# Encryption and Error-Correction Coding Using $D$ Sequences

SUBHASH C. KAK, SENIOR MEMBER, IEEE

*Abstract* —**This paper presents several new properties of $D$ sequences that have applications to encryption and error coding. It also considers the problem of joint encryption and error-correction coding and proposes a solution using $D$ sequences. The encryption operation considered is equivalent to exponentiation, which forms the basis of several public-key schemes. An application of $D$ sequences to generating events with specified probabilities is also presented.**

*Index Terms* —**Cryptography, data security, $D$ sequences, error coding, public-key systems, random sequences.**

## I. Introduction

THIS paper is a study of several applications of $D$ sequences to encryption and error coding. $D$ sequences are obtained in expansions of fractions or irrational numbers and thus are "decimal" sequences to arbitrary bases. A standard account of elementary properties of $D$ sequences may be found in the text by Hardy and Wright [1]. Some properties of $D$ sequences that make them potentially useful for coding and multiple access have been described recently [2], [3]. A peculiar structural redundancy was pointed out in [4]. Blum *et al.* [5] have shown that $D$ sequences as pseudorandom sequences are cryptographically insecure.

Two important problems that have been considered in this paper are those of computing discrete logarithms and joint encryption and error coding. The discrete logarithm problem has been discussed recently by Hellman *et al.*, Adleman, and Coppersmith, [6]–[9]. Since exponentiation is at the basis of several modern cryptographic schemes, an efficient solution to the discrete logarithm problem is of great significance. In our paper we present a new approach to this problem which is based on the use of the autocorrelation function method. While our approach does not yield a computationally attractive solution, it opens up a new line of inquiry which may prove fruitful.

The transmission of encrypted blocks of data over a noisy channel requires an additional step of error-correction coding. We describe a method where the encrypted block digits generate a sequence, and therefore, sending more digits than the minimum necessary for uniquely defining the cipher block provides a corresponding degree of redundancy that can be used for error correction. The sequence digits are generated recursively, and therefore, the number of extra digits needed for a specific noise situation can be adjusted easily without having to change the error-correction codes necessary otherwise.

We call our method joint encryption and error-correction coding because both these operations are in the group of digits modulo an appropriate number. This is in contrast to separate error-correction coding where operations are usually in $GF(p^n)$. It should be noted that our use of $D$ sequences for encryption is not in the cryptographically insecure style of Blum *et al.* [5].

We also present several new results on $D$ sequences. These include results on frequency characteristics of the subsequences as well as on Hamming distance and autocorrelation function characteristics. It has been shown how the autocorrelation function for a binary $D$ sequence can be computed efficiently. A decoding procedure for error-correcting codes using $D$ sequences is also described.

Section II of the paper reviews some structural properties of $D$ sequences necessary as background and also presents new results especially on characteristics of subsequences. Section III describes an elementary relationship between a $D$ sequence and the finite exponential. Section IV presents new results on Hamming distance and autocorrelation bounds for $D$ sequences. It has also been shown how the autocorrelation method can be used for computing discrete logarithms. Section V shows how, starting with a number chosen randomly out of a residue set modulo a prime, events of arbitrary probability can be generated. Sections VI and VII address the problem of error coding and encryption.

## II. Structural Properties of $D$ Sequences

We begin with the observation that the digits in the decimal expansion of an irrational number satisfy most criteria of randomness. This suggests that the randomness properties of the decimal expansions of rational numbers might also be good. Let us now take the rational number $1/q$ and express it as a $D$ sequence in base $r$. It is known that this sequence will repeat itself with a period $v$ where $v$ is the order of $r$ mod $q$. If $q$ is a prime, and $r$ is a primitive root of $q$, then the decimal sequence is termed a maximum-length $D$ sequence (MLDS) in base $r$. An MLDS will often be represented merely by the string of its first $(q - 1)$ digits without showing the decimal or as $\{1/q\}$ or $\{1/q\}_r$. We will now enumerate some basic properties of MLDS's. The proofs of Properties 1–3 may be found in [2]. Note that $q$ is always a prime.

Manuscript received January 23, 1983; revised October 12, 1984. This paper was presented in part at the IEEE Symposium on Security and Privacy, Oakland, CA, April 25–27, 1983.

The author is with the Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA 70803.

0018-9340/85/0900-0803$01.00 © 1985 IEEE

*Property 1:* An MLDS $\{1/q\}$, when multiplied by $p$, $p < q$, is a cyclic permutation of itself.

*Example:* Consider $\{1/7\}$ in base 10. We see that 10 is a primitive root of 7 because $10^6 \equiv 1 (\mathrm{mod}\ 7)$ and $10^2 \not\equiv 1 (\mathrm{mod}\ 7)$, $10^3 \not\equiv 1 (\mathrm{mod}\ 7)$. Therefore, this $D$ sequence is of maximum length. The $D$ sequence is 1 4 2 8 5 7, which corresponds to the remainder sequence 3 2 6 4 5 1 where these remainders refer to the values obtained in the long division of 1 by 7. The remainder sequence has considerable structure. Thus, 10, $10^2$, $10^3$, $10^4$, $10^5$, $10^6$, all computed modulo 7, yield the successive digits of the sequence. If $x = \{3/7\}$ the remainder sequence starts with $30 \equiv 2 (\mathrm{mod}\ 7)$ and in fact is now 2 6 4 5 1 3, and therefore the decimal sequence for 3/7 is 4 2 8 5 7 1.

*Property 2:* For an MLDS $\{1/q\} = a_1 a_2 \cdots a_k$, $k = q - 1$, in the base $r$,

$$a_i + a_{i+k/2} = r - 1 .$$

This implies that maximum-length binary $D$ sequences $(r = 2)$ will be skew-symmetric about their midpoint. Note that all maximum-length sequences are of even length, and therefore the latter $(q - 1)/2$ digits for a binary sequence will be the complements of the first $(q - 1)/2$ digits.

The above property holds even for nonmaximum-length $D$ sequences, so long as the period is even.

*Property 3:* If the period $k$ of the $D$ sequence of $1/q$ is even in the base $r$,

$$a_i + a_{i+k/2} = r - 1 .$$

If the first $k/2$ digits of a decimal sequence of an even period are represented by $A$ and the remaining $k/2$ digits are represented by $B$, one can prove that $B$ divided by $A$ yields a quotient of $q - 1$ and a remainder of $q - 2$. To see this, divide $r^{k/2}$ by $q$, which yields the remainder of $q - 1$ and quotient of $A$. Therefore, dividing $r^{k/2}$ by $A$ will yield the remainder of $q - 1$ and quotient of $q$. Now, since by Property 3 $B = r^{k/2} - A - 1$, the result follows.

Let the $i$th remainder in the division of 1 by $q$ be represented by $m_i$ where $m_0 = 1$,

$$m_i = rm_{i-1} - qa_i = r^i \bmod q . \tag{1}$$

We can now easily establish the following:

$$m_{i+j} = r^{j+1} m_{i-1} - ql_i(j + 1) \tag{2}$$

where $l_i(j + 1) = r^j a_i + r^{j-1} a_{i+1} + \cdots + r a_{i+j-1} + a_{i+j}$. The sequence $l_i(j + 1)$ is, therefore, a $(j + 1)$ digit long subsequence of $\{1/q\}$ starting at its $i$th position.

*Property 4:* For a $D$ sequence $\{1/q\}_r$, if $r^m > q$, then all $l_i(m)$ are different. In other words, for such a sequence, all subsequences of length $m$ are different.

This can be seen by considering

$$-ql_i(j - 1) = m_{i+j} - r^{j+1} m_{i-1} .$$

Now, if $l_i = l_k$, then

$$m_{i+j} - r^{j+1} m_{i-1} = m_{k+j} - r^{j+1} m_{k-1} \tag{3}$$

or

$$m_{i+j} \bmod q \equiv m_{k+j} \bmod q \bmod r^{j+1}. \tag{4}$$

However, if $r^{j+1} > q$, we know that all the remainders must be distinct. Hence, if $r^{j+1} > q$, $l_i(j + 1) \neq l_k(j + 1)$, $i \neq k$, or all subsequences of length $m = j + 1$, where $r^m > q$, are different.

As an example, consider $\{1/17\}_{10}$. The remainder sequence and the $D$ sequence are shown in Table I. We note that whenever $m_i \bmod 10 \equiv m_k \bmod 10$, the corresponding $D$ sequence digits are equal.

### A. Frequency Characteristics of the Subsequences

Consider $l_i(j)$, the subsequence of length $j$ at the $i$th place, for $j$ such that $r^j < q$. Our objective is to extend Property 4. If $j = [\log_r q]$, where $[x]$ is the integer less than and closest to $x$, congruence (3) implies that

$$m_{i+j} \equiv m_{k+j} \bmod r^j, \tag{5}$$

and since $r^j$ is just smaller than $q$, $l_i = l_k$ at most $[q/r^j]$ times. Therefore, the number $N(j)$ of identical subsequences of length $j = [\log_r q]$ is computed by

$$N(j) = [q/r^j] \quad \text{or} \quad [q/r^j] + 1 . \tag{6}$$

In general, for $j < [\log_r q]$, a similar argument implies the following.

*Property 5:* Each subsequence of length $j$ in an MLDS occurs $N(j)$ times where

$$N(j) = [q/r^j] + C \tag{7}$$

and $C$ is 0 or 1.

The frequency of any subsequence of length $j$ in a period would therefore be $f(j) = N(j)/(q - 1)$; or

$$f(j) = [q/r^j]/(q - 1) + C/(q - 1),$$

and as $q$ becomes large

$$f(j) \rightarrow 1/r^j, \tag{8}$$

which is the frequency one would expect in a random sequence. This shows that one can use an MLDS as a pseudorandom sequence. For the frequency characteristics of reciprocals of integral powers of primes, see the paper by Stoneham [10].

## III. D Sequences and the Finite Exponential

Since the $i$th remainder is $m_i = r^i \bmod q$, the relationship between the $D$ sequence $\{1/q\}_r$ and the finite exponential is evident. Given any subsequence $l_i(j)$ in the expansion of $1/q$, the corresponding remainder $m_i$ can be obtained using (2), which can be rewritten as

$$m_{i-1+j}/r^j = m_{i-1} - ql_i(j)/r^j. \tag{9}$$

Using the fact that $m_{i-1+j} < q$, we obtain the following property.

*Property 6:*

$$l_i(j)q/r^j < m_{i-1} < \{l_i(j) + 1\}q/r^j. \tag{10}$$

TABLE I

| index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| remainder sequence | 10 | 15 | 14 | 4 | 6 | 9 | 5 | 16 | 7 | 2 | 3 | 13 | 11 | 8 | 12 | 1 |
| D sequence | 0 | 5 | 8 | 8 | 2 | 3 | 5 | 2 | 9 | 4 | 1 | 1 | 7 | 6 | 4 | 7 |

This implies that given enough digits of a subsequence the corresponding remainder $m_i$ can be easily determined.

For the digits $a_i$ of a $D$ sequence $(a_1 a_2 \cdots)$, the Fermat result [1] holds in the following form:

$$\left\{\left[\frac{(a_i r^{j-1} + \cdots\cdots + a_{i+j-1})q}{r^j}\right] + 1\right\}^{q-1} \equiv 1 \bmod q.$$

This shows again how the remainders and the $D$ sequence digits can be considered on an equal basis.

## IV. Hamming Distance and Autocorrelation Bounds and Computing Discrete Logarithms

*Property 7:* For a binary MLDS $\{1/q\}_2$: $a_1 a_2 \cdots a_k$, $k = q - 1$,

$$a_i = (2^i \bmod q) \bmod 2. \tag{11}$$

*Proof:* For $2^i < q$, $a_i = 0$. The $i$ for which $2^i$ is larger than $q$ for the first time, when $2^i \bmod q$ is even, would naturally yield an $a_i$ which is 1. Thereafter, each time $2^i \bmod q$ is odd it implies that the quotient in the division of 1 by $q$ is 1, and when $2^i \bmod q$ is even the quotient is 0. The following results also hold.

*Property 8:* The minimum Hamming distance $d_i$ between the maximum-length binary sequence $\{1/q\}_2$ and its cyclic shifts equals the integer closest to $q/3$ or $[2q/3] - [q/3]$.

*Property 9:* The Hamming distance between $\{1/q\}_2$ and $\{u/q\}_2$ is given by:

a) odd numbers in $(1, q2^{-i})$ + even numbers in $(q2^{-i}, q2^{-i+1})$ + odd numbers in $(q^{-i+1}, q2^{-i+2}) + \cdots$,
when $u = 2^i < q$,

b) $[2q/u] - [q/u] + [4q/u] - [3q/u] + \cdots + [(u-1)q/u] - [(u-2)q/u]$,
when $u$ is an odd number,

c) odd numbers in $(1, q/u)$ + even numbers in $(q/u, 2q/u)$ + odd numbers in $(2q/u, 3q/u) + \cdots$,
otherwise.

*Proof:* We wish to determine the smallest value of the distance between $\{1/q\}$ and $\{u/q\}$, $u < q$, $u \neq 1$.

$$\{1/q\}: a_1 a_2 \cdots a_k$$

$$\{u/q\}: b_1 b_2 \cdots b_k.$$

Then

$$\{(u-1)/q\}: (b_1 - a_1)(b_2 - a_2) \cdots (b_k - a_k)$$

where the $(b_i - a_i)$'s are 0, $-1$, or $-1$. The Hamming distance between $\{u/q\}$ and $\{1/q\}$ would therefore equal the number of nonzero $(b_i - a_i)$. Now, by (11),

$$a_i = (2^i \bmod q) \bmod 2$$

$$b_i = (u2^i \bmod q) \bmod 2.$$

Since the autocorrelation function, as also the Hamming distance, should be symmetric for $j = 0$, $(q-1)$, therefore $d_j = d_{q-j-1}$. In terms of $u$, since $u = 2^j (\bmod q)$ (see Property 1), therefore one needs to consider Hamming distances only for $u = 2^j (j \leq (q-1)/2)$. If $u = 2$ and if $2^i \bmod q$ is even and less than $q/2$, $u2^i \bmod q$ is less than $q$, and therefore $a_i$ and $b_i$ are both even and $(b_i - a_i)$ is 0. If $2^i \bmod q$ is odd and less than $q/2$, $u2^i \bmod q$ is less than $q$ and even, and therefore $(b_i - a_i)$ is $-1$. If $q/2 \leq 2^i \bmod q \leq q - 1$ and $a_i$ is even, then $b_i$ is odd and $(b_i - a_i)$ is 1. Hence, the Hamming distance for $u = 2$ (which clearly corresponds to $j = 1$) is

odd numbers in $(1, q/2)$ + even numbers in $(q/2, (q-1))$

$$= \begin{cases} \dfrac{q-1}{2}, & \text{if } (q-1) \text{ is divisible by } 4 \\[2ex] \dfrac{q+1}{2}, & \text{if } (q-1) \text{ is divisible by } 2. \end{cases}$$

If $u = 2^i < q$, then one can use an argument similar to that for $u = 2$. The Hamming distance would now be

$$\text{odd numbers in } (1, q2^{-i}) + \text{even numbers in } (q2^{-i}, q2^{-i+1}) + \text{odd numbers in } (q2^{-i+1}, q2^{-i+2}) + \cdots. \tag{12}$$

Clearly, this distance would be approximately equal to $(q-1)/2$.

Let us now take $u$ to be an odd number. If $2^i \bmod q < q/u$, $u2^i \bmod q$ is less than $q$, and therefore $a_i$ and $b_i$ are both even or odd and $(b_i - a_i)$ is 0. For $q/u \leq 2^i \bmod q < 2q/u$, $a_i$ and $b_i$ would now be even or odd or vice versa, leading to $(b_i - a_i)$, which is $+1$ or $-1$. Over $2q/u \leq 2^i \bmod q < 3q/u$, $(b_i - a_i)$ is again 0 and so on. The value of the Hamming distance is therefore equal to

$$[2q/u] - [q/u] + [4q/u] - [3q/u] + \cdots + [(u-1)q/u] - [(u-2)q/u], \tag{13}$$

which represents a part of the total interval of $(q-1)$. The smallest value of this expression is clearly defined for $u = 3$, which divides up the interval most favorably, giving

$$d_j = [2q/3] - [q/3]. \tag{14}$$

In words, this is the integer closest to $q/3$, or equivalently it is $(q-1)/3$ or $(q+1)/3$, whichever is an integer.

The next smallest value of the Hamming distance is likewise

$$[2q/5] - [q/5] + [4q/5] - [3q/5].$$

which will be attained for $u = 5$. This is approximately 1.2 times greater than (14).

When $u$ is 2 times an odd number the expression for the Hamming distance is likewise

$$\text{odd numbers in } (1, q/u) + \text{even numbers in } (q/u, 2q/u) + \text{odd numbers in } (2q/u, 3q/u) + \cdots, \qquad (15)$$

which can never be smaller than (14), which is attained for $u = 3 = 2^v (\text{mod } q)$ and the corresponding $u' = 2^{v+1} (\text{mod } q)$.

*Corollary:* The autocorrelation function $C(j)$ for a binary MLDS, in the symmetric $(1, -1)$ form, is

$$C(j) \leq 1/3, \qquad j \neq 0, j < q.$$

*Remarks:* The above result follows on substituting the Hamming distance lower bound. However, since $d_j$ itself varies considerably from its minimum value to a maximum of $q - 1$ (for $j = (q - 1)/2$), therefore the actual value of $C(j)$ would be distributed correspondingly.

The autocorrelation function of $\{1/q\}$, in the symmetric form, is 1 for $j = 0$ and $-1$ for $j = (q - 1)/2$. For $j$ = even, it is generally small and close to 0. The autocorrelation function is symmetric about both $j = 0$ and $j = (q - 1)/2$ and is spiky elsewhere. The magnitude of these spikes is between $(-1/\sqrt{q}, 1/\sqrt{q})$ for most values of the argument, and therefore the sequences appear more random as they become longer. The Fourier transform of the $D$ sequence or its autocorrelation function is a crude approximation of the constant function where alternate points are 0. Therefore, the autocorrelation properties of $D$ sequences are not as good as those of maximum-length shift register sequences, but their performance improves as their periods become longer. Fig. 1 is a plot of the autocorrelation function of the $D$ sequence $\{1/379\}_2$ in the $(-1, -1)$ form, which is typical for all binary MLDS. Figs. 2 and 3 give the autocorrelation function and the Fourier transform of two $D$ sequences in base 10. The autocorrelation function plots of Figs. 1 and 2 have similarities even though the bases are different. The Fourier transform plot of Fig. 3 shows up the frequency structure of the $D$ sequence in base 10.

The results of (12), (13), and (15) provide, in theory, a method of taking discrete logarithms mod $q$, in base 2. For the given $r = 2^i$ mod $q$, one can compute the Hamming distance between $\{1/q\}$ and $\{r/q\}$ using the above-mentioned equations. If a plot of the Hamming distances or, equivalently, the autocorrelation function of $\{1/q\}$ were available, one could read off the index $i$. In practice, this method will be infeasible when $q$ is large. This suggests that more research needs to be done to understand the properties of the autocorrelation function to determine if this approach could prove worthwhile.

It appears that an approach that is partly statistical may have some usefulness, even if basic new results on the autocorrelation function are not obtained. For example, if the position of the local peaks of the autocorrelation function could be determined, *a priori*, even if these values were determined only probabilistically, this information could be exploited to determine some characteristics of the index.

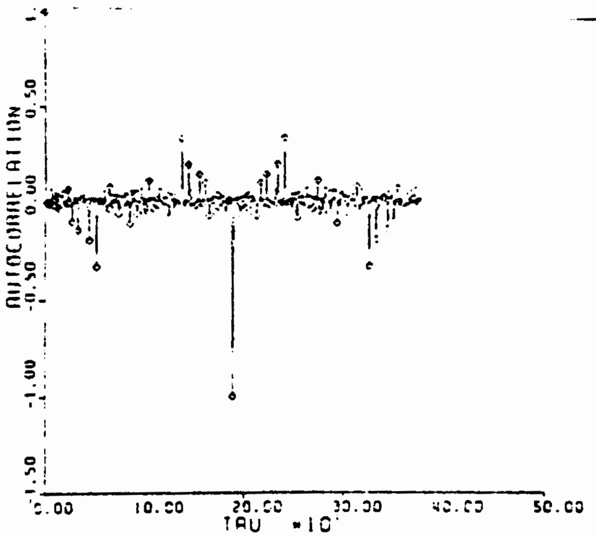

Fig. 1. Autocorrelation function of the sequence {1/379} in base 2.

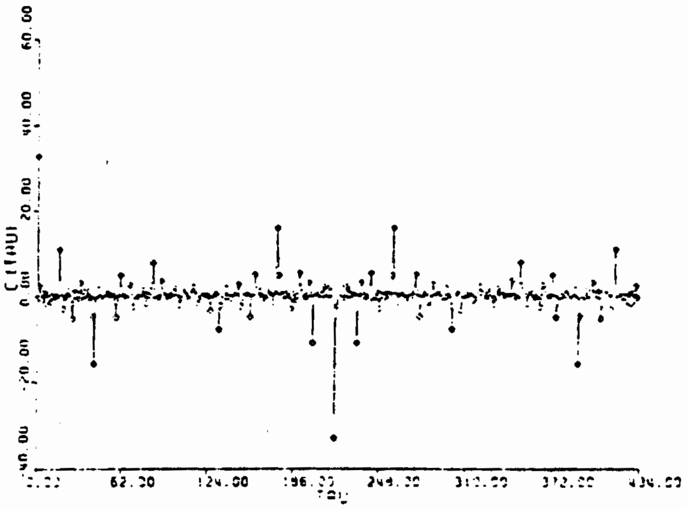

Fig. 2. Autocorrelation function of the sequence {1/433} in the symmetric form in base 10.

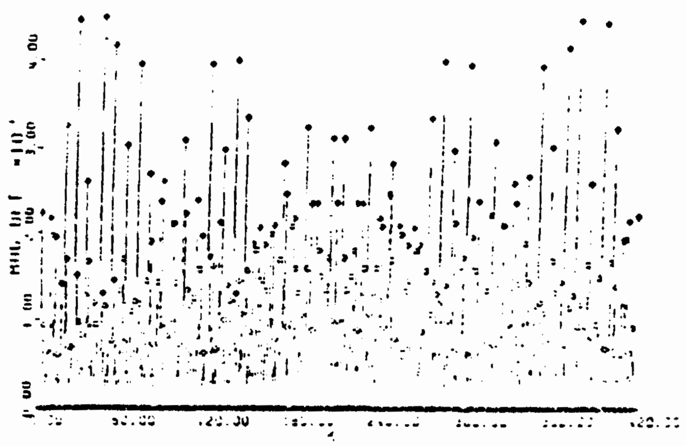

Fig. 3. Fourier transform of the sequence {1/419} in base 10.

## V. Generating Events with Different Probabilities

Our proof of Property 9 allows us to make a useful assertion about the integers less than a prime. Since the binary MLDS can be taken to be the sequence of even/odd parity of the remainders (1 through $q - 1$ in the order given by $2^i$ mod $q$), the Hamming distance between a residue set and that obtained by multiplying each element by $u$ is precisely the number of positions where the parity of the corresponding elements from the two sets is different. The expressions (12), (13), and (15) give the number of integers that when multiplied by $u$ give a result with a different parity, which we define to be the event $x_u$.

As an example, the number of times $t < q$, when multiplied by 3, yields a $3t$ mod $q$ with parity different from that of $t$ is precisely $[2q/3] - [q/3]$. This means that if a number $t$ is randomly chosen the probability of finding the parity of $3t$ different from it is

$$\text{Prob}(x_3) = 1/(q - 1)\{[2q/3] - [q/3]\} \cong 1/3 .$$

Some other probabilities of interest are

$$\text{Prob}(x_{q-1}) = 1$$

$$\text{Prob}(x_2) \cong 0.5 .$$

Similarly, from (13), for odd $u$,

$$\text{Prob}(x_u) \cong (u - 1)/2u .$$

Using appropriate $x_i$'s, more than once if necessary, events of any arbitrary probability can be designed.

It is necessary to ensure that the number $t$ is selected randomly for the above probability values to be valid. This is because given the number $t$ one can calculate what the parity of $ut$ will be. To illustrate this, if $u = 3$, the parity of 1 through $[q/3]$ and their negatives does not change, and those of all others do.

To select $t$ randomly, one needs to use a cryptographically secure random number generator whose output has been certified. In the casino setting a player would add his chosen number to the number produced for that play by the certified random number generator.

## VI. Error-Correction Coding

The fact that there exists a minimum Hamming distance between an MLDS and its cyclic shifts, generated by multiplication by integers (Property 2), suggests that the set could be used as error-correcting codewords. One can also use nonmaximum-length sequences for the same purpose.

*Definition:* A $D$ code for a message expressed as integer $u$ is defined as

$$x = \{u, q\}$$

where $u \leq q - 1$.

When $\{1/q\}$ is a maximum-length binary sequence, we obtain a binary maximum-length $D$ (MLD) code, which is a cyclic code. The codewords in a binary code have equal numbers of 0's and 1's. It is quite clear that for a binary code the lower bound on errors detected is $(d - 1)$ and the lower bound on errors corrected is $[(d - 1)/2]$, $d = [2q/3] - [q/3]$. Since $d$ is often larger than its minimum value, the actual error-detection and -correction capability will be higher.

*Example:* Let $q = 11$. The information word to the codeword mapping in the MLD code is

$$0001 \rightarrow 0001011101$$

$$0010 \rightarrow 0010111010$$

$$0011 \rightarrow 0100010111$$

etc.

One observes that the codewords are linear in terms of ordinary addition, but nonlinear in terms of modulo 2 addition.

An MLD code can also be viewed as a product code $u(2^{q-1} - 1)/q$ because $\{1/q\} = (2^{q-1} - 1)/q$.

The code corresponding to a nonmaximum-length $D$ sequence will be called an NMLD code, which for $r = 2$ is a product code $u(2^k - 1)/q$ where $k$ is the order of 2(mod $q$), $u < q$.

*Example:* Let $q = 23$, $k = 11$. The NMLD code is the product code $89u$. Some of the codewords for this case are

$$00001 \Rightarrow 00001011001$$

$$00111 \Rightarrow 01001101111$$

etc.

All the remaining codewords are the cyclic shifts of the above two codewords.

One may also use a shortened $D$ code. By Property 4, only $m$ digits are required to fix $u$ (where $r^m > q$) in a radix $r$ representation. Therefore, if one does not need the full error detection/correction provided by a complete $D$ code, the extra number of codeword digits can be deleted. The remainder, and thereby the message, can then be constructed by the use of Property 6. Several properties of arithmetic codes are reviewed by Clark and Liang [11].

### A. A Decoding Procedure

We sketch a decoding algorithm for an MLD code; a similar algorithm will apply for NMLD codes. Let the code length be $m + p$ where $[r^m/q] = 0$. We construct a table for all possible $l_i(j)$ for $j = m, m - 1, \cdots, 1$. Using the relationship

$$[l_i(j)q/r^j] < m_{i-1} < [\{l_i(j) + 1\}q/r^j]$$

we construct a sequence of possible $m_i$'s for each $j$, and check which ones are consistent with the relationship $m_i = rm_{i-1}$ mod $q$. For a given $j$, we determine the index values $i$ where two different possible remainder sequences meet. The code digit is changed for this $i$, and the remainder sequence is again checked for consistency. If it checks out, the $i$th bit was in error. If it does not, the procedure is repeated for $j - 1$. We illustrate our algorithm by means of an example of a single error.

*Example:* Consider $\{1/13\}_2$. Let the message sequence be

TABLE II

| $i$ = | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| $l_i(4)$ | | 13 | 11 | 7 | 14 | | | |
| $m_i(4)$ | 11 | 9 | 6 | 12 | | | | |
| $l_i(3)$ | | 6 | 5 | 3 | 7 | 6 | | |
| $m_i(3)$ | 10<br>11 | 9<br>10 | 5<br>6 | 12 | 10<br>11 | | | |
| $l_i(2)$ | | 3 | 2 | 1 | 3 | 3 | 2 | |
| $m_i(2)$ | 10<br>11<br>12 | 7<br>8<br>9 | 4<br>5<br>6 | 10<br>11<br>12 | 10<br>11<br>12 | 7<br>8<br>9 | | |
| $l_i(1)$ | | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| $m_i(1)$ | 7 to 12 | 7 to 12 | 1 to 6 | 7 to 12 | 7 to 12 | 7 to 12 | 7 to 12 | |

1000, which is transformed by the $D$ code ($p = 3$) into the codeword 1 0 0 1 1 1 0. Let the received codeword be 1 1 0 1 1 1 0, which is in error at the second location. We take the received word and construct a table (Table II) of $l_i(j)$'s and $m_i$'s for various values of $i$'s and $j$'s.

We start with an examination of $l_i(4)$ and $m_i(4)$. 11, 9 and 6, 12 are possible remainder sequences, but since they do not meet at an $i$, we take up $l_i(3)$ and $m_i(3)$. The sequences 11, 9, 5 and 6, 12, 11 meet at $i = 2$. The second bit is changed from 0 to 1, and $m_i(3)$ and $m_i(4)$ are checked to show that they are consistent. Hence, the second bit was in error.

Had examination of $m_i(3)$ not resolved the matter, we would have checked $m_i(2)$. Again, sequences 11, 9, 5 and 6, 12, 11, 9 meet at $i = 2$, which indicates that the bit at the second place might be in error.

The computational effort required for low bit error rate situations (which is a realistic assumption for cryptographic applications) is not excessive, and therefore this procedure can be used in practice.

## VII. Joint Encryption and Error Coding

The transmission of encrypted blocks of data over a noisy channel requires an additional step of error-correction coding. We describe a method where the cipher block generates a continuing $D$ sequence, and therefore, sending more digits than the minimum necessary for uniquely defining the cipher block provides a corresponding degree of redundancy that can be used for error correction. The sequence digits are generated recursively, which makes it easy to adjust the number of extra digits needed for a specific noise situation. In contrast to sequential encryption and error correction this does not require a change of the coder itself.

We consider the Diffie–Hellman key distribution scheme [12] to form the basis of our cryptographic system. In this system one assumes that all users, as also the cryptanalyst, have access to a large prime $q$ and one of its primitive roots $r$. When $A$ and $B$ wish to communicate, they first generate random numbers $k1$ and $k2$, respectively. $A$ transmits to $B$ the number $r^{k1}$ mod $q$, and $B$ transmits to $A$ the number $r^{k2}$ mod $q$. Both $A$ and $B$ can now generate the key $r^{k1k2}$ mod $q$ to use as the secret key to exchange messages.

In the Diffie–Hellman method it is important that the transmission of the $r^k$ mod $q$ numbers be error free. This requires error-correction coding. In order to eliminate this step of error correction we propose sending, instead of $r^k$ mod $q$, $j$ digits in base $q$, where $r^j > q$, in the following fashion.

Step 1: Generate $r^k$ mod $q$.

Step 2: Find the first $j$ digits of the expansion

$$\frac{r^k \bmod q}{q} \text{ in base } r.$$

Let the digits be represented by $a_{k1}a_{k2}\cdots a_{kj}$. These $j$ digits uniquely identify $r^k$ mod $q$.

Step 3: Transmit $a_{k1}a_{k2}\cdots a_{kj}$.

If the sequence of digits in the $D$ sequence expansion of $l/q$ in base $r$ is represented by $a_1a_2\cdots$, then $a_{k1} = a_{k+1}$, $a_{k2} = a_{k+2}$, etc. This means that the transformation $r^k$ mod $q$ is equivalent to sending $j$ digits of the expansion of $1/q$ in base $r$ starting at the $(k + 1)$ position. When $j$ is greater than $m$ where $m$ makes $r^m$ just greater than $q$, each extra digit provides redundancy against errors in transmission. Since the protection provided is equivalent to that in $D$ code, the performance can be easily evaluated.

The reason why we call our method joint encryption and error coding is because both the operations are performed mod $q$, which is not the case for most standard techniques of error-correction coding. Our method also shows a connection between "decimal" expansions and encryption.

The digits of the expansion of $(r^k \bmod q)/q$ in (16) can also be expressed in any other base $b$. Again, at least $j$ digits are required where $b^j > q$. The expansion does not admit the elegant interpretation of being the $D$ sequence of $1/q$ anymore. It can still be used for error correction so long as $j$ is larger than $m$ where $\lfloor b^m/q \rfloor = 0$.

For other exponentiation transformations like the RSA [13], (16) can again be used for error correction where the digits are expressed to some appropriate base. As an example, $M^e$ mod $n$, for a message $M$, would be substituted by $(M^e \bmod n/n)$ in base $b$.

This method can also be applied for error-correction coding when the finite exponential is used as a one-way transformation for user authentication. Further, it can be used to exchange messages secretly by employing the Shamir–Rivest–Adleman (SRA) protocol for mental poker [14]. In the SRA and RSA methods, however, use of $D$ sequences amounts merely to a coding technique, and does not admit the elegant interpretation possible for the modification of the Diffie–Hellman method via the $D$ sequence.

## VIII. Concluding Remarks

We have shown that $D$ sequences can have significant applications in error coding and data security. The Diffie–Hellman scheme of key distribution has already been implemented, and therefore its variant, the joint encryption and error coding scheme proposed in this paper, can have immediate applications. Some directions in which further research needs to be done are

1) a deeper study of the autocorrelation function of a $D$ sequence and its use for computing discrete logarithms;

2) more efficient decoding algorithms for $D$ codes;

3) implementation protocols for joint encryption and error coding:

4) study of the performance of $D$ codes when they are shortened: and

5) implementing $D$ codes for password protection and for the RSA algorithm.

## References

[1] G. H. Hardy and E. M. Wright. *An Introduction to the Theory of Numbers*. London. England: Oxford Univ. Press. 1954.

[2] S. C. Kak and A. Chatterjee. "On decimal sequences." *IEEE Trans. Inform. Theory*. vol. IT-27. Sept. 1981.

[3] S. C. Kak. "Decimal sequences and their applications in communications." in *Proc. Int. Conf. Commun.*. June 1981.

[4] ——. "A structural redundancy in $D$ sequences." *IEEE Trans. Comput.*. vol. C-32. pp. 1069–1070. Nov. 1983.

[5] L. Blum. M. Blum. and M. Shub. "Comparison of two pseudo-random number generators." in *Adv. Cryptology: Proc. Crypto 82*. New York: Plenum. 1983. pp. 61–78.

[6] S. C Pohlig and M. E. 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.

[7] L. Adleman. "A subexponential algorithm for the discrete logarithm problem with applications to cryptography." in *Proc. Twentieth IEEE Symp. Foundations Comput. Sci.*. Oct. 1979. pp. 55–60.

[8] M. E. Hellman and J. M. Reyneri. "Fast computation of discrete logarithms in $GF(q)$." in *Adv. Cryptology: Proc. Crypto 82*. New York: Plenum. 1983.

[9] D. Coppersmith. "Fast evaluation of logarithms in fields of characteristic two." *IEEE Trans. Inform. Theory*. vol. IT-30. pp. 587–594. July 1984.

[10] R. G. Stoneham. "The reciprocals of integral powers of primes and normal numbers." *Proc. Amer. Math. Soc.*. vol. 15. pp. 200–208. 1964.

[11] W. E. Clark and J. J. Liang. "Weak radix representation and cyclic codes over Euclidean domains." *Commun. Algebra*. vol. 4. pp. 999–1028. 1978.

[12] W. Diffie and M. E. Hellman. "Privacy and authentication: An introduction to cryptography." *Proc. IEEE*. vol. 67. pp. 397–427. Mar. 1979.

[13] R. L. Rivest. A. Shamir. and L. Adleman. "A method for obtaining digital signatures and public-key cryptosystems." *Commun. ACM*. vol. 21. pp. 120–126. Feb. 1978.

[14] A. Shamir. R. L. Rivest. and L. Adleman. "Mental poker." in *The Security of Data in Networks*. D. W. Davies. Ed. Silver Spring. MD: IEEE Computer Society Press. 1981.

**Subhash C. Kak** (SM'77) is a Professor of Electrical and Computer Engineering at Louisiana State University. Baton Rouge. LA. where he has been since 1979. His earlier appointments have been at the Tata Institute of Fundamental Research. Bell Laboratories. Imperial College (University of London). and the Indian Institute of Technology. Delhi. He has worked in signal processing. information theory. coding. cryptography. and theoretical physics. His current work includes the study of algorithm complexity and parallel processing.