set of polynomials $\hat{F}$ is defined:

$$f_{ij} = Y_1 X_1^i Z_1^j + Y_2 X_2^i Z_2^j + \cdots + Y_v X_v^i Z_v^j - s_{ij}, \qquad i,j \in \hat{Q}, \tag{19}$$

$$h_j = C(X_j, Z_j), \qquad 1 \le j \le v, \tag{20}$$

$$l_{1j} = X_j^q - X_j, \qquad l_{2j} = Z_j^q - Z_j, \qquad l_{3j} = Y_j^q - Y_j, \qquad 1 \le j \le v. \tag{21}$$

Thus, for the case of $v(\le t)$ errors, the problem of decoding the algebraic geometry codes is equivalent to a determination of $V(\hat{F})$. It follows from (21) that $0 < |V(\hat{F})| < \infty$.

Let $E_x = \{\beta | (x_1^*, \cdots, \beta, \cdots, x_v^*, z_1^*, \cdots, z_v^*, y_1^*, \cdots, y_v^*) \in V(\hat{F})\}$ and $E_z = \{\beta | (x_1^*, \cdots, x_v^*, z_1^*, \cdots, \beta, \cdots, z_v^*, y_1^*, \cdots, y_v^*) \in V(\hat{F})\}$. Then, the theorem for decoding the algebraic geometry codes is the following.

Theorem 5: Let $g_{x_j}(X_j) \in K[X_j]$ and $g_{z_j}(Z_j) \in K[Z_j]$ denote the monic generator polynomials of the principal ideal $I(\hat{F}) \cap K[X_j]$ and $I(\hat{F}) \cap K[Z_j]$ for $j = 1, 2, \cdots, v$, respectively. Then,

$$g_{x_1}(X) = g_{x_2}(X) = \cdots = g_{x_v}(X) = \prod_{x_a^* \in E_x} (X - x_a^*), \tag{22}$$

and

$$g_{z_1}(Z) = g_{z_2}(Z) = \cdots = g_{z_v}(Z) = \prod_{z_b^* \in E_z} (Z - z_b^*), \tag{23}$$

where all of the error locations can be obtained by suitable combinations (x_a^*, z_b^*) for $1 \le a, b \le v$ of the zeros of these polynomials.

The proof of this theorem is also similar to the proofs of Theorem 8 and Corollary 9 in [1]. One only needs to notice the symmetry properties of the polynomials in $\hat{F}$ (in terms of (x_l, z_l, y_l)). Theorem 5 can be used to find v error locations from the set of syndrome equations, given in (18). For this case, both (22) and (23) can also be found by using some Gröbner basis algorithm (or any other elimination algorithm) on the above defined set of polynomials.

Acknowledgment

The authors would like to express their appreciation to Professors C. Kirfel, P. V. Kumar, and L. Welch, and to the referees, for their useful comments, which helped improve this paper. Also, the authors would like to acknowledge Professor H. Imai for having kindly suggested the skeleton of this shortened version of the originally submitted paper.

References

[1] X. Chen, I. S. Reed, T. Helleseth, and T. K. Truong, "Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance," *IEEE Trans. Inform. Theory*, this issue.

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

[3] J. L. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 122–127, Jan. 1969.

[4] E. R. Berlekamp, *Algebraic Coding Theory.* New York: McGraw-Hill, 1968.

[5] R. T. Chien, "Cyclic decoding procedures for Bose-Chaudhuri-Hocquenghem codes," *IEEE Trans. Inform. Theory*, vol. IT-10, pp. 357–363, Oct. 1964.

[6] A. Poli and L. Huguet, *Error Correcting Codes: Theory and Applications.* Englewood Cliffs, NJ: Prentice Hall International (UK) Ltd., 1992.

[7] B. Buchberger, "Gröbner bases: An algorithmic method in polynomial ideal theory," in *Multidimensional Systems Theory*, N. K. Bose, Ed. pp. 184–232, Dordrecht: Reidel, 1985, pp. 184–232.

[8] S. Lang, *Algebra*, 2nd ed. Menlo Park, CA: Addison-Wesley, 1984.

[9] J. Justesen, K. J. Larsen, H. E. Jensen, and T. Høholdt, "Fast decoding of codes from algebraic plane curves," *IEEE Trans. Inform. Theory*, vol. 38, pp. 111–119, Jan. 1992.

Decoding of Convolutional Codes Using a Syndrome Trellis

V. Sidorenko and V. Zyablov, *Member, IEEE*

***Abstract*—Soft-decision maximum-likelihood decoding of convolutional codes using the Viterbi algorithm with a syndrome trellis is proposed. The parity check matrix of a convolutional code is used to construct the trellis. This trellis is minimal. The number of operations for the decoding of one block of a q-ary rate k/n convolutional code is $\sim nq^{\min(k,n-k)}q^{\nu}$, where ν is the memory size of the code. When the code rate satisfies $k/n > \frac{1}{2}$, the proposed algorithm is simpler than the classical Viterbi algorithm that has complexity $\sim nq^k q^{\nu}$.**

***Index Terms*—Convolutional codes, maximum-likelihood decoding, trellis decoding, complexity estimation.**

I. Introduction

In this correspondence, we consider a soft-decision maximum-likelihood (ML) decoding of linear q-ary codes. For the ML decoding of a code we use the Viterbi algorithm with some trellis representation of the code.

Definition 1: The complexity of a code trellis is the number of operations (selections and additions) required to decode a block of the code by the Viterbi algorithm with the specified code trellis.

For *block* (n, k)-codes the construction of a code trellis was proposed in [1], [2]. We call such a trellis a *syndrome trellis* because it is constructed using a parity check matrix of the code. The trellis is also known as the Wolf trellis. The complexity of the syndrome trellis is $\sim nq^{\min(k,n-k)}$. The complexity decreases as the code rate increases for rate $r = k/n > \frac{1}{2}$.

Consider a rate $R = k/n$ *convolutional* code with the memory size ν. The *memory size* of a code is the number of memory elements in the minimal encoder of the code. The complexity of the classical Viterbi trellis [3] is $\sim nq^k q^{\nu}$. Note that the complexity grows when R grows, even for $R > \frac{1}{2}$.

For soft-decision ML decoding of a convolutional code, we propose to use the Viterbi algorithm wit h a syndrome trellis of the convolutional code. The syndrome trellis of a convolutional code is also constructed using a parity check matrix of the code. We show that the complexity of the syndrome trellis of the convolutional code is $\sim nq^{\min(k,n-k)}q^{\nu}$, so the complexity of the syndrome trellis decreases when R grows, $R > \frac{1}{2}$. When $R < \frac{1}{2}$, the complexity of the syndrome trellis is approximately the same as that of the Viterbi trellis.

Manuscript received April 21, 1993; revised December 6, 1993. This paper was presented in part at the Sixth Joint Swedish–Russian International Workshop on Information Theory, Molle, Sweden, August 1993.

The authors are with the Institute for Problems of Information Transmission, Russian Academy of Sciences, 19 Ermolova Street, 101447, Moscow GSP-4, Russia.

IEEE Log Number 9405280.

The proposed syndrome trellis is minimal [4] for a given code and does not depend (up to isomorphism) on the particular parity check matrix of the code. The algorithm proposed does not depend on what particular encoder is used for the given code, so, for example, one can use a systematic encoder.

Consider not only a given convolutional code, but all equivalent codes obtained by applying the same permutations of code symbols inside each n-block. The complexity of the syndrome trellis depends on these permutations. The syndrome trellis can be optimized by considering all $n!$ permutations of the columns of the parity check matrix. For convolutional codes it is not usually complex, because n is small.

The trellises with small complexity for partial unit memory (PUM) and UM convolutional codes were suggested in [5]. For a PUM code or a UM code, the complexity of the proposed syndrome trellis coincides with the complexity of the trellis [5]. Recently, in [6], the method [5] was applied to construct a code trellis of an arbitrary convolutional code, since any such code may be considered as a UM code. It follows from [6] that for an arbitrary convolutional code, the trellis [5], [6] is similar to the proposed syndrome trellis. Nevertheless, the trellis construction methods proposed here and in [5], [6] are quite different. The approach proposed in this paper seems to be more natural and understandable, but for implementation one can use either method.

A hard-decision decoding algorithm for high-rate convolutional codes was also suggested in [7]. In contrast to that algorithm [7], we do not calculate a syndrome sequence. Rather, we estimate a *transmitted codeword* using the Viterbi algorithm with a syndrome trellis. The trellis does not depend on the received sequence and may be constructed using only the parity check matrix of the convolutional code.

II. Definitions

Let a q-ary convolutional rate k/n code C be defined by the $k \times n$ polynomial generator matrix $G(D) = [g_{ij}(D)]$, where $g_{ij}(D)$ is a polynomial over an alphabet $\mathrm{GF}(q)$. Given the polynomial generator matrix $G(D)$, the memory size of the encoder is the sum

$$\nu_G = \sum_{i=1}^{k} \max_{j} \,[\deg g_{ij}(D)].$$

The same code C can be defined by different generator matrices $G(D)$. For a given code C we consider a generator matrix $G(D)$ (the minimal encoder), which has the minimum memory size ν_G. The code C can also be defined by the $(n-k) \times n$ parity check matrix $H(D) = [h_{ij}(D)]$, which satisfies $G(D)H^T(D) = 0$. The memory size of the dual code $C^{\perp}$ is ν_H:

$$\nu_H = \sum_{i=1}^{n-k} \max_{j} \,[\deg h_{ij}(D)]. \tag{1}$$

Forney [8] showed that one can choose the parity check matrix H, such that $\nu_H = \nu_G = \nu$.

Let

$$m_{ij} = \deg h_{ij}(D),$$

$$h_{ij}(D) = h_{ij}^{(0)} + h_{ij}^{(1)}D + h_{ij}^{(m_{ij})}D^{m_{ij}}, \tag{2}$$

$$m = \max_{i,j} m_{ij},$$

$$H_l = |h_{ij}^{(l)}|, \qquad l = 0, 1, \cdots, m. \tag{3}$$

The parity check matrix $H(D)$ can be represented as a semi-infinite matrix H over $\mathrm{GF}(q)$. The matrix H consists of $(n-k) \times n$ matrices H_l. For example, for $m = 2$ the matrix H is as follows:

$$H = \left| \begin{array}{llll} H_0 & & & \\ H_1 & H_0 & & \\ H_2 & H_1 & H_0 & \\ & H_2 & H_1 & \cdots \\ & & H_2 & \cdots \end{array} \right|.$$

The matrix H_0 is nonsingular with $\operatorname{rank}(H_0) = n - k$. Let the columns of the parity check matrix H be denoted by $\boldsymbol{h}_i$, therefore $H = \|\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots\|$. A codeword $\boldsymbol{c}$ of the code C is any sequence which satisfies the relation $\boldsymbol{c}H^T = 0$, that is,

$$c_1\boldsymbol{h}_1 + c_2\boldsymbol{h}_2 + \cdots = 0. \tag{4}$$

III. The Syndrome Trellis

A trellis is a directed graph. Nodes of the graph are decomposed into a union of disjoint subsets that are called levels. The levels are indexed by $0, 1, \cdots$. The zeroth level consists of a single start node. The nodes of adjacent levels can be connected by a directed branch, labeled by a symbol of the alphabet. A path is a sequence of branches connecting the nodes of levels $0, 1, 2, \cdots$. Each path has a corresponding sequence of branch labels $\boldsymbol{x} = (x_1, x_2, \cdots)$. The trellis is called a code trellis of the code C if there is a one-to-one mapping between the codewords of the code C and the paths $\boldsymbol{x}$ of the trellis.

The syndrome trellis is constructed as follows [1], [2]. For each codeword $\boldsymbol{c}$ that satisfies (4), we construct the path of the trellis, which passes at level l, $l = 1, 2, \cdots$ through the node $s_l(\boldsymbol{c})$:

$$s_l(\boldsymbol{c}) = c_1\boldsymbol{h}_1 + c_2\boldsymbol{h}_2 + \cdots + c_l\boldsymbol{h}_l. \tag{5}$$

We assume that the nodes of a level l are indexed by q-ary vectors $s_l(\boldsymbol{c})$. The syndrome trellis can be constructed by connecting the full syndrome subtrellises [5] of the parity check matrix

$$\tilde{H} = \left| \begin{array}{c} H_0 \\ \vdots \\ H_m \end{array} \right|.$$

The full syndrome subtrellis for an $r \times n$ matrix $\tilde{H} = \|\tilde{h}_1, \tilde{h}_2, \cdots, \tilde{h}_n\|$ can be constructed as follows. Each of $n + 1$ levels of the trellis has q^r nodes indexed by q-ary r-dimensional vectors $\boldsymbol{s}$. Each node $\boldsymbol{s}$ of level $l - 1$ is connected by directed branches with q nodes $\boldsymbol{s} + \alpha\tilde{h}_l$, $\alpha \in \mathrm{GF}(q)$, of level l, $l = 1, 2, \cdots, n$. The connection of subtrellises is accomplished to satisfy the relation (5). Note that branches that are not used in a full subtrellis can be thrown out.

As an example, consider the $k/n = 2/3$ code, defined by the generator matrix

$$G(D) = \left| \begin{array}{ccc} 1 + D & D & 1 + D \\ 1 & 1 & D \end{array} \right|.$$

The parity check matrix of the code is

$$H(D) = |1 + D + D^2 \quad 1 + D^2 \quad 1|,$$

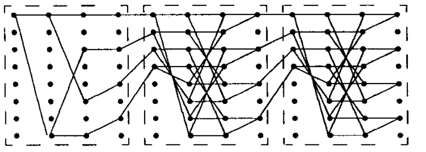

Fig. 1. Syndrome trellis diagram for rate 2/3 code.

or in ordinary form,

$$H = \left|\begin{array}{ccccccccccccccc} 1 & 1 & 1 & & & & & & & & & & & & \\ 1 & 0 & 0 & 1 & 1 & 1 & & & & & & & & & \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & & & & & & \\ & & & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & & & \\ & & & & & & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ & & & & & & & & & \cdots & & & & & \end{array}\right|.$$

The memory size of the code is $\nu = 2$. The syndrome trellis of the code is shown in Fig. 1. The horizontal lines are labeled by symbol 0, all the others by 1. The lines that show the connections of the subtrellises are not labeled. The subtrellis is formed using the parity check matrix

$$\tilde{H} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{vmatrix}.$$

IV. Estimation of the Complexity of Syndrome Trellis Decoding

Let N_l be the number of nodes of the lth level of the syndrome trellis.

Lemma 1: $N_l \leq q^{n-k+\nu}$.

Proof: Suppose a codeword $\boldsymbol{c}$ reaches level l of the trellis at node $s_l(\boldsymbol{c})$ given by (5). From (4) it follows that nonzero components of vector $s_l(\boldsymbol{c})$ are linear combinations of columns of the following matrix:

$$H^* = \left|\begin{array}{ccccc} H_m & \cdots & H_2 & H_1 & H_0 \\ & H_m & \cdots & H_2 & H_1 \\ & & H_m & \cdots & H_2 \\ & & & H_m & \cdots \\ & & & & H_m \end{array}\right|.$$

We show that $\operatorname{rank}(H^*) \leq (n-k) + \nu$. Let $\nu_i = \max_j m_{ij}$, $i = 1, 2, \cdots, n-k$. It follows from (2) and (3) that the ith row in matrices $H_{\nu+1}, H_{\nu+2}, \cdots, H_m$ is the zero vector. So the number of nonzero rows in matrix H^* is no more than $\sum_i \nu_i + (n-k) = \nu + (n-k)$ from (1), and hence $\operatorname{rank}(H^*) \leq (n-k) + \nu$. □

Lemma 2: $N_l \leq q^{\nu}$ for levels $l = in$, $i = 1, 2, \cdots$.

Proof: The proof is similar to that of Lemma 1. One has to consider the matrix H^{**}, where

$$H^{**} = \left|\begin{array}{cccc} H_m & \cdots & H_2 & H_1 \\ & H_m & \cdots & H_2 \\ & & H_m & \cdots \\ & & & H_m \end{array}\right|,$$

and to show that $\operatorname{rank}(H^{**}) \leq \nu$. □

Lemma 3: The number of paths that start from a given node at level $t = in$ and go to nodes of level $t + n$ is equal to q^k.

Proof: Such a path $\boldsymbol{c}_i = (y_{in+1}, y_{in+2}, \cdots, y_{(i+1)n})$ has to satisfy the equation $\boldsymbol{c}_i H_0^T = \mathit{const}$ because of (4). The number of solutions $\boldsymbol{c}_i$ to this equation is q^k, because $\operatorname{rank}(H_0) = n - k$. □

Theorem 1: $N_l \leq q^{\min(k, n-k)+\nu}$.

Proof: From Lemma 2 we have that the number of nodes used at level $t = in$ is no more than q^{ν}. No more than q^k paths connect each node of level t with nodes of level $t + n$, by Lemma 3. Hence no more than $q^{k+\nu}$ paths connect nodes of level t and $t + n$ and for any level l, $in \leq l \leq (i+1)n$, the number of nodes used is no more than $q^{k+\nu}$. On the other hand, the number of nodes of one trellis level is no more than $q^{n-k+\nu}$, by Lemma 1. □

The precise number N_l can be calculated for a given matrix H using [9], [4]. In [4], it was shown that for a given code C the syndrome trellis is minimal, that is, consists of the minimal number of nodes. Syndrome trellises obtained from different parity check matrices of the same code are isomorphic [4]. We started with the parity check matrix of the code C that had the minimal memory size ν_H. Now it is clear that one can use an arbitrary parity check matrix of the code C to construct the minimal syndrome trellis.

We can consider not only the code C, but all equivalent codes defined by a permutation of columns of the matrix $H(D)$. The number of nodes of the syndrome trellis, in general, depends on these permutations. The minimal trellis in the class of equivalent codes can be found considering all $n!$ of such permutations.

The complexity of the syndrome trellis is no more than $(2q-1)nq^{\min(k,n-k)}2^{\nu}$ operations (additions and selections), because no more than q branches arrive at any one node. This is an upper bound on the trellis complexity. For a given matrix $H(D)$, one can calculate the precise number of operations. The complexity of the Viterbi trellis is $\sim nq^k q^{\nu}$, so the proposed syndrome trellis is simpler than the Viterbi trellis, at least when code rate $R = k/n > \frac{1}{2} + (\log_q(2q-1))/2n$.

In fact, the syndrome trellis complexity is less than the proposed upper bound. Consider, for example, the rate 2/3 code. One can see that the syndrome trellis (Fig. 1) has no more than $2^{\min(k,n-k)}2^{\nu} = 8$ states at each level. The Viterbi trellis for the same code has only $2^{\nu} = 4$ states, but with $2^k = 4$ branches entering each node. The number of operations (additions and selections) required to decode one block using the classical Viterbi trellis is 60, while the syndrome trellis requires only 44 operations. For larger values of n and k ($R > \frac{1}{2}$), this difference becomes greater.

The syndrome trellis decoding algorithm finds the ML codeword. To recover an information sequence from the codeword, one can either use the code C in systematic form or multiply the codeword by $G^{-1}(D)$.

Acknowledgment

The authors wish to thank the referees for their careful review and for helpful comments that improved this correspondence.

References

[1] J. K. Wolf, "Efficient maximum likelihood decoding of linear block codes using a trellis," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 76–80, Jan. 1978.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar. 1974.

[3] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. 13, pp. 260–269, Apr. 1967.

[4] V. Zyablov and V. Sidorenko, "Bounds on complexity of trellis decoding of linear block codes," *Probl. Inform. Transm.*, pp. 3–9, July–Sept. 1993 [in Russian].

[5] V. Zyablov and V. Sidorenko, "Soft-decision decoding of partial-unit memory codes," *Probl. Inform. Transm.*, vol. 28, no. 1, pp. 22–27, Jan.–Mar. 1992 [in Russian]; pp. 18–22, July 1992 [in English].

[6] U. Dettmar and U. Sorger, "On maximum-likelihood decoding of unit memory codes," in *Proc. 6th Joint Swedish–Russian Int. Workshop Inform. Theory* (Molle, Sweden), Aug. 1993, pp. 184–188.

[7] J. P. M. Schalkwijk, A. J. Vink, and K. A. Post, "Syndrome decoding of binary rate k/n convolutional codes," *IEEE Trans. Inform. Theory*, vol. 24, pp. 553–562, Sept. 1978.

[8] G. D. Forney, Jr., "Convolutional codes I: Algebraic structure," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 720–738, Nov. 1970.

[9] G. D. Forney, Jr., "Coset codes—part II: Binary lattices and related codes," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1152–1187, Sept. 1988.

Six New Binary Quasi-Cyclic Codes

Zhi Chen

Abstract—**Six new quasi-cyclic codes are presented, which improve the lower bounds on the minimum distance for a binary code. A local exhaustive search is used to find these codes and many other quasi-cyclic codes which attain the lower bounds.**

Index Terms—**Quasi-cyclic codes, best known binary codes, coding and codes.**

I. Introduction

As a generalization of cyclic codes, quasi-cyclic (QC) codes contain many good linear codes. Much work has been done to find good QC codes with the help of computers, and many good QC codes have been found [1]–[4]. It should be noted that an exhaustive search is intractable with the increase in the code dimensions. Gulliver and Bhargava [1]–[3] presented a nonexhaustive method based on the exhaustive method developed by Tilborg [4]. However, it is not feasible to search for codes with large code dimensions, so some other methods should be developed. In this correspondence, a local exhaustive method is used to find good binary QC codes. New QC codes which improve the lower bounds on the minimum distance for a binary linear code are presented, and many other QC codes which attain the best known lower bounds are found.

II. New Quasi-Cyclic Codes

A code is said to be quasi-cyclic (QC) if a cyclic shift of any codeword by p positions is still a codeword. Thus, a cyclic code is a QC code with $p = 1$. The block length n of a QC code is a multiple of p, i.e., $n = mp$. A subset of QC codes can be constructed from $m \times m$ circulant matrices. Let

$$G = [C_0\ C_1 \cdots C_{p-1}] \tag{1}$$

where C_i are circulant matrices, $i = 0, 1, \cdots, p-1$. A circulant matrix C is defined to be a cyclic square matrix of the form

$$C = \begin{bmatrix} c_0 & c_1 & \cdots & c_{m-1} \\ c_{m-1} & c_0 & \cdots & c_{m-2} \\ \cdots & \cdots & \cdots & \cdots \\ c_1 & c_2 & \cdots & c_0 \end{bmatrix}. \tag{2}$$

The algebra of circulant $m \times m$ matrices over GF(2) is isomorphic to the algebra of polynomials in the ring $f(x)/(x^m + 1)$ if C is mapped onto the polynomial $c(x) = c_0 + c_1 x + \cdots + c_{m-1}x^{m-1}$. Let $c_0(x), c_1(x), \cdots, c_{p-1}(x)$ be the polynomials corresponding to circulant $m \times m$ matrices $C_0, C_1, \cdots, C_{p-1}$. Seguin and Drolet [5] defined 1-generator quasi-cyclic codes. The order of a 1-generator QC code V is defined as

$$h(x) = (x^m + 1)/\gcd\{x^m + 1, c_0(x), c_1(x), \cdots, c_{p-1}(x)\} \tag{3}$$

and the dimension k of V is equal to the degree of $h(x)$. If $h(x)$ has degree m, the dimension of V is $k = m$, and (1) is a generator matrix for V. If $k < m$, a generator matrix for V can be constructed by deleting $m - k$ rows of (1). Therefore, a 1-generator QC code is a $[pm, k]$ code.

The quasi-cyclic structure of the code can be used to simplify the search. The first step is to find all polynomials of degree less than m, which are divisible by another polynomial $a(x)$ of degree $m - k$ and $\gcd(x^m + 1, a(x)) = a(x)$. The equivalent polynomials which generate the equivalent codes are eliminated. The remaining polynomials are grouped according to their weights. Let $S_i(x)$ be sets of such polynomials with weight i, $i = 1, 2, \cdots, m - 1$.

The search is initialized with r given generator polynomials $c_0(x), c_1(x), \cdots, c_{r-1}(x)$, and an initial value of minimum distance d. To obtain a QC code with $p = r + 1$, $r + 2$, or $r + 3$, one, two, or three more generator polynomials are chosen from one, two, or three sets $S_i(x)$ of polynomials, respectively. For example, to obtain QC code with $p = r + 2$ and the minimum distance $> d$, two polynomials $c_r(x)$ and $c_{r+1}(x)$ must be chosen from two sets of polynomials $S_t(x)$ and $S_q(x)$, respectively, where

$$wt(c_0(x)) + wt(c_1(x)) + \cdots + wt(c_{r-1}(x)) + t + q > d.$$

Only the polynomials in the chosen sets are examined exhaustively. For each possible choice, the program produces its codewords one by one and checks the weights of the produced codewords. If a nonzero codeword with weight less than or equal to d is found, the program continues to examine another choice of polynomials. If no nonzero codewords with weights less than or equal to d are found, a QC code with the minimum distance $> d$ is constructed, and the program records the new code and updates the minimum distance d. This process is repeated until all possible polynomials in the given sets are investigated.

With this local exhaustive search, many good QC codes have been obtained. Among these, six QC codes improve the lower bounds on the minimum distance for a binary linear code, and 19 entries in the table of [6] are thus updated. Table I lists these codes and their generator polynomials in octal, with the least

Manuscript received November 8, 1993; revised January 26, 1994. This paper was presented in part at ISIT '94, Norway, 1994.

The author was with the Department of Electrical Engineering, Linköping University, Sweden. He is now in the Technical Department, University of Kristianstad, 281 38 Hässleholm, Sweden.

IEEE Log Number 9404936.

0018-9448/94$04.00 © 1994 IEEE