Constructions of Binary Constant-Weight Cyclic Codes and Cyclically Permutable Codes

Nguyen Q. A, László Györfi, and James L. Massey, *Fellow, IEEE*

Abstract—**A general theorem is proved showing how to obtain a constant-weight binary cyclic code from a p-ary linear cyclic code, where p is a prime, by using a representation of $GF(p)$ as cyclic shifts of a binary p-tuple. Based on this theorem, constructions are given for four classes of binary constant-weight codes. The first two classes are shown to achieve the Johnson upper bound on minimum distance asymptotically for long block lengths. The other two classes are shown similarly to asymptotically meet the low-rate Plotkin upper bound on minimum distance. A cyclically permutable code is a binary code whose codewords are cyclically distinct and have full cyclic order. A simple method is given for selecting virtually the maximum number of cyclically distinct codewords with full cyclic order from Reed–Solomon codes and from Berlekamp–Justesen maximum-distance-separable codes. Two correspondingly optimum classes of constant-weight cyclically permutable codes are constructed by appropriate selection of codewords from the first two classes of binary constant-weight codes. It is shown that cyclically permutable codes provide a natural solution to the problem of constructing protocol-sequence sets for the M-active-out-of-T users collision channel without feedback.**

Index Terms—**Collision channel, constant-weight codes, cyclic codes, cyclically permutable codes, maximum-distance-separable codes, protocol sequences, Reed–Solomon codes.**

I. Introduction

THIS paper has a threefold purpose, the first of which is to present some new constructions of binary constant-weight cyclic codes. By a *cyclic code*, we will always mean a block code such that the cyclic shift of every codeword is again a codeword. This terminology is not new (cf. [1, p. 176]), but it is more common to reserve the term "cyclic code" for a code that is *both* linear *and* closed under cyclic shifting, cf. [2, p. 93], [3, p. 188], and [4, p. 206]. To avoid ambiguity, we will always say "linear cyclic code" when we mean a "cyclic code" in this more restrictive sense. A *constant-weight* code is a block code whose codewords all have the same Hamming weight. Thus, every binary constant-weight cyclic code, except the trivial code containing only the all-zero codeword, is a nonlinear code.

The second purpose of this paper is to present some constructions of constant-weight cyclically permutable codes based on our constructions of binary constant-weight cyclic codes. Gilbert [1] has defined a *cyclically permutable code* to be a binary block code of block length N such that each codeword has cyclic order N (i.e., has N distinct cyclic shifts) and such that the codewords are cyclically distinct (i.e., no codeword can be obtained by the cyclic shifting, one or more times, of another codeword).

The third and final purpose of this paper is to show an interesting application for constant-weight cyclically-permutable codes, viz., as the set of protocol sequences for the T potential users of the collision channel without feedback when it is known that at most M users are actively using the channel at any given time.

In Section II, we present a correspondence between $m \times n$ arrays and N-tuples with $N = mn$ that play a key role in the constructions of binary constant-weight cyclic codes that follow. Four classes of binary constant-weight cyclic codes are constructed in Section III and shown to be asymptotically optimum in the sense of achieving either the Johnson upper bound or the low-rate Plotkin upper bound on minimum distance for large block lengths. Section IV presents a simple method to select a large subset of codewords with full cyclic order from a Reed–Solomon code or from a Berlekamp–Justesen maximum-distance-separable code. When combined with the results of Section III, this yields constructions of constant-weight cyclically permutable codes. Section V describes the application of these latter codes as protocol-sequence sets for the random accessing of the M-active-out-of-T-users collision channel without feedback. We close in Section VI with some remarks.

II. Two-Dimensional Arrays and N-Tuples

We first develop a correspondence between two-dimensional arrays and N-tuples and prove some properties of this correspondence that will be exploited in the next section.

Our interest is in $m \times n$ arrays

$$A = \begin{bmatrix} a(0,0) & \cdots & a(0,n-1) \\ \vdots & \vdots & \vdots \\ a(m-1,0) & \cdots & a(m-1,n-1) \end{bmatrix}$$

with entries in an arbitrary alphabet. When the positive integers m and n are relatively prime, i.e., when $\gcd(m, n) = 1$, then the Chinese remainder theorem [2, pp. 285–286] specifies a one-to-one correspondence between such arrays A and mn-tuples $\boldsymbol{b} = [b_0, b_1, \cdots, b_{mn-1}]$ over the same alphabet in the manner that

$$b_i = a(i \bmod m,\ i \bmod n), \tag{1}$$

Manuscript received July 3, 1985; revised August 8, 1989. This work was presented in part at the IEEE International Symposium on Information Theory, Brighton, United Kingdom, June 23–28, 1985.

N. Q. A and L. Györfi are with the Hungarian Academy of Sciences, Technical University of Budapest, Stoczek u. 2, H-1111 Budapest, Hungary.

J. L. Massey is with the Signal and Information Processing Laboratory, Swiss Federal Institute of Technology, CH-8092 Zürich, Switzerland.

IEEE Log Number 9105766.

where here and hereafter "$i \bmod m$" denotes the remainder when i is divided by m. For instance, the 2×3 array

$$A = \begin{bmatrix} a & e & c \\ d & b & f \end{bmatrix}$$

corresponds in this way to the 6-tuple

$$\boldsymbol{b} = [a, b, c, d, e, f].$$

We will always mean this "Chinese remainder theorem correspondence" (CRT correspondence) whenever we speak of an $m \times n$ array and its corresponding mn-tuple.

Let $\boldsymbol{R}$ denote the operator that shifts the columns of an $m \times n$ array cyclically one position *rightwards*, and let $\boldsymbol{D}$ denote the operator that shifts the rows cyclically one position *downwards*. For the above example, we have

$$\boldsymbol{R}(A) = \begin{bmatrix} c & a & e \\ f & d & b \end{bmatrix},$$

$$\boldsymbol{D}(A) = \begin{bmatrix} d & b & f \\ a & e & c \end{bmatrix},$$

and

$$\boldsymbol{DR}(A) = \boldsymbol{RD}(A) = \begin{bmatrix} f & d & b \\ c & a & e \end{bmatrix}.$$

Notice that $\boldsymbol{DR}(A) = \boldsymbol{RD}(A)$ corresponds to the 6-tuple

$$[f, a, b, c, d, e] = \boldsymbol{S}(\boldsymbol{b}),$$

where $\boldsymbol{S}$ is the (rightward) cyclic shift operator on N-tuples and $N = mn$. The following result, which is the key to the constructions of constant-weight cyclic codes that will be given later, generalizes an argument that has been used for linear cyclic product codes [3, p. 570].

Lemma 1: A set of $m \times n$ arrays over an arbitrary alphabet, where $\gcd(m, n) = 1$, is closed under both the rightward column shift operator $\boldsymbol{R}$ and the downward row shift operator $\boldsymbol{D}$, if and only if the corresponding set of mn-tuples is closed under the (rightward) cyclic shift operator $\boldsymbol{S}$.

Proof: Let the $m \times n$ array A correspond to the mn-tuple $\boldsymbol{b}$. In $\boldsymbol{S}^k(\boldsymbol{b})$, the entry b_i of $\boldsymbol{b}$ is replaced by $b_{i-k \bmod mn}$. In $\boldsymbol{R}(A)$, the entry $a(i \bmod m, i \bmod n)$ of A is replaced by $a(i \bmod m, i - 1 \bmod n)$. Because $\gcd(m, n) = 1$, there is an m' with $1 \le m' < n$ such that $mm' \bmod n = 1$. It follows from (1) that

$$\begin{aligned} b_{i-mm' \bmod mn} &= a(i - mm' \bmod m, i - mm' \bmod n) \\ &= a(i \bmod m, i - 1 \bmod n), \end{aligned}$$

and hence, that $\boldsymbol{S}^{mm'}(\boldsymbol{b})$ corresponds to $\boldsymbol{R}(A)$. Thus, the set of $m \times n$ arrays is surely closed under $\boldsymbol{R}$ when the set of mn-tuples is closed under $\boldsymbol{S}$. A completely analogous argument shows that $\boldsymbol{S}^{nn'}(\boldsymbol{b})$ corresponds to $\boldsymbol{D}(A)$ where n', $1 \le n' < m$, satisfies $nn' \bmod m = 1$. Hence, the set of $m \times n$ arrays is surely also closed under $\boldsymbol{D}$ when the set of mn-tuples is closed under $\boldsymbol{S}$.

Conversely, we note that $\boldsymbol{RD}(A) = \boldsymbol{DR}(A)$ is the $m \times n$ array corresponding to the mn-tuple $\boldsymbol{S}^{mm'+nn'}(\boldsymbol{b})$. But $mm' + nn' \bmod m = 1$ and $mm' + nn' \bmod n = 1$ so that the Chinese remainder theorem assures us that $mm' + nn' \bmod mn = 1$. Hence, $\boldsymbol{RD}(A) = \boldsymbol{DR}(A)$ corresponds to $\boldsymbol{S}(\boldsymbol{b})$, which guarantees that the set of mn-tuples is closed under $\boldsymbol{S}$ when the set of $m \times n$ arrays is closed under both $\boldsymbol{D}$ and $\boldsymbol{R}$. □

III. Some Constructions of Binary Constant-Weight Codes

A. Cyclic Representation of GF(p)

Our code constructions will be based on certain linear cyclic codes over the finite field of p elements, $\mathrm{GF}(p)$, where p is a prime, whose elements are $0, 1, 2, \cdots, p - 1$. In this subsection, we introduce the special representation of $\mathrm{GF}(p)$ that will be used to convert such p-ary codes to binary codes.

We recall that the *cyclic order* of an N-tuple $\boldsymbol{b}$ is the smallest positive integer i such that $\boldsymbol{S}^i(\boldsymbol{b}) = \boldsymbol{b}$. It follows that the cyclic order of an N-tuple must be a divisor of N. Because p is a prime, a binary p-tuple $\boldsymbol{v}$ has cyclic order either 1 or p. Hence, $\boldsymbol{v}$ has cyclic order p unless $\boldsymbol{v} = \mathbf{0}$ or $\boldsymbol{v} = \mathbf{1}$, where here and hereafter we write $\mathbf{0}$ and $\mathbf{1}$ to denote the all-zero and all-one p-tuple, respectively. For any $\boldsymbol{v}$ not $\mathbf{0}$ or $\mathbf{1}$, we define the *$\boldsymbol{v}$-representation of* $\mathrm{GF}(p)$ to be the representation in which the element i of $\mathrm{GF}(p)$ is represented by the p-tuple $\boldsymbol{S}^i(\boldsymbol{v})$, the ith rightward cyclic shift of $\boldsymbol{v}$.

Example 1: The $\boldsymbol{v}$-representation of GF(7) for the choice $\boldsymbol{v} = [0, 0, 1, 1, 1, 0, 1]$ is as follows:

$$\begin{array}{ll} 0 & [0,0,1,1,1,0,1] \\ 1 & [1,0,0,1,1,1,0] \\ 2 & [0,1,0,0,1,1,1] \\ 3 & [1,0,1,0,0,1,1]. \\ 4 & [1,1,0,1,0,0,1] \\ 5 & [1,1,1,0,1,0,0] \\ 6 & [0,1,1,1,0,1,0] \end{array}$$

Lemma 2: In the $\boldsymbol{v}$-representation of $\mathrm{GF}(p)$, where *p is a prime and $\boldsymbol{v}$* is a p-tuple not $\mathbf{0}$ or $\mathbf{1}$, the representation of $i + 1$ is the rightwards cyclic shift of the representation of i for all i in $\mathrm{GF}(p)$.

The truth of this lemma follows immediately from the definition of the $\boldsymbol{v}$-representation of $\mathrm{GF}(p)$ and the fact that $(p - 1) + 1 = 0$ in $\mathrm{GF}(p)$. That a representation with this cyclic property is possible for $\mathrm{GF}(p)$ follows from the fact that the additive group of $\mathrm{GF}(p)$ is a cyclic group—it is not possible for $\mathrm{GF}(p^s)$ when $s > 1$, since then the additive group is not cyclic.

We note that the p-tuples in the $\boldsymbol{v}$-representation of $\mathrm{GF}(p)$ form a binary constant-weight cyclic code. We will write $d(\boldsymbol{v})$ to denote the minimum (Hamming) distance of this code, and we will call the $\boldsymbol{v}$-representation *equidistant* if the Hamming distance between every pair of distinct codewords

in this code is equal to $d(v)$. The following assertion is obvious.

Lemma 3: For every prime p, the p-tuple $v = [1, 0, 0, \cdots, 0]$ yields an equidistant v-representation of GF(p) with $d(v) = 2$.

The v-representation of GF(p) with $v = [1, 0, 0, \cdots, 0]$ has been used before, cf. [5]–[7], but for purposes different from that in this paper.

Example 2: The 7-tuple $v = [0, 0, 1, 1, 1, 0, 1]$ used in Example 1 yields an equidistant v-representation of GF(7) with $d(v) = 4$.

The p-tuple v of Examples 1 and 2 is just an "m-sequence" and the equidistant property of its v-representation reflects the familiar "shift-and-add" property of m-sequences [8, p. 287]. There exist binary m-sequences of length $2^n - 1$ for every integer $n > 1$. Hence, there exists a v-representation of GF(p) for which v is an m-sequence, if and only if $p = 2^n - 1$ for some n, i.e,. if and only if p is a *Mersenne prime* [9, pp. 15–16]. Note that if $p = 2^n - 1$ is a Mersenne prime, then $(p - 1)/2 = 2^{n-1} - 1$ is necessarily odd. The first 10 Mersenne primes correspond to $n = 2, 3, 5, 7, 13, 17, 19, 31, 61$, and 89; it is a longstanding open question in number theory whether the set of Mersenne primes is finite or infinite.

Lemma 4: If p is a Mersenne prime and v is a binary m-sequence of length p, then the v-representation of GF(p) is equidistant with $d(v) = (p + 1)/2$.

A nonzero element i of GF(p), where p is an odd prime, is said to be a *quadratic residue* (*QR*) *modulo p* if i is the square of some element of GF(p) and to be a *quadratic nonresidue* (*QNR*) *modulo p* otherwise; the element 0 of GF(p) is neither a QR residue modulo p nor a QNR modulo p. For example, $1 = 1^2$, $2 = 3^2$, and $4 = 2^2$ are the QR's modulo 7, while 3, 5, and 6 are the QR's modulo 7. By a *Legendre sequence*, we mean a binary sequence $v = [v_0, v_1, \cdots, v_{p-1}]$ of prime length p, where $(p - 1)/2$ is odd, such that $v_i = 0$ if i is a QNR modulo p and $v_i = 1$ if i is a QNR modulo p; the digit v_0 can be either 0 or 1. For example,

$$v = [0, 0, 0, 1, 0, 1, 1] \quad \text{and} \quad v = [1, 0, 0, 1, 0, 1, 1]$$

are the two Legendre sequences of length $p = 7$.

Lemma 5: For every prime p such that $(p - 1)/2$ is odd, a Legendre sequence v of length p yields an equidistant v-representation of GF(p) with $d(v) = (p + 1)/2$.

Proof: It is well known that Legendre sequences, when converted to ± 1 sequences by mapping 0 and 1 to $+1$ and -1, respectively, have a periodic autocorrelation function that is -1 at all off-peak shifts, cf. Boehmer [10] who attributes this result to Turyn [11]. This is equivalent to saying that the Hamming distance between v and each of its $p - 1$ distinct cyclic shifts is $(d + 1)/2$, which proves the lemma. Alternatively, the lemma can be seen as a simple consequence of a theorem in MacWilliams and Sloane [3, Theorem 6, p. 46] together with the fact that, because $(p - 1)/2$ is odd, i is a QR modulo p, if and only if $-i$ is a QNR modulo p. □

B. The Main Theorem Used in the Constructions

Our constructions will make use of p-ary linear cyclic (n, k, d) codes where n, k, and d are the blocklength, dimension and minimum Hamming distance, respectively, of these codes with code digits in GF(p). We will use the same notation, $\mathbf{1}$, for the all-one n-tuple as we used earlier for the all-one p-tuple and will rely on the context to determine which is meant.

Theorem 1: Let p be a prime and let V be a p-ary linear cyclic (n, k, d) code such that $\gcd(n, p) = 1$ and such that $\mathbf{1}$ is a codeword. Let v be a binary p-tuple with Hamming weight $w(v)$ where $0 < w(v) < p$. Let each codeword $\boldsymbol{c} = [c_0, c_1, \cdots, c_{n-1}]$ in V determine a $p \times n$ array A in the manner that the ith column of A is the transpose of the p-tuple that is the v-representation of the ith component of $\boldsymbol{c}$, and let $\boldsymbol{b}$ be the binary N-tuple (where $N = np$) that corresponds to the array A by the Chinese remainder theorem correspondence of Section II. Then the set of p^k N-tuples $\boldsymbol{b}$ corresponding in this way to the p^k codewords $\boldsymbol{c}$ of V form a binary cyclic code whose codewords have constant weight $w = nw(v)$ and whose minimum distance, $d_{\min}$, satisfies

$$d_{\min} \geq d\,d(v) \tag{2}$$

with equality when the v-representation of GF(p) is equidistant.

Proof: We first show that the set of p^k N-tuples $\boldsymbol{b}$ is closed under cyclic shifting. Let $\boldsymbol{c}$ and A be the codeword in V and the corresponding $p \times n$ array, respectively. Because V is a (linear) cyclic code, the rightward cyclic shift of $\boldsymbol{c}$ is also in V and hence, the corresponding array, which is $\boldsymbol{R}(A)$, is another array in the set. Because $\mathbf{1}$ is a codeword in V and V is linear, $\boldsymbol{c} + \mathbf{1}$ is also in V. But, by Lemma 2, $\boldsymbol{c} + \mathbf{1}$ corresponds to the array $\boldsymbol{D}(A)$ so that $\boldsymbol{D}(A)$ is another array in the set. Thus, the set of p^k arrays A is closed under both the $\boldsymbol{R}$ and $\boldsymbol{D}$ operators. It now follows from Lemma 1 that the corresponding set of p^k binary N-tuples $\boldsymbol{b}$ is closed under cyclic shifting, i.e., it is a binary cyclic code.

Because all vectors in the v-representation of GF(p) have Hamming weight $w(v)$, it follows that every N-tuple $\boldsymbol{b}$ in the set has Hamming weight $nw(v)$ so that the set is a constant weight code. For distinct codewords $\boldsymbol{c}$ and $\boldsymbol{c}'$ in V, the corresponding arrays A and A' will differ in precisely those columns corresponding to positions where $\boldsymbol{c}$ and $\boldsymbol{c}'$ differ, i.e., in $d(\boldsymbol{c}, \boldsymbol{c}')$ columns were $d(\cdot, \cdot)$ denotes Hamming distance. Each of the corresponding columns of A and A' will differ in at least $d(v)$ positions with equality if the v-representation of GF(p) is equidistant. Thus, the binary codewords $\boldsymbol{b}$ and $\boldsymbol{b}'$ corresponding to A and A', respectively, will differ in at least $d(\boldsymbol{c}, \boldsymbol{c}')\,d(v)$ positions with equality if the v-representation is equidistant. Because $d(\boldsymbol{c}, \boldsymbol{c}') \geq d$ with equality for some codewords $\boldsymbol{c}$ and $\boldsymbol{c}'$ in V, this proves the theorem. □

Lemma 6: The condition in Theorem 1 that $\mathbf{1}$ be a codeword in V is equivalent to the condition that 1 *not* be a root of $g(x) = 0$, where $g(x)$ is the generator polynomial of the linear cyclic code V.

Proof: The n-tuple $\mathbf{1}$ corresponds to the polynomial

$$1 + x + \cdots + x^{n-1} = (x^n - 1)/(x - 1).$$

Thus, $\mathbf{1}$ is a codeword in V, if and only if $g(x)$, which must divide $x^n - 1$, also divides $(x^n - 1)/(x - 1)$. If $x^n - 1$ has no repeated factors, this is equivalent to the condition that $x - 1$ not divide $g(x)$. But the condition $\gcd(n, p) = 1$ in Theorem 1 is precisely the condition that $x^n - 1$ have no repeated factors.

C. Constructions of Binary Constant-Weight Cyclic Codes

We will give four general constructions of binary constant-weight cyclic codes based on Theorem 1. First, however, we give some examples based on the Golay codes, cf. [3, pp. 634–650], to illustrate the method.

Example 3: Choose $p = 2$, $v = [1, 0]$ and V as the cyclic $(n, k, d) = (23, 12, 7)$ binary Golay code. Because $\mathbf{1}$ is a codeword in V, i.e., 1 is not a root of $g(x) = 0$, the construction described in Theorem 1 can be applied, with the help of Lemma 3, to obtain a binary constant-weight cyclic code with $2^{12} = 4096$ codewords of length 46 and weight 23 with $d_{\min} = 14$.

The function $A(n, d, w)$, defined as the maximum number of codewords in a (not necessarily cyclic) binary code of blocklength n, constant weight w, and minimum distance at least d, is of considerable interest in coding theory, cf. [3, pp. 524–534] and [12]. Example 3 shows that

$$A(46, 14, 23) \geq 4096$$

and, moreover, that this lower bound can be obtained with a cyclic constant-weight code. The best previous lower bound for this case [3, Theorem 33, p. 558] appears to have been

$$A(46, 14, 23) \geq 3834,$$

however it is possible to strengthen this bound as we show next.

Let $A(n, d)$, as customary, denote the maximum number of codewords in a (not necessarily cyclic) binary code of block length n and minimum distance at least d. We first observe that, when d is even, there exists a length n code with $A(n, d)$ codewords and minimum distance at least d, all of whose codewords have even weight, and that there exists as well such a code all of whose codewords have odd weight. To see this, suppose that C is any code with $A(n, d)$ codewords and minimum distance at least d where d is even. If we change, as necessary, only the last digit of each codeword to make all codewords have even Hamming weight, we obtain a code C' with $A(n, d)$ codewords whose minimum distance d' is at least $d - 1$. Because all codewords of C' have even Hamming weight, d' must be even and hence, since d is even, d' must be at least d. Complementing the last digit of every codeword in C' gives a code C'' with the same minimum distance and same number of codewords, all of whose codewords have odd weight. Now suppose that d is even, that w is odd, and that C is a code with $A(n, d)$ codewords, all of odd weight, and minimum distance at least d. For any n-tuple $\boldsymbol{b}$ with even Hamming weight, the set $\boldsymbol{b} + C$ is also a code with $A(n, d)$ codewords, all of odd weight, and minimum distance at least d. Hence, at most $A(n, d, w)$ of the codewords in $\boldsymbol{b} + C$ can lie on the surface of the Hamming sphere of radius w centered at the origin because this subset of codewords is a constant-weight cyclic code with minimum distance at least d. Considering all 2^{n-1} choices of $\boldsymbol{b}$ as an n-tuple of even weight, one concludes that the total count of codewords lying on the surface of the sphere will be at most $2^{n-1}A(n, d, w)$. On the other hand, for each codeword $\boldsymbol{c}$ in C, there are exactly $\binom{n}{w}$ choices of $\boldsymbol{b}$ such that $\boldsymbol{b} + \boldsymbol{c}$ will lie on the surface of the sphere of radius w centered at the origin, viz., the $\binom{n}{w}$ choices of $\boldsymbol{b}$ as an n-tuple at distance w from $\boldsymbol{c}$. Thus, the total count of codewords lying on the surface of the sphere will be exactly $\binom{n}{w}A(n, d)$. (An entirely similar argument applies when w is even.) We have proved the following bound, which improves by a factor of 2 on the right, the bound on $A(n, d, w)$ in [3, Theorem 33, p. 558] and which is due to van Pul (see Acknowledgment at end of paper).

Improved Lower Bound on A(n, d, w): For every even integer d, $0 \leq d \leq n$, and every w, $0 \leq w \leq n$,

$$A(n, d, w) \leq \binom{n}{w} \frac{1}{2^{n-1}} A(n, d). \tag{3}$$

This improved bound gives

$$A(46, 14, 23) \geq 7668,$$

which shows that there are certainly constant-weight codes of length 46, Hamming weight 23 and minimum distance 14 that are better than the cyclic constant-weight code of Example 3—whether there are better *cyclic* constant-weight codes is a different and open question.

Example 4: Choose $p = 3$, $v = [1, 0, 0]$ and V as the cyclic $(n, k, d) = (11, 6, 5)$ ternary Golay code. Because $\mathbf{1}$ is a codeword in V, the construction described in Theorem 1 can again be applied, with the help of Lemma 3, to obtain a binary constant-weight cyclic code with $3^6 = 729$ codewords of Length 33 and weight 11 with $d_{\min} = 10$.

It follows from Example 4 that

$$A(33, 10, 11) \geq 729.$$

The improved lower bound (3), which appears to be better than any previous bound for this case, gives only $A(33, 10, 11) \geq 554$.

Our general constructions will utilize *maximum-distance-separable* (MDS) codes, i.e., linear (n, k, d) codes with $d = n - k + 1$. All Reed–Solomon (RS) codes are MDS. Moreover, for every odd prime p and every divisor n of $p - 1$ with $n > 1$, there exists for every k, $1 \leq k < n$, a p-ary linear cyclic RS (n, k, d) code that contains the codeword $\mathbf{1}$ (or, equivalently by Lemma 6, such that $\mathbf{1}$ is not a root of $g(x) = 0$). Note that $\gcd(p, p - 1) = 1$ and hence, also that $\gcd(n, p) = 1$.

Construction I: Let p be an odd prime, let n $(n > 1)$ be a divisor of $p - 1$, and let k satisfy $1 \leq k < n$. Choosing V

to be a p-ary linear cyclic (n, k, d) Reed–Solmon code and choosing v as the p-tuple $[1, 0, 0, \cdots, 0]$ yields, by Theorem 1 and Lemma 3, a binary constant-weight cyclic code with p^k codewords of length $N = np$ and weight $w = n$ that has minimum distance $d_{\min} = 2(n - k + 1)$.

Berlekamp and Justesen [13] have given, for the finite fields GF(q) where q is a power of 2, constructions of q-ary *Bose–Chaudhuri–Hocquenghem* (BCH) codes of length $n = q + 1$ that are MDS. Their results, however, are easily extended to an arbitrary finite field, cf. [14]. We will slightly modify the Berlekamp–Justesen construction to ensure that **1** is never a root of $g(x) = 0$, where $g(x)$ is the generator polynomial of the BCH code. We will refer to the resulting codes as *generalized Berlekamp–Justesen (BJ) codes*. We will use these codes in our constructions only when q is an odd prime, but we will formulate generalized BJ codes for arbitrary q as their properties are no more difficult to establish in the general case.

Lemma 7: For any prime power q and any divisor n $(n > 1)$ of $q + 1$, there exists, for every odd k $(1 \le k \le n)$, a q-ary generalized BJ code of blocklength n and dimension k that is an MDS linear cyclic code and contains the codeword **1**.

Proof: A primitive element of GF(q^2) has multiplicative order $q^2 - 1 = (q - 1)(q + 1)$. Thus, for any divisor n $(n > 1)$ of $q + 1$, we can choose an element α in GF(q^2) with multiplicative order n. Moreover, $(\alpha^i)^{q+1} = 1$ or, equivalently, $(\alpha^i)^q = \alpha^{-i}$ for all i. But if $\alpha^i \notin$ GF(q), then $(\alpha^i)^q = \alpha^{-i}$ is its only conjugate with respect to GF(q). It follows that if n is even so that $\alpha^{n/2} = -1 \in$ GF(q), then, for any even d (say $d = 2\delta + 2$), the set of $d - 1 = 2\delta + 1$ consecutive powers of α

$$\{\alpha^{n/2-\delta}, \alpha^{n/2-\delta+1}, \cdots, \alpha^{n/2}, \alpha^{n/2+1}, \cdots, \alpha^{n/2+\delta}\}$$

is closed under conjugation in GF(q). Hence, the monic polynomial $g(x)$ having these $d - 1$ elements as roots of $g(x) = 0$ is a polynomial of degree $n - k = d - 1$ with coefficients in GF(q), and is thus, the generator polynomial of a BCH code with minimum distance $d = n - k + 1$, i.e., an MDS code. Moreover, **1** is not a root of $g(x) = 0$ so that, by Lemma 6, **1** is a codeword in this linear cyclic code. Similarly, if n is odd, then for any odd d (say $d = 2\delta + 1$), the set of $d - 1 = 2\delta$ consecutive powers of α

$$\{\alpha^{(n+1)/2-\delta}, \alpha^{(n+1)/2-\delta+1}, \cdots, \alpha^{(n+1)/2}, \alpha^{(n+1)/2+1}, \cdots, \alpha^{(n+1)/2+\delta-1}\}$$

is the root set of $g(x) = 0$ for the appropriate generator polynomial $g(x)$. Our argument has shown that n and $d = n - k + 1$ must either both be even or both be odd, which is equivalent to the condition that k be odd. □

Construction II: Let p be an odd prime, let n $(n > 1)$ be a divisor of $p + 1$, and let k $(1 \le k \le n)$ be odd. Choosing V to be a p-ary linear cyclic (n, k, d) generalized BJ code and choosing v as the p-tuple $[1, 0, 0, \cdots, 0]$ yields, by Theorem 1 and Lemma 3, a binary constant-weight cyclic code with p^k codewords of length $N = np$ and weight $w = n$ that has minimum distance $d_{\min} = 2(n - k + 1)$.

The essential difference between the codes of Constructions I and II is that the latter codes are two p-ary digits longer when n is chosen as large as possible for the same p.

The cyclic codes given by Construction I with $n = p - 1$ and by Construction II with $n = p + 1$ are "asymptotically optimum" constant-weight codes in the sense that, for fixed k, they meet the *Johnson upper bound*, cf. [3, Corollary 5, p. 528], with equality as $p \to \infty$, as will now be explained. This upper bound states that, for even d (which is no real restriction since the minimum distance of a binary constant-weight code is always even),

$$A(N, d, w) \le \prod_{i=0}^{w-d/2} \frac{N - i}{w - i}. \tag{4}$$

For Construction I with $n = p - 1$, we have $w - d/2 = k - 1$, $N = p(p - 1)$, and $w = p - 1$; so that the right side of (4) becomes

$$\prod_{i=0}^{k-1} \frac{p(p-1) - i}{p - 1 - i} = p^k(1 + o(p)),$$

where $o(p) \to 0$ as $p \to \infty$. The codes of Construction I have exactly p^k codewords and hence can be said to be "asymptotically optimum." The codes of Construction II with $n = p + 1$, by an entirely similar argument, are also "asymptotically optimum" in the same sense.

If we replace the sequence $[1, 0, 0, \cdots, 0]$ by an m-sequence or by a Legendre sequence as the choice for v in Constructions I and II, then, with the aid of Lemmas 4 and 5, we obtain the two following constructions.

Construction III: Let p be a prime such that $(p - 1)/2$ is odd, let n $(n > 1)$ be a divisor of $p - 1$ and let k satisfy $1 \le k \le n$. Then Construction I altered only in that v is chosen to be a Legendre sequence of length p (or, alternatively, as an m-sequence of length p in case p is a Mersenne prime) yields a binary constant-weight cyclic code with p^k codewords of length $N = np$ and weight $w = n(p + 1)/2$ that has minimum distance $d_{\min} = (n - k + 1)(p + 1)/2$.

Construction IV: Let p be a prime such that $(p - 1)/2$ is odd, let n $(n > 1)$ be a divisor of $p + 1$ and let k $(1 \le k \le n)$ be odd. Then Construction II altered only in that v is chosen to be a Legendre sequence of length p (or, alternatively, as an m-sequence of length p in case p is a Mersenne prime) yields a binary constant-weight cyclic code with p^k codewords of length $N = np$ and weight $w = n(p + 1)/2$ that has minimum distance $d_{\min} = (n - k + 1)(p + 1)/2$.

The Plotkin bound [3, p. 41] asserts that any binary code of length N and minimum distance d for which $N < 2d$ has at most $2\lfloor d/(2d - N) \rfloor$ codewords where $\lfloor \cdot \rfloor$ denotes the integer part of the enclosed number. Letting $\Delta = 2d - N > 0$ and noting that

$$\begin{aligned} 2\lfloor d/(2d - N) \rfloor &= 2\lfloor (N/2 + \Delta/2)/\Delta \rfloor \\ &= 2\lfloor N/(2\Delta) + 1/2 \rfloor \\ &\le 2\lfloor N/2 + 1/2 \rfloor \le N + 1, \end{aligned}$$

it follows that, for any binary code of length N with more than $N+1$ codewords,

$$\frac{d}{N} \leq \frac{1}{2}, \tag{5}$$

which we will call the *low-rate Plotkin bound*. The length N binary codes of Construction III with $n = p-1$ and those of Construction IV with $n = p+1$ have more than $N+1$ codewords for $k \geq 2$ and for $k \geq 3$, respectively. But, for the codes of either construction and for any fixed k,

$$\frac{d_{\min}}{N} = \frac{1}{2} + o(p), \tag{6}$$

where $o(p) \to 0$ as $p \to \infty$. Thus, the codes of both constructions are "asymptotically optimum" in the sense that they achieve the low-rate Plotkin bound as $p \to \infty$. In fact, it is easy to see that, for any sequence of codes with increasing p and nondecreasing k obtained from Construction III (or from Construction IV) using the largest possible n, (6) still holds provided only that $k/p \to 0$ as $p \to \infty$. Thus, these codes are "asymptotically optimum" in a rather strong sense.

IV. Constructions of Cyclically Permutable Codes

Two N-tuples $\boldsymbol{b}$ and $\boldsymbol{b}'$ are said to be in the same *cyclic equivalence class* if $S^i(\boldsymbol{b}) = \boldsymbol{b}'$ for some i, $0 \leq i < N$. If $\boldsymbol{b}$ has cyclic order j (cf. Section II), then the cyclic equivalence class containing $\boldsymbol{b}$ contains a total of j N-tuples and is also said to have order j. It follows that a *cyclically permutable code* (cf. Section I) can equivalently be defined as a binary block code such that its codewords lie in distinct cyclic equivalence classes and each of these classes has order equal to the blocklength. The *cyclic minimum distance*, d_c, of a cyclically permutable code is defined as the minimum Hamming distance from a codeword to one of its own distinct cyclic shifts or to some cyclic shift of another codeword. In other words, d_c is the minimum distance $d_{\min}$ of the binary cyclic code obtained from the cyclically permutable code by replacing each of its codewords with all the N-tuples in the cyclic equivalence class (necessarily of order N) containing that codeword. We will write CPC (N, M_c, d_c) to denote a cyclically permutable code of length N having M_c codewords and cyclic minimum distance d_c. We remark that there has been a paucity of constructions for cyclically permutable codes, the only such constructions known to us being those of [1], [15], and [16].

If one selects at most one codeword from each cyclic equivalence class of order N in a binary cyclic code of length N and minimum distance $d_{\min}$, then one obviously obtains a CPC (N, M_c, d_c) code with $d_c \geq d_{\min}$ where M_c is the number of codewords selected. For such a procedure to qualify as a "construction," the selection rule must be easily implementable. For the construction to be "good," the number M_c of codewords selected should be close to its maximum possible value M/N (where M is the number of codewords in the cyclic code) and the cyclic code should have $d_{\min}$ close to the maximum possible value for the given M and N. We now show how to construct some cyclically permutable codes that are "good" in this sense and that have the further property of being constant-weight codes, which is often a requirement in applications (see, e.g., Section V). Our constructions will be based on the constant-weight cyclic codes of Constructions I and II, which guarantees large $d_{\min}$. We now develop methods for selecting cyclically distinct codewords of full cyclic order from these codes that are easily implementable and that select close to M/N codewords.

With an n-tuple $\boldsymbol{c} = [c_0, c_1, \cdots, c_{n-1}]$ having components in GF (q), we associate in the usual manner the polynomial $c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$, and we note that the t-times rightward cyclically shifted n-tuple $S^t(\boldsymbol{c})$ corresponds in this way to the polynomial $x^t c(x) \bmod (x^n - 1)$, where here and hereafter we write $c(x) \bmod (x^n - 1)$ to denote the remainder when $c(x)$ is divided by $x^n - 1$. The following result is the key to our codeword selection method for the codes of Construction I. We state this result more generally than we actually require because it may be of some independent interest in the theory of RS codes.

Theorem 2: Let α have multiplicative order n $(n > 1)$ in GF (q) and let $e(x) = 1 + x + \cdots + x^{n-1}$. Then the code polynomials (i.e., the polynomials corresponding to codewords) in the (n, k, d) Reed–Solomon code with generator polynomial $g(x)$ such that $\alpha, \alpha^2, \cdots, \alpha^{n-k}$ are the roots of $g(x) = 0$ can be written as

$$c(x) = \sum_{j=0}^{k-1} u_j e(\alpha^j x), \tag{7}$$

where $u_0, u_1, \cdots, u_{k-1}$ may be considered as the k q-ary information symbols. Moreover, the q^{k-1} codewords c corresponding to $u_1 = 1$ all have full cyclic order n and are cyclically distinct.

Remark: Theorem 2 shows that the vast majority of codewords in an RS code have full cyclic order n. Taking $n = q - 1$, the maximum possible value, we see from Theorem 2 that at least $nq^{k-1} = q^k - q^{k-1}$ of the q^k codewords in the RS code have full cyclic order. We remark further that Vajda and Einarsson [17] have made use in a frequency-hopping scheme of precisely the codes given by the choice $k = 3$ in Theorem 2. Their codes, in which u_0 was the message and u_2 was the user's address, have desirable synchronization capabilities.

Proof: Because $\alpha, \alpha^2, \cdots, \alpha^n$ are all the roots of $x^n - 1 = 0$, it follows that α^i is a zero of $e(x) = (x^n - 1)/(x - 1)$ for $1 \leq i < n$ but not for $i = n$. Thus, for $1 \leq i \leq n$, α^i is a zero of $e(\alpha^j x)$ except for $i = n - j$. It follows that the right side of (7) is the general form of a q-ary polynomial having α^i as a zero for $1 \leq i \leq n - k$, i.e., a general code polynomial in the RS code whose generator polynomial has α^i as a zero for $1 \leq i \leq n - k$, and that we may take the coefficients $u_0, u_1, \cdots, u_{k-1}$ in (7) as the information symbols of this code.

We next note that $e(1) = 1 + 1 + \cdots + 1 = n$, where of course n must be interpreted as an element of GF (q), i.e.,

as reduced modulo the characteristic of this field. Now replacing x by α^{n-i} in (7) gives

$$u_i = c(\alpha^{n-i})/n, \qquad \text{for } 0 \le i < k, \tag{8}$$

as an explicit formula for the information symbols in the RS code. Next suppose that $\boldsymbol{c}^* = \boldsymbol{S}^t(\boldsymbol{c})$ or, equivalently, that $c^*(x) = x^t c(x) \bmod (x^n - 1)$. It follows from (8) that

$$u_i^* = \alpha^{(n-i)t} u_i = \alpha^{-it} u_i, \qquad \text{for } 0 \le i < k. \tag{9}$$

Suppose further that $u_1 = 1$. Then (9) gives $u_1^* = \alpha^{-t}$, which cannot be 1 for $1 \le t < n$ because α has multiplicative order n. Thus, $\boldsymbol{c} = \boldsymbol{S}^t(\boldsymbol{c})$ is impossible for $1 \le t < n$, i.e., $\boldsymbol{c}$ has full cyclic order n. Because $\boldsymbol{c}^* = \boldsymbol{S}^t(\boldsymbol{c})$ is possible only for $t = 0$, i.e., for $\boldsymbol{c}^* = \boldsymbol{c}$, it follows further that different codewords with $u_1 = 1$ lie in distinct cyclic equivalences classes, as was to be shown. □

In the following construction, we exploit Theorem 2 for the case where q is a prime.

Construction V: Let p, n, k, and v be as in Construction I with the further *proviso* that $p \ge 5$ and $3 \le k < n$. Let α have multiplicative order n in GF(p) and let V be the $(n, k, d = n - k - 1)p$-ary Reed–Solomon code described in Theorem 2. Let each codeword $\boldsymbol{c}$ in V with corresponding information vector $\boldsymbol{u} = [u_0, u_1, \cdots, u_{k-1}]$ determine a binary N-tuple $\boldsymbol{b}$ in the same manner as in Construction I with $N = np$. Then the set $\boldsymbol{B}$ of binary N-tuples $\boldsymbol{b}$ corresponding to those n-tuples $\boldsymbol{c}$ in V for which $u_0 = 0$ and $u_1 = 1$ is a constant-weight $w = n$ CPC(N, M_c, d_c) code C with $M_c = p^{k-2}$ codewords and cyclic minimum distance $d_c \ge 2(n - k + 1)$.

Proof: It follows from our remarks at the beginning of this section that it suffices to show that each N-tuple in $\boldsymbol{B}$ has cyclic order $N = np$ and that the p^{k-2} N-tuples in $\boldsymbol{B}$ are cyclically distinct.

Let $\boldsymbol{c}$ be the codeword in the p-ary RS code V corresponding to $\boldsymbol{b}$, let $\boldsymbol{u}$ be the corresponding information vector in the RS code, and let A be the $p \times n$ array corresponding to $\boldsymbol{b}$. Let C be the subset of the RS code V for which $u_0 = 0$ and $u_1 = 1$. The array $A' = \boldsymbol{D}^i(A)$ corresponds by Lemma 2 to the RS codeword $\boldsymbol{c}' = \boldsymbol{c} + i\boldsymbol{1}$ where on the right i must be interpreted as an element of GF(p). Suppose now that $\boldsymbol{c} \in C$ so that $u_0 = 0$. But $\boldsymbol{c}'' = \boldsymbol{1}$ is also a codeword in the RS code V and has $u_0'' = 1$ and $u_1'' = 0$, as follows from (7) and the fact that $c''(x) = e(x)$. Thus, $\boldsymbol{c}' = \boldsymbol{c} + i\boldsymbol{1}$ has $u_0' = i$, from which we conclude that $A' \ne A$ for $1 \le i < p$. It follows that the array A has vertical cyclic order p. Next, we note that $\boldsymbol{c}^* = \boldsymbol{S}^j(\boldsymbol{c})$ corresponds to $A^* = \boldsymbol{R}^j(A)$. It now follows from Theorem 2 that, because $\boldsymbol{c} \in C$ so that $u_1 = 1$, $\boldsymbol{c}$ has cyclic order n or, equivalently, A has horizontal cyclic order n. Moreover, the facts that $\gcd(p, p - 1) = 1$ and that n divides $p - 1$ imply that $\gcd(p, n) = 1$. Thus, the array A under the operator $\boldsymbol{DR} = \boldsymbol{RD}$ has order np. It now follows from Lemma 1 that $\boldsymbol{b}$ has cyclic order $N = np$.

It remains only to show that the codewords in $\boldsymbol{B}$ are cyclically distinct, i.e., to show that if $\boldsymbol{b} \in \boldsymbol{B}$ then $\boldsymbol{S}^t(\boldsymbol{b}) \notin \boldsymbol{B}$ for $1 \le t < N$. Suppose then that $\boldsymbol{b} \in \boldsymbol{B}$. Because $\boldsymbol{b}$ has cyclic order $N = np$, it follows from the proof of Lemma 1 that $\boldsymbol{b}' = \boldsymbol{S}^t(\boldsymbol{b})$ corresponds to $A' = \boldsymbol{R}^t(\boldsymbol{D}^t(A)) = \boldsymbol{R}^j(\boldsymbol{D}^i(A))$ where $i = t \bmod p$ and $j = t \bmod n$. But $\boldsymbol{D}^i(A)$ corresponds to $\boldsymbol{c} + i\boldsymbol{1}$ and thus, $\boldsymbol{R}^j(\boldsymbol{D}^i(A))$ corresponds to $\boldsymbol{c}' = \boldsymbol{S}^j(\boldsymbol{c} + i\boldsymbol{1}) = \boldsymbol{S}^j(\boldsymbol{c}) + i\boldsymbol{1}$. Letting $\boldsymbol{c}''' = \boldsymbol{S}^j(\boldsymbol{c})$ and recalling that the RS codeword $\boldsymbol{c}'' = \boldsymbol{1}$ has $u_0' = 1$ and $u_1' = 0$, we see that $u_1' = u_1'''$. Thus, $\boldsymbol{c}'$ can be in the subset C or, equivalently, $\boldsymbol{b}$ can be in the subset $\boldsymbol{B}$ only if $u_1''' = 1$, which by Theorem 2 implies that $j = 0$. For $j = 0$, $\boldsymbol{c}' = \boldsymbol{c} + i\boldsymbol{1}$ so that $u_0' = u_0 + i = i$. Thus, $\boldsymbol{c}'$ can be in C or, equivalently, $\boldsymbol{b}'$ can be in $\boldsymbol{B}$ only if $i = 0$. But $i = j = 0$ implies $t = 0$, and we have thus shown that $\boldsymbol{S}^t(\boldsymbol{b})$ can be in $\boldsymbol{B}$ for $0 \le t < N$ only when $t = 0$. It follows that the N-tuples in $\boldsymbol{B}$ are cyclically distinct, which completes the proof of the validity of Construction V. □

We now consider the construction of cyclically permutable codes based on the generalized BJ codes of Lemma 8. We begin with a result analogous to Theorem 2 for RS codes.

Theorem 3: Let α have multiplicative order n in GF(q^2), where n is a divisor of $q + 1$, and let $e(x) = 1 + x + \cdots + x^{n-1}$. Then the code polynomials in the (n, k, d) MDS generalized Berlekamp–Justesen code such that $\alpha^{n/2-\delta}, \alpha^{n/2-\delta+1}, \cdots, \alpha^{n/2}, \alpha^{n/2+1}, \cdots, \alpha^{n/2+\delta}$, where $d = 2\delta + 2$, are the zeros of the generator polynomial when n is even [or $\alpha^{(n+1)/2-\delta}, \alpha^{(n+1)/2-\delta+1}, \cdots, \alpha^{(n+1)/2}, \alpha^{(n+1)/2+1}, \cdots \alpha^{(n+1)/2+\delta-1}$, where $d = 2\delta + 1$, are the zeros of the generator polynomial when n is odd] can be written as

$$c(x) = u_0 e(x) + \sum_{j=1}^{n/2-\delta-1} \left[u_j e(\alpha^j x) + u_j^q e(\alpha^{-j} x)\right] \tag{10}$$

when n is even (or can be written as

$$c(x) = u_0 e(x) + \sum_{j=1}^{(n-1)/2-\delta} \left[u_j e(\alpha^j x) + u_j^q e(\alpha^{-j} x)\right] \tag{11}$$

when n is odd) where $u_0 \in \mathrm{GF}(q)$ and $u_j \in \mathrm{GF}(q^2)$ for $j > 0$ may be considered as the information symbols. The q^{k-2} codewords c corresponding to $u_1 = 1$ in (10) (or in (11)) all have cyclic order n and are cyclically distinct.

Proof: We consider only the case where n is even, as the proof for odd n is entirely similar. The same argument used in the proof of Theorem 2 now shows that

$$c(x) = \sum_{j=-n/2+\delta+1}^{n/2-\delta-1} u_j e(\alpha^{-j} x), \tag{12}$$

where $u_j \in \mathrm{GF}(q^2)$ for all j, is a general code polynomial in the Reed–Solomon code over GF(q^2) for which $\alpha^{n/2-\delta}, \alpha^{n/2-\delta+1}, \cdots, \alpha^{n/2}, \alpha^{n/2+1}, \cdots, \alpha^{n/2+\delta}$ are zeros of the generator polynomial $g(x)$. But the generalized BJ code is just the q-ary subcode of this RS code and hence $c(x)$ as given by (12) is a code polynomial in the BJ code, if

and only if $[c(x)]^q = c(x^q)$. But (12) gives

$$[c(x)]^q = \sum_{j=-n/2+\delta+1}^{n/2-\delta-1} u_j^q e(\alpha^{-j}x^q), \qquad (13)$$

where we have used the fact that $[e(x)]^q = e(x^q)$ and hence that $[e(\alpha^j x)]^q = e(\alpha^{jq}x^q) = e(\alpha^{-j}x^q)$, where the last step follows from the fact, shown in the proof of Lemma 7, that $\alpha^{qi} = \alpha^{-i}$ for all i. Comparing (12) and (13), we see now that $[c(x)]^q = c(x^q)$ or, equivalently, that $c(x)$ is a code polynomial in the generalized BJ code, if and only if $u_0 \in$ GF(q) and $u_{-j} = u_j^q$ for $1 \le j \le n/2 - \delta - 1$. Using these relations in (12) now gives the desired expression (10) for a generalized BJ code polynomial. The proof that $u_1 = 1$ in (12) gives cyclically distinct codewords $\boldsymbol{c}$ all having cyclic order n is entirely analogous to the corresponding part of the proof of Theorem 2 and will thus be omitted. □

Construction VI: Let p, n, and k be as in Construction II with the further *proviso* that $p \ge 5$ and $5 \le k < n$. Let α have multiplicative order n in GF(p^2) and let V be the $(n, k, d = n - k + 1)$ p-ary generalized Berlekamp–Justesen code described in Theorem 3. Let each codeword $\boldsymbol{c}$ in V with information symbols $u_0, u_1, \cdots, u_{(k-1)/2}$ (where $u_0 \in$ GF(q) and $u_i \in$ GF(q^2) for $i > 0$) determine a binary N-tuple $\boldsymbol{b}$ in the same manner as in Construction II with $N = np$. Then the set B of binary N-tuples $\boldsymbol{b}$ corresponding to those n-tuples $\boldsymbol{c}$ in V for which $u_0 = 0$ and $u_1 = 1$ is a constant-weight $w = n$ CPC(N, M_c, d_c) code with $M_c = p^{k-3}$ codewords and cyclic minimum distance $d_c \ge 2(n - k + 1)$.

The proof of the validity of this construction is entirely analogous to the proof of the validity of Construction V and will be omitted.

We now consider the efficiency of the selection procedures by which the constant-weight cyclically permutable codes of Constructions V and VI were obtained. The RS-based Construction I gives a binary constant-weight cyclic code with $M = p^k$ codewords of length $N = p(p-1)$ when n is chosen as its maximum value $p - 1$. Thus, $M/N = p^{k-1}/(p-1)$ is an upper bound on the number of codewords in a cyclically permutable code selected therefrom. But Construction V selects $M_c = p^{k-2}$ codewords, which is smaller than the upper bound by only the factor $(p-1)/p$, which is nearly 1 for large primes p. The generalized BJ-based Construction II gives a binary constant-weight cyclic code with $m = p^k$ codewords of length $N = p(p+1)$ when n is chosen as its maximum value $p + 1$. Thus, $M/N = p^{k-1}/(p+1)$ is an upper bound on the number of codewords in a cyclically permutable code selected therefrom. Construction VI, however, selects only $M_c = p^{k-3}$ codewords, which is smaller than the upper bound by the factor $(p+1)/p^2 \approx 1/p$. Thus, this selection procedure is not highly efficient for large p, but the codes nonetheless appear to be of interest because of their greater length $N = p(p+1)$ and hence, their greater cyclic minimum distance d_c for the same number of codewords, compared to the codes of Construction V.

V. Protocol Sequence Sets for the M-Active-Out-of-T Users Collision Channel Without Feedback

In this section, we will show that constant-weight cyclically permutable codes, such as those constructed in the previous section, provide a very natural solution to an interesting random-accessing problem. The situation to be considered is that in which each one of a total population of T users occasionally has traffic to send over a common communications channel. This traffic is in the form of "packets" of some fixed length that we assume take values in the finite field GF(Q) for some, in general large, Q. The time axis is assumed to be partitioned into slots whose duration corresponds to the transmission time for one packet; it is further assumed that all users know the slot boundaries but are otherwise unsynchronized. When a user transmits a packet, he must transmit it exactly within a slot.

The common communications channel is assumed to be the *collision channel without feedback* [18], [19]. If, in a particular slot, none of the users are sending a packet (in which case we say that each user "sends" the *silence symbol* Λ), then the channel output in that slot is the silence symbol Λ. If exactly one user is sending a packet in a particular slot, then the channel output in that slot is this packet value, which is an element of GF(Q). If two or more users are sending packets in a particular slot, then the channel output in that slot is the collision symbol Δ. There is no feedback available to inform the senders of the channel outputs in previous slots.

Each user, say user i, has a *protocol sequence*, which is a binary sequence $s_i = [s_{i1}, s_{i2}, \cdots, s_{iN}]$ of length N that controls his sending of packets in the following manner. When user i becomes active (after some period of inactivity), he *must* send a packet in the jth slot $(1 \le j \le N)$ of this activity if $s_{ij} = 1$ and must be silent in this slot if $s_{ij} = 0$. He continues to use his protocol sequence periodically in this manner until he has no further messages to send, in which case he again becomes inactive and must remain inactive for at least $N - 1$ slots. If s_i has Hamming weight w_i, then user i will send w_i packets in each *frame* of N slots corresponding to his protocol sequence. User i will code his packets (i.e., transmit redundant packets) so that packets "last" in collisions can, under specified conditions, be recovered at the receiver. The task of the receiver in each received frame of N consecutive slots is two-fold, viz.,

1) to determine which users, if any, are *frame-active* in the sense that they have sent packets in this received frame and that their protocol sequence begins in the first slot of this received frame (the *identification problem*), and
2) to determine for each frame-active user, say user i, the transmitted values of his packets in those w_i slots of this received frame where user i has sent packets (the *decoding problem*).

The random-accessing problem where in each received frame at most M out of the total T of users can be active in the sense of sending at least one packet in this frame was

introduced in [20] and [21]. The set $\{s_1, s_2, \cdots, s_T\}$ of binary sequences of length N is said to be a (T, M, N, σ) *protocol sequence set* if, when these sequences are used as protocol sequences for the T users and provided that at most M of the users are active in each received frame, each frame-active user can be identified by the receiver and at least σ of the packets transmitted by each frame-active user are sent without collision. The following theorem shows how constant-weight cyclically permutable codes can be used as (T, M, N, σ) protocol sequence sets.

Theorem 4: For any integer σ with $1 \le \sigma \le w$, a binary constant-weight w cyclically permutable code CPC $(N, M_c = T, d_c)$ is a (T, M, N, σ) protocol sequence set for

$$M = \min\{T, \lfloor (w-1)/(w-d_c/2) \rfloor, \lfloor (w-\sigma)/(w-d_c/2) \rfloor + 1\}. \quad (14)$$

Example 5: Taking $p = 13$, $n = 12$ and $k = 4$ in Construction V yields a binary, constant-weight $w = 12$, cyclically permutable code CPC $(N = 156, M_c = 169, d_c = 18)$. By Theorem 4, this code can be used as a $(T = 169, M, N = 156, \sigma = 6)$ protocol sequence set for $M = \min\{169, 3, 3\} = 3$. In other words, provided that at most $M = 3$ out of the $T = 169$ users are active in each received frame of $N = 156$ slots, each frame-active user will be guaranteed at least $\sigma = 6$ collision-free packet transmissions among the $w = 12$ packets that he sends in a frame.

The *correlation* between two binary N-tuples is defined as the number of positions in which both contain a 1. Our proof of Theorem 4 will be based on the following simple fact about such correlations.

Lemma 8: In a constant-weight w CPC (N, M_c, d_c) code, the correlation ρ between any codeword and its distinct cyclic shifts or between any cyclic shifts of any two distinct codewords satisfies

$$\rho \le w - d_c/2.$$

Proof: It follows from the definition of a cyclically permutable code that any two N-tuples as hypothesized in the lemma are separated by Hamming distance at least d_c, and of course each has Hamming weight w. But two binary N-tuples of Hamming weight w at Hamming distance d both contains 1's in exactly $w - d/2$ positions. □

Proof of Theorem 4: We first consider a strategy by which the receiver can solve the identification problem. For an arbitrary received frame of N slots, let the binary N-tuple $\tau = [\tau_1, \tau_2, \cdots, \tau_N]$ be the *transmission-activity vector* in the manner that τ_j is 0 or 1 according as the silence symbol or some other symbol (i.e., a packet or the collision symbol Δ), respectively, is received in the jth slot of this frame. The receiver decides that user i is frame-active in this frame if and only if the 1's in τ cover all 1's in the protocol sequence s_i. If user i is indeed frame-active, this decision rule will always correctly so indicate. However, if user i is not frame-active, then this decision rule can err. If user i is not frame-active, if at most M users are active in the frame, and if ρ is the maximum number of 1's that the transmission-activity vector of any one of these users alone would cover in s_i, then $M\rho < w$ is a sufficient condition for the identification decision on user i to be correct. But $\rho \le w - d_c/2$ as follows from Lemma 8 and the fact the transmission activity vector of any other user alone must be some cyclic shift of his protocol sequence [possibly with some digits on the left or on the right replaced by 0's in case the activity of that user begins or ends, respectively, somewhere within the frame]. Thus, $M(w - d_c/2) < w$ or, equivalently, $M(w - d_c/2) \le w - 1$ is a sufficient condition for correct identification. It follows that the largest M consistent with this sufficient condition for correct identification is

$$M = \lfloor (w-1)/(w-d_c/2) \rfloor. \quad (15)$$

We next derive a sufficient condition for each frame-active user to have at least σ successful packet transmissions among his w packet transmission attempts when at most M users are active in the frame. Suppose user i is frame-active. Because each of the at most $M - 1$ other users can cause at most $w - d_c/2$ collisions with the packets of user i, it follows that user i achieves at least $w - (M-1)(w - d_c/2)$ successes. Thus, $\sigma \ge w - (M-1)(w - d_c/2)$ or, equivalently,

$$M = \lfloor (w-\sigma)/(w-d_c/2) \rfloor + 1 \quad (16)$$

is the largest M that guarantees at least σ successes for each frame-active user.

Trivially, however, $M \le T$ must be satisfied. Thus, if M is the minimum of T and the two integers on the right in (15) and (16), we are assured that identification of the frame-active users will be correct and that each of these users will achieve at least σ successful packet transmission in the frame. □

We now briefly consider how the users can code their packets so that each user can send σ information packets in each frame of his activity and the receiver can correctly decode these packets. Each user employs an $(n' = w, k' = \sigma, d' = w - \sigma + 1)$ shortened RS code over GF (Q) to code his σ information packets into his w transmitted packets. Such a code exists provided only that $w \le Q + 1$ when we allow the use of the so-called doubly-extended RS codes, cf. [2, p. 221]. If a user is frame-active and has σ successful packet transmissions, the decoding problem at the receiver is equivalent to having erasures in the at most $w - \sigma$ positions where this user's packets suffer collisions. Because $d' = w - \sigma + 1$, the receiver can always correct these erasures by a standard erasure-correcting algorithm for the RS code and hence, can correctly recover the σ information packets from this user.

Because a (T, M, N, σ) protocol-sequence set allows each of the M active users to send σ information packets successfully in a frame of N slots when the users code their packets as described above, it follows that R_{sum}, the total information transmission rate that can be achieved, is

$$R_{\text{sum}} = (M\sigma)/N \quad \text{(packets/slot)}. \quad (17)$$

For instance, in Example 5, a sum rate of

$$R_{\text{sum}} = (3 \times 6)/156 = 3/26 \quad \text{(packets/slot)}$$

can be achieved.

VI. Conclusion

We have given in this paper rather many constructions of good binary constant-weight cyclic codes and constant-weight cyclically permutable codes. It will be obvious to the reader that a variety of further such codes can be constructed from Theorem 1 by different choices of the sequence v used in the v-representation of GF(p) and by different choices of the p-ary linear cyclic code V. We have limited ourselves primarily to the case where V is an MDS code as this guarantees good codes but Examples 3 and 4, which used the Golay codes, show that other choices of V can yield very good binary constant-weight cyclic codes.

As an application of constant-weight cyclically permutable codes, we have considered their use as protocol-sequence sets for the M-active-out-of-T-users collision channel without feedback. We expect these codes to find other applications in various problems of an essentially asynchronous nature. For instance, we suspect that these codes could form the basis for the construction of some interesting comma-free codes, cf. [22].

Acknowledgment

The authors are grateful to C. van Pul of Philips Crypto in Eindhoven, The Netherlands, both for communicating to them the improved lower bound on $A(n, d, w)$ given in Section III-C and for suggesting the use of the Legendre sequences in Constructions III and IV. The authors are also grateful to the referees for their careful reading of the previous manuscripts and their helpful comments.

References

[1] E. N. Gilbert, "Cyclically permutable error-correcting codes," *IEEE Trans. Inform. Theory*, vol. 9, pp. 175–182, July 1963.

[2] R. E. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1984.

[3] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes*. Amsterdam: North-Holland, 1977.

[4] W. W. Peterson and E. J. Weldon, Jr., *Error-Correcting Codes*. Cambridge, MA: M.I.T. Press, 1972.

[5] V. A. Zinoviev, "Cascade equal-weight codes and maximal packing," *Probl. Contr. Inform. Theory*, vol. 12, pp. 3–9, 1983.

[6] G. Einarsson, "Address assignment for a time-frequency-coded spread-spectrum system," *Bell Syst. Tech. J.*, vol. 59, pp. 1241–1255, Sept. 1980.

[7] W. H. Kautz and R. C. Singleton, "Nonrandom binary superimposed codes," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 363–377, Oct. 1964.

[8] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread Spectrum Communications*, vol. I. Rockville, MD: Computer Science Press, 1985.

[9] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*, 4th ed. London: Oxford Univ. Press, 1960.

[10] A. N. Boehmer, "Binary pulse compression codes," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 156–167, Apr. 1967.

[11] R. Turyn, "Optimum codes study," Sylvania Electron. Syst., final rep., Contract AF19(604)-5473, Jan. 29, 1960.

[12] A. E. Brouwer, J. B. Shearer, N. J. A. Sloane, and W. D. Smith, "A new table of constant weight codes," *IEEE Trans. Inform. Theory*, vol. 36, pp. 1334–1380, Nov. 1990.

[13] E. R. Berlekamp and J. Justesen, "Some long cyclic linear binary codes are not so bad," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 351–356, May 1974.

[14] V. C. da Rocha, Jr., Maximum distance separable multilevel codes," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 547–548, May 1984.

[15] D. E. Maracle and C. T. Wolverton, "Generating cyclically permutable codes," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 554–555, July 1974.

[16] P. G. Neumann, "On a class of cyclically permutable error-correcting codes," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 75–78, Jan. 1964.

[17] I. Vajda and G. Einarsson, "Code acquisition for a frequency-hopping system," *IEEE Trans. Commun.*, vol. COM-35, pp. 566–568, May 1987.

[18] J. L. Massey, "The capacity of the collision channel without feedback," *Abstracts of Papers, IEEE Int. Symp. Inform. Theory*, p. 101, 1982.

[19] J. L. Massey and P. Mathys, "The collision channel without feedback," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 192–204, March 1985.

[20] B. S. Tsybakov and N. B. Likhanov, "Packet Communication on a channel without feedback," *Probl. Inform. Transm.*, vol. XIX, no. 2, pp. 69–84, 1983.

[21] L. A. Bassalygo and M. S. Pinsker, "Limited multiple-access of a nonsynchronous channel," (in Russian) *Probl. Inform. Transm.*, vol. XIX, no. 4, pp. 92–96, 1983.

[22] S. W. Golomb, B. Gordon, and L. R. Welch, "Comma-free codes," *Canadian J. Math.*, vol. 10, pp. 202–209, 1958.