immune Functions

In this section we show, for any given integers m and n with $1 \le m < n$, how to construct a balanced-output mth-order correlation-immune combining function of n binary variables in which (when $m < n-1$) product terms of order $n-m-1$ are present, which is the maximum order possible according to Theorem 1.

Again we write $\boldsymbol{X} = (X_1, X_2, \cdots, X_n)$, where $X_1, X_2, \cdots, X_n$ are balanced i.i.d. binary random variables. Let f_1 and f_2 be any binary-valued functions of n binary random variables and define the random variables Z_1 and Z_2 by $Z_1 = f_1(\boldsymbol{X})$ and $Z_2 = f_2(\boldsymbol{X})$. Our construction will be based on the following result.

Theorem 2: If f_1 and f_2 are mth-order correlation-immune functions of n binary variables such that $P(Z_1=1) = P(Z_2=1) = p$, then the binary-valued function f of $n+1$ binary random variables defined by the GF(2) expression

$$f(X_1, X_2, \cdots, X_{n+1}) = X_{n+1} f_1(\boldsymbol{X}) + (X_{n+1}+1) f_2(\boldsymbol{X}) \tag{11}$$

is also mth-order correlation immune and gives

$$P[f(X_1, X_2, \cdots, X_{n+1}) = 1] = p.$$

Proof: Suppose that f is defined by (11) and that f_1 and f_2 satisfy the hypothesis of Theorem 2. Because $Z_1 = f_1(\boldsymbol{X})$ and $Z_2 = f_2(\boldsymbol{X})$ are independent of X_{n+1}, it follows that

$$\begin{aligned} P(Z_i = 1 | X_1 = x_1, X_2 = x_2, \cdots, X_m = x_m, X_{n+1} = x_{n+1}) &= P(Z_i = 1 | X_1 = x_1, \cdots, X_m = x_m) \\ &= P(Z_i = 1), \qquad i = 1, 2, \end{aligned} \tag{12}$$

for any choice of $x_1, x_2, \cdots, x_m$ and x_{n+1}. Writing $Z = f(X_1, X_2, \cdots, X_n, X_{n+1})$, we see from (11) and (12) that

$$P(Z=1 | X_1 = x_1, X_2 = x_2, \cdots, X_m = x_m, X_{n+1} = 1) = P(Z_1 = 1), \tag{13a}$$

$$P(Z=1 | X_1 = x_1, X_2 = x_2, \cdots, X_m = x_m, X_{n+1} = 0) = P(Z_2 = 1). \tag{13b}$$

But $P(Z_1=1) = P(Z_2=1) = p$ by hypothesis, so (13a) and (13b) imply

$$P(Z=1 | X_1 = x_1, X_2 = x_2, \cdots, X_m = x_m, X_{n+1} = x_{n+1}) = P(Z=1) = p$$

for any choice of $x_1, x_2, \cdots, x_m$ and x_{n+1}. The above argument clearly applies to any m of the random variables $X_1, X_2, \cdots, X_n$, not only to the first m. Thus,

$$P(Z=1 | X_{i_1} = x_{i_1}, \cdots, X_{i_m} = x_{i_m}, X_{n+1} = x_{n+1}) = P(Z=1)$$

for any $1 \le i_1 < i_2 < \cdots < i_m \le n$ and any choice of $x_{i_1}, x_{i_2}, \cdots, x_{i_m}$ and x_{n+1}. In particular, this implies that

$$P(Z=1 | X_{j_1} = x_{j_1}, \cdots, X_{j_m} = x_{j_m}) = P(Z=1)$$

for any $1 \le j_1 < j_2 < \cdots < j_m \le n+1$, and shows that f is indeed mth-order correlation-immune, as was to be shown.

The construction of an mth-order correlation-immune function of n variables can start with the following two mth-order correlation-immune functions of the first $m+2$ variables having

$$P(Z_1 = 1) = P(Z_2 = 1) = 1/2:$$

$$f_1(X_1, X_2, \cdots, X_{m+2}) = X_1 + X_2 + \cdots + X_m + X_{m+1}$$

and

$$f_2(X_1, X_2, \cdots, X_{m+2}) = X_1 + X_2 + \cdots + X_m + X_{m+2}.$$

Then, using Theorem 2, a new function $f(X_1, X_2, \cdots, X_{m+3})$ of $m+3$ binary variables is found. A second mth-order immune function of $m+3$ variables (for the next recursion step) can be obtained by permuting the variables of $f(X_1, X_2, \cdots, X_{m+3})$ such that the highest order (in this case the second order) terms in the second function do not completely coincide with those of the first function. These terms are combined to produce an mth-order correlation-ummune function of $m+4$ variables, etc.

Example: $m=2$, $n=7$.

Step 1 (initialization):

$$f_1(X_1, X_2, X_3, X_4) = X_1 + X_2 + X_3,$$

$$f_2(X_1, X_2, X_3, X_4) = X_1 + X_2 + X_4.$$

Step 2:

$$\begin{aligned} f_1(X_1, \cdots, X_5) &= X_5 f_1(X_1, \cdots, X_4) + (X_5+1) f_2(X_1, \cdots, X_4) \\ &= X_1 + X_2 + X_4 + X_3X_5 + X_4X_5. \end{aligned}$$

We then choose the permutation: $1 \to 2$, $2 \to 3$, $3 \to 4$, $4 \to 5$, $5 \to 1$ and produce

$$f_2(X_1, \cdots, X_5) = X_2 + X_3 + X_5 + X_1X_4 + X_1X_5.$$

Step 3:

$$\begin{aligned} f_1(X_1, \cdots, X_6) &= X_6 f_1(X_1, \cdots, X_5) + (X_6+1) f_2(X_1, \cdots, X_5) \\ &= X_2 + X_3 + X_5 + X_1X_4 + X_1X_5 + X_1X_6 \\ &\quad + X_3X_6 + X_4X_6 + X_5X_6 + X_1X_4X_6 \\ &\quad + X_1X_5X_6 + X_3X_5X_6 + X_4X_5X_6. \end{aligned}$$

We then choose the permutation: $1 \to 3$, $2 \to 4$, $3 \to 5$, $4 \to 6$,

$5 \rightarrow 1$, $6 \rightarrow 2$ and produce

$$f_2(X_1,\cdots,X_6) = X_4 + X_5 + X_1 + X_3X_6 + X_1X_3 + X_2X_3 + X_2X_5 + X_2X_6 + X_1X_2 + X_2X_3X_6 + X_1X_2X_3 + X_1X_2X_5 + X_1X_2X_6.$$

Step 4:

$$\begin{aligned} f_1(X_1,\cdots,X_7) &= X_7f_1(X_1,\cdots,X_6) + (X_7+1)f_2(X_1,\cdots,X_6) \\ &= X_1 + X_4 + X_5 + X_1X_2 + X_1X_3 + X_2X_3 \\ &\quad + X_2X_5 + X_2X_6 + X_3X_6 + X_1X_7 + X_2X_7 \\ &\quad + X_3X_7 + X_4X_7 + X_1X_2X_3 + X_1X_2X_5 \\ &\quad + X_1X_2X_6 + X_2X_3X_6 + X_1X_2X_7 \\ &\quad + X_1X_3X_7 + X_1X_4X_7 + X_1X_5X_7 \\ &\quad + X_1X_6X_7 + X_2X_3X_7 + X_2X_5X_7 \\ &\quad + X_2X_6X_7 + X_4X_6X_7 + X_5X_6X_7 \\ &\quad + X_1X_2X_3X_7 + X_1X_2X_5X_7 \\ &\quad + X_1X_2X_6X_7 + X_1X_4X_6X_7 \\ &\quad + X_1X_5X_6X_7 + X_2X_3X_6X_7 \\ &\quad + X_3X_5X_6X_7 + X_4X_5X_6X_7. \end{aligned}$$

This is our desired second-order correlation-immune function of seven variables containing product terms of order $n - m - 1 = 4$. An mth-order correlation-immune function resulting from this construction contains products of each order up to the maximum order according to Theorem 1, if for each permutation used, the highest order products of f_2 are (at least partly) different from those of f_1. The greatest possible number of highest order products in the function f is obtained with the above procedure, if the used permutation produces the minimum possible number of coincidences in the highest order products of f_2 and f_1.

V. Linear Equivalence and Correlation Immunity

In cryptographic applications the length (number of stages) of the shortest equivalent linear feedback shift register of a considered nonlinear generator has to be large. Nonlinear operations on (sub) generator output sequences can greatly increase the length of the linear equivalent [5], [6]. The analysis done by Key [5] shows that a sequence generated by multiplying the sequences of t pn-generators has a shortest linear equivalent of length equal to or less than the product of the lengths of the t pn-generators; moreover, equality holds if the length of the component shift registers are pairwise relatively prime. Therefore, to obtain a large linear equivalent for the pn-generator of Fig. 1, the combining function f should contain high order products. On the other hand, for a high correlation immunity the combining function should not contain high order products, according to Theorem 1. From the function $f(X_1,\cdots,X_n) = X_1X_2\cdots X_n$, for example, a large linear equivalent can be obtained; however, the statistics of the output sequence is poor and the generator is not correlation immune. On the contrary, the function $f(X_1,\cdots,X_n) = X_1 + X_2 + \cdots + X_n$ has the maximum possible order $n - 1$ of correlation immunity; however, the resulting length of the linear equivalent is not greater than the sum of the linear equivalents of the n subgenerators. This leads to a trade-off between the length of the linear equivalent and the order of correlation immunity of the nonlinear pn-generator shown in Fig. 1.

VI. Generalization to Nonbalanced Sequences

Up to now the n subgenerators in Fig. 1 have been assumed to be independent binary sources, generating sequences of i.i.d. binary random variables with a balanced distribution. To prove Theorem 2, however, no assumptions concerning the distribution of the random variables have been made. Theorem 2 holds whenever the two functions $Z_1 = f_1(X_1, X_2,\cdots, X_n)$ and $Z_2 = f_2(X_1, X_2,\cdots, X_n)$ are mth order correlation immune and $P(Z_1 = 1) = P(Z_2 = 1)$. The probabilities $0 \le P(X_{n+1} = 1) \le 1$, $0 \le P(X_{n+2} = 1) \le 1,\ldots$ can be arbitrary numbers. Hence, the definition of correlation immunity can be generalized. The combining function f is mth-order correlation immune if every m-tuple obtained by choosing m random variables from $X_1, X_2,\cdots, X_n$ is statistically independent of

$$Z = f(X_1, X_2,\cdots, X_n),$$

where $X_1, X_2,\cdots, X_n$ are independent binary random variables, not all of them necessarily balanced. As an example, consider the following set of functions:

$$Z' = X_1 + X_2 + \cdots + X_{m+1} + h(X_{m+2},\cdots, X_n),$$

where h is an arbitrary binary-valued function of $n - (m + 1)$ independent but arbitrary distributed binary random variables, and $X_1,\cdots, X_{m+1}$ are balanced i.i.d. random variables. These functions are (trivially) mth order correlation immune, as can be seen from the corresponding zero capacity discrete memoryless channel (DMC) in Fig. 2.

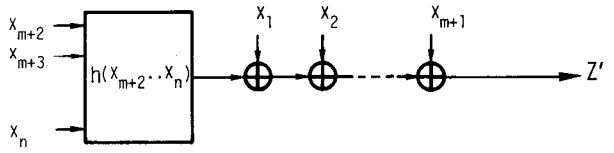

Fig. 2. Corresponding DMC for the set of functions Z'.

The additions of the random variables $X_1,\cdots, X_{m+1}$ have the same effect as a series of $m + 1$ binary symmetric channels, each having capacity zero. The output Z' provides no information about the input vector X_k, consisting of $k \le m$ arbitrary components X_i, $i \in \{1, 2,\cdots, n\}$ because still $m + 1 - k \ge 1$ binary symmetric zero capacity channels remain. The special case where $h(\cdot) = \text{const.} = 0$ or $h(\cdot) = \text{const.} = 1$ shows that functions of the form $f(X) = X_1 + X_2 + \cdots + X_n + c$, as given in Section III are indeed $(n - 1)$st order correlation-immune.

VII. Conclusions

A class of pn-generators consisting of n subgenerators and a memoryless combining function f has been investigated. It has been pointed out that a weakness of these generators may be the statistical dependence between a single subgenerator sequence (or between pairs, triples, etc., of sequences) and the keystream. A definition of correlation immunity has been given, together with a necessary condition for a combining function f to be mth-order correlation immune. A recursive construction method to obtain such mth-order correlation-immune functions of n variables for any $1 \le m < n$ has been described, and the trade-off between the length of the shortest linear equivalent of the nonlinear generator and its correlation immunity has been discussed. Finally, we have proposed a generalization to nonbalanced sequences.

Acknowledgment

The author is much indebted to Prof. Dr. J. L. Massey for his kind help and valuable suggestions concerning the concept of this work. Also thanks go to Prof. Dr. P. Leuthold for his support and interest in this work.

References

[1] T. Siegenthaler, "Correlation attacks on certain stream ciphers with nonlinear generators," presented at IEEE Int. Symp. Inform. Theory, Saint Jovite, Canada, Sept. 26–29, 1983.

[2] ——, "Decrypting a class of stream ciphers using ciphertext only," *IEEE Trans. Comput.*, vol. C-33, no. 10, Oct. 1984.

[3] J. O. Bruer, "On nonlinear combinations of linear shift register sequences," Linkoeping Univ. Sweden, Internal report March 83, presented at IEEE Int. Symp. Inform. Theory, les Arcs, France, June 21–25, 1982.

[4] P. R. Geffe, "How to protect data with ciphers that are really hard to break," *Electronics*, pp. 99–101, Jan. 4, 1973.
[5] E. L. Key, "An analysis of the structure and complexity of nonlinear binary sequence generators," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 6, p. 736, Nov. 1976.
[6] E. J. Groth, "Generation of binary sequences with controllable complexity," *IEEE Trans. Inform. Theory*, vol. IT-13, no. 3, pp. 288–296, May 1971.

Correction to "DES-Like Functions Can Generate the Alternating Group"

SHIMON EVEN AND ODED GOLDREICH

Manuscript received March 9, 1984.
S. Even is with the Computer Science Department, Technion—Israel Institute of Technology, Haifa 32000, Israel.
O. Goldreich was with the Laboratory for Computer Science, Massachusetts Institute of Technology, Room NE 43836, Cambridge, MA 02139. He is now with the Computer Science Department, Technion–Israel Institute of Technology, Haifa 32000, Israel.

Remark:

The Epilogue of the above paper[1] consisted of information concerning further research carried out by the authors at the time the manuscript was revised. This information was regrettably not accurate. It was only proved that

a) for any **even** $n \geq 8$, $2 \leq k \leq n-2$, k-functions on $V_{q,n}$ generate $A_{V_{q,n}}$;
b) for every $m, n > 1$, such that $m \cdot n$ is even and $m \cdot n \geq 8$, any generalized block processor of m q-ary n-vectors generates $A_{V_{q,m\cdot n}}$ when q is even or $(q-1)\cdot n \equiv 0 \pmod 4$ otherwise it generates $S_{V_{q,m\cdot n}}$.

For further details consult [1].

References

[1] O. Goldreich, "On the generating power of q-ary block ciphers," TR 264, Computer Science Department, Technion, Haifa, Israel, February 1983.

[1] S. Even and O. Goldreich, *IEEE Trans. Inform. Theory*, vol. IT-29, no. 6, pp. 863–865, Nov. 1983.

Book Reviews

Digital Communications, John G. Proakis (New York: McGraw Hill, 1983, xvi + 608 pp.).

J. E. MAZO, MEMBER, IEEE

This book is not a textbook on information theory, nor is it one on data communication networks. Rather, this is a major textbook on point-to-point data transmission. Although intended as an introductory course for communication engineers, the book includes many advanced topics popular in the last several years. In particular, it treates digital encoding of speech, least-squares algorithms for equalization and their lattice realizations, and coding for fading channels and for spread-spectrum communications.

Chapters 1, 3, and 4 comprise the classical and introductory parts. They contain a discussion of probability, power spectra of random signals, complex representation of bandpass signals (used well throughout the book) and, finally, geometric representation of signals, orthogonal signals, and coherent and noncoherent demodulation. The latter leads to a discussion of the probability distribution of quadratic forms of Gaussian variables. The depth of treatment in these chapters is sometimes uneven. For example, phase-locked loops are introduced, block diagrams are given, but there is no discussion of why they work. Yet an appendix is provided that treats the analytic calculation of FM spectra in considerable detail. This unevenness pervades the entire book.

Chapter 2 is concerned with digitizing analog sources. Various pulse-code modulation and delta modulation schemes are described but not analyzed. The treatment of linear predictive coding is confusing, but, on the other hand, a very nice appendix on the Levinson–Durbin algorithm is offered.

Chapter 5 is devoted to binary coding theory, including block codes, convolutional codes, Viterbi decoding algorithm, and the distinctions between hard and soft decisions. This eighty page chapter is the most readable introduction to coding theory that I have encountered. Pedagogically, it is the highlight of the book.

Intersymbol interference is the dominant theme of Chapter 6. Here we meet, too briefly, Nyquist's criterion for no intersymbol interference, and we quickly move to partial response signaling. Then we come to linear equalization. This is treated from the optimum linear receive viewpoint with (surprisingly) but a quick mention of the adaptive aspects. Decision feedback and Viterbi decoding of intersymbol interference are given their due. Finally, we come to thirty pages of least-square algorithms and lattice structures. I find again that once I read past the classical Kalman filter, I am adrift in a sea of formalism. This has happened to me so often that it may be my failing.

The last two chapters, 7 and 8, deal with fading channels and spread spectrum communications, respectively. An outstanding feature of both chapters is the integration of the coding material learned earlier. The spread-spectrum discussion is mostly concerned with jamming rather than code-division multiple-access.

The book does have some serious errors, but, fortunately, their effects are localized. The first error I saw occurred on page 174. There, quadratic noise terms are discarded on the grounds that the noise is small. However, when the noise causes an error it is not small, and such terms must be retained. The next error occurred in the equalizer convergence discussion, which begins on page 373. This discussion is based on a deterministic gradient search. This is a useful and necessary heuristic, but it leads to a completely wrong prediction of what the step-size should be for the algorithm to be stable. Thus the right side of $(6 \cdot 4 \cdot 74)$ should be reduced by a factor of N, where N is the number of equalizer taps. Lastly, on page 139 it is flatly stated that redundancy results in an increase of channel bandwidth. Not true, since an increase in the number of channel levels will do just as well. In fact, the areas of trellis codes and lattice block codes for the Gaussian channel exemplify recent research based on this type of redundancy (and are not discussed in this volume).

0018-9448/84/0900-0780$01.00