NEW MULTI-LEVEL CODES OVER GF(q)*

Jiantian Wu
Daniel J. Costello, Jr.
Department of Electrical and Computer Engineering
University of Notre Dame
Notre Dame, IN 46556

Draft
July, 1990

Abstract

In this paper, we apply set partitioning to multi-dimensional signal spaces over $GF(q)$, particularly $GF^{q-1}(q)$ and $GF^{q}(q)$, and show how to construct both multi-level block codes and multi-level trellis codes over $GF(q)$. We present two classes of multi-level (n, k, d) block codes over $GF(q)$ with block length n, number of information symbols k, and minimum distance $d_{\min} \geq d$, where $n = n_1 n_2, k = n - \sum_{i=0}^{n_1-1} \min\left\{\lceil \frac{d}{i+1} \rceil - 1, n_2\right\}$, $n_1 = q-1$ or q, $n_2 = q-1, q$, or $q+1$, and $\lceil x \rceil$ is the smallest integer larger than or equal to x. These two classes of codes use Reed-Solomon codes as component codes. They can be easily decoded as block length $q-1$ Reed-Solomon codes or block length q or $q+1$ extended Reed-Solomon codes using multi-stage decoding. Many of these codes have larger distances than comparable q-ary BCH codes. Longer block codes can be constructed by using q-ary BCH codes, or other q-ary block codes, as component codes. Low rate q-ary convolutional codes, word error-correcting convolutional codes, and binary-to-q-ary convolutional codes can also be used to construct multi-level trellis codes over $GF(q)$ or binary-to-q-ary trellis codes, some of which have better performance than the above block codes. All of the new codes have simple decoding algorithms based on hard decision multi-stage decoding.

*This work was supported by NASA Grant NAG5-557 and NSF Grant NCR89-03429.

1 Introduction

In this paper, we combine multi-level coding with set partitioning of multi-dimensional signal spaces to construct several new classes of block and trellis codes over $GF(q)$. Many of these codes have a better trade-off of minimum distance, information rate, and decoding complexity than previously known q-ary codes. A simple, fast decoding algorithm based on hard decision multi-stage decoding is also presented.

The technique of multi-level coding has been introduced in several recent papers [1-6, 10, 11]. Most researchers have considered the case where the signals are points in an N-dimensional Euclidean space and the codes are designed to maximize the minimum Euclidean distance. Binary block or convolutional codes which maximize the minimum Hamming distance have been used as component codes to construct multi-level codes based upon binary (two-way) set partition chains. However, little work has been done on multi-level codes based upon q-way ($q > 2$) set partition chains, which require the use of q-ary codes as component codes. In the following sections, we will apply set partitioning to multi-dimensional signal spaces over $GF(q)$, particularly $GF^{q-1}(q)$ and $GF^{q}(q)$, and show how to construct both multi-level block codes and multi-level trellis codes over $GF(q)$. These new codes use q-ary block and convolutional codes as component codes.

In Section 2, we construct two q-way set partition chains for $GF^{q-1}(q)$ and $GF^{q}(q)$ by using Reed-Solomon codes and shortened extended Reed-Solomon codes, respectively. Based on these set partition chains, in Section 3 we construct two classes of multi-level (n, k, d) block codes over $GF(q)$ with $n = n_1 n_2$ and $k = n - \sum_{i=0}^{n_1-1} \min\left\{\lceil \frac{d}{i+1} \rceil - 1, n_2\right\}$, where $n_1 = q - 1$ or q, $n_2 = q - 1, q$, or $q + 1$, and $\lceil x \rceil$ is the smallest integer larger than or equal to x. (Throughout the paper, an (n, k, d) block code means that the code has block length n, number of information symbols k, and design distance d, which may be less than the minimum distance $d_{\min}$ of the code.) These two classes of codes use Reed-Solomon codes as component codes and have the following advantages over q-ary BCH codes and Reed-Solomon codes:

1. Block lengths of order q^2 can be achieved, as opposed to block lengths of order q using Reed-Solomon codes.

2. For the same Hamming distance, many of these codes have higher information rates than q-ary BCH codes.

3. Since these codes have a multi-level structure, they can be simply decoded using hard decision multi-stage decoding of the component Reed-Solomon codes.

In Section 4, we use q-ary convolutional codes as component codes to obtain a class of q-ary trellis codes with higher information rates than the two above classes of block codes. In Section 5, we present another class of codes, binary-to-q-ary trellis codes, which provide more trade-offs between information rate and decoding complexity for the same minimum distance.

Although this study of multi-level codes over $GF(q)$ is motivated by the problem of finding multi-level codes based on higher-way set partition chains for QAM and PSK signal constellations, the new codes are interesting in their own right and can be used to correct both random errors and burst errors if the channel symbols are elements in a subfield of $GF(q)$.

2 The Set Partition Chain of $GF^{q-1}(q)$ and $GF^{q}(q)$

The new codes use a multi-level construction based on set partition chains of the multi-dimensional signal spaces $GF^{q-1}(q)$ and $GF^{q}(q)$. The purpose of this section is to construct these two set partition chains. In the following, we use both polynomials and vectors to represent codewords, i.e., the polynomial representation of the codeword $(c_0, c_1, \ldots, c_{n-1})$ is $c_0 + c_1 x + \cdots + c_{n-1} x^{n-1}$.

For simplicity, and without loss of generality, suppose the generator polynomial of the $(n_1, n_1 - i, i+1)$ Reed-Solomon code over $GF(q)$, donated by $RS(i)$, is

$$g_i(x) = (x-1)(x-\partial)\ldots(x-\partial^{i-1}), \quad i = 1, 2, \ldots, q-1, \tag{1}$$

where n_1 is equal to $q-1$ and ∂ is a primitive element of $GF(q)$. In particular, let $RS(0) = GF^{q-1}(q)$. Also, let the minimum distance of a single point in the set $GF^{q-1}(q)$ (a single codeword in $RS(0)$) be ∞. Next define

$$P_i(x) \triangleq \frac{(x-1)(x-\partial)\ldots(x-\partial^{i-1})}{(\partial^i-1)(\partial^i-\partial)(\partial^i-\partial^{i-1})}, \quad i = 1, 2, \ldots, q-1, \tag{2}$$

and

$$P_0(x) \triangleq 1. \tag{3}$$

Lemma 1 *For $i = 0, 1, \ldots, q-1$, if $C_i(x)$ is a code polynomial in $RS(i)$, i.e., $C_i(x) \in RS(i)$, then $C_i(x) - P_i(x)C_i(\partial^i) \in RS(i+1)$.*

This can be easily proved by showing that $1, \partial, \ldots, \partial^i$ are roots of $C_i(x) - P_i(x)C_i(\partial^i)$, i.e., $g_{i+1}(x) | C_i(x) - P_i(x)C_i(\partial^i)$. The next lemma follows directly from Lemma 1.

Lemma 2 *For any $i = 0, 1, \ldots, q-1$, and for any arbitrary $C_i(x) \in RS(i), C_i(x)$ can be uniquely expressed as*

$$C_i(x) = P_i(x)C_i(\partial^i) + C_{i+1}(x), \tag{4}$$

where $C_{i+1}(x) \in RS(i+1)$. In other words, $P_i(x)y, \ y \in GF(q)$, generates q coset representatives of $RS(i+1)$ in $RS(i)$.

From the above two lemmas, we have

Theorem 1 *$GF^{(q-1)}(q) = RS(0)/RS(1)/\ldots/RS(q-2)/RS(q-1) = \{0\}$ is a set partition chain with Hamming distances $1/2/\ldots/q-1/\infty$.*

Multi-level codes or other coset codes based on higher dimensional signal sets can sometimes achieve larger coding gains than codes based on lower dimensional signal sets. This motivates us to also construct a set partition chain for $GF^q(q)$, corresponding to shortened extended Reed-Solomon codes. The extended Reed-Solomon codes have two more information symbols than the Reed-Solomon codes while maintaining the same minimum distance and number of redundant symbols [12]. But we cannot construct a set partition chain for $GF^{q+1}(q)$. To obtain a set partition chain for $GF^q(q)$, we use shortened extended Reed-Solomon codes, which are obtained by dropping the last symbol of extended Reed-Solomon codes. These codes can be defined as follows.

Definition 1 *Let d be an arbitrary integer. A shortened extended Reed-Solomon code is a linear code over $GF(q)$ of block length $n = q$ whose codewords $(c_-, c_0, c_1, \ldots, c_{q-2})$ have the following properties:*

1. *$(c_0, c_1, \ldots, c_{q-2})$ is a codeword of a $(q-1, q-d+1, d-1)$ Reed-Solomon code with generator polynomial*

$$G(x) = (x-1)(x-\partial^1)\cdots(x-\partial^{d-3}) \quad (d \geq 3), \tag{5}$$

where ∂ is a primitive element of $GF(q)$ and $G(x) = 1$ for $d = 2$;

2.

$$c_- = c_0 + c_1\partial^{-1} + c_2\partial^{-2} + \ldots + c_{q-2}\partial^{-(q-2)}. \tag{6}$$

Again without loss of generality, let $RS'(0) = GF^q(q)$ and $RS'(i)$ be a shortened extended Reed-Solomon code with $d = i+1$, for $i = 1, 2, \ldots, q-1$. The next lemmas are similar to Lemmas 1 and 2 and follow directly from Blahut [12].

Lemma 3 *The minimum Hamming distance of $RS'(i)$ is $i+1$, for $i = 0, 1, 2, \ldots, q-1$, and $RS'(i) \supset RS'(i+1), i = 0, 1, \ldots, q-2$.*

Lemma 4 *1. For any arbitrary codeword $C'_0 = (c'_-, c'_0, \ldots, c'_{q-2}) \in RS'(0), C'_0$ can be uniquely expressed as*

$$C'_0 = (c'_- - \sum_{j=0}^{q-2} c'_j \partial^{-j}, 0, 0, \ldots, 0) + C'_1, \tag{7}$$

where $C'_1 \in RS'(1)$ and $\partial^{-j} = \partial^{q-1-j}$. That is, $(y, 0, 0, \cdots, 0)$, $y \epsilon GF(q)$, generates q coset representatives of C'_1 in C'_0.

2. For any arbitrary codeword $C'_i \in RS'(i)$, $i = 1, 2, \ldots, q-1$, C'_i can be uniquely expressed as

$$C'_i = \left(\sum_{j=1}^{i-1} p_j^{(i)} \partial^{-j} y, p_0^{(i)} y, p_1^{(i)} y, \cdots, p_{i-1}^{(i)} y, 0, 0, \cdots, 0 \right) + C'_{i+1} \tag{8}$$

where $C'_{i+1} \in RS'(i+1)$, $C_{i-1} \in RS(i-1)$, $C'_i = (c_-, C_i)$, $y = C_{i-1}(\partial^{i-1})$, and the $p_j^{(i)}(j = 0, 1, \ldots, i-1)$ are the coefficients of

$$P_{i-1}(x) = p_0^{(i)} x + \ldots + p_{i-1}^{(i)} x^{i-1}. \tag{9}$$

That is, $(\sum_{j=0}^{i-1} p_j^{(i)} \partial^{-j} y, p_0^{(i)} y, p_1^{(i)} y, \cdots, p_{i-1}^{(i)} y, 0, 0, \cdots, 0)$, $y \epsilon GF(q)$, generates q coset representatives of C'_{i+1} in C'_i, $i = 1, 2, \cdots, q-1$.

These two lemmas lead to

Theorem 2 *$GF^q(q) = RS'(0)/RS'(1)/\ldots/RS'(q-1)/RS'(q) = \{0\}$ is a set partition chain with Hamming distances $1/2/\ldots/q/\infty$.*

3 Constructions of Block Codes Over $GF(q)$

The general structure of multi-level codes has been described in many references [1-6, 10, 11]. Here we briefly discuss the principle of encoding for multi-level codes based on a q-way set partition chain.

Suppose H_0 is a signal set in a multi-dimensional space over $GF(q)$ and it generates a group under some operation, for example, addition in $GF(q)$. For $i = 1, 2, \ldots, m$, H_i is a subgroup of H_{i-1}, and H_m contains a single element of the space. The coset representative of H_{i-1} in H_i is denoted by $[H_{i-1}/H_i]$, and the number of cosets is $|H_{i-1}/H_i| = q$, for $i = 1, 2, \ldots, m$. From the theory of basic algebra, H_0 can be expressed as

$$H_0 = \sum_{i=1}^{m} [H_{i-1}/H_i]. \tag{10}$$

Thus we have a partition chain $H_0/H_1/\ldots/H_m$ with distances $\Delta_0/\Delta_1/\ldots/\Delta_{m-1}/\infty$, where Δ_i is the minimum subset distance of H_i under the distance metric in H_0, i.e., Hamming distance.

Figure 1 shows the structure of an encoder for a multi-level code based on the set partition chain $H_0/H_1/\ldots/H_m = \{0\}$, where E_i is the encoder corresponding to code C_i with information rate R_i and minimum free Hamming distance d_i, $i = 0, 1, \ldots, m-1$. The coding procedure is as follows: First, the information sequence is partitioned into m component information sequences having rates $R_0, R_1, \ldots, R_{m-1}$, $(0 \leq R_i \leq 1, \quad i = 0, 1, \ldots, m-1)$. The i^{th} component information sequence enters encoder E_i, for $i = 0, 1, \ldots, m-1$. In principle, code C_i may be any kind of code with output symbols over $GF(q)$. Each output symbol of E_i selects a coset of H_i/H_{i+1}. In this section, we only discuss the case where every component code is a block code. In the following two sections, we will show how to improve the information rate by using convolutional codes as component codes.

Suppose C_i is a block code with block length n_2, for $i = 0, 1, \ldots, m-1$. Let $I_i(x)$ be an information polynomial of encoder E_i and $Y_i(x)$ be a code polynomial in C_i, $i = 0, 1, \ldots, m-1$. For $i = 0, 1, \ldots, m-1$, $Y_i(x)$ can be expressed as

$$Y_i(x) = y_0^{(i)} + y_1^{(i)}x + \ldots + y_{n_2-1}^{(i)}x^{n_2-1} \tag{11}$$

For fixed j, each $y_j^{(i)}$ specifies a coset of H_{i+1} in H_i, for $i = 0, 1, \ldots, m-1$. By (10), an unique point in H_0, denoted by S_j, $j = 0, 1, \ldots, n_2 - 1$, is specified by $y_j^{(i)}$ $(i = 0, 1, \ldots, m-1)$. So the multi-level block code is the set of output signals

$$\mathbf{S} = \{(S_0, S_1, \ldots, S_{n_2-1}) : Y_i(x) \in C_i, \;\; i = 0, 1, \ldots, m-1\}, \tag{12}$$

where $S_i \in H_0, \;\; i = 0, 1, \ldots, n_2 - 1$. From references [2-5, 11], a lower bound exists on the minimum distance of the code:

$$d_{\min} \geq \underline{d} \triangleq \min\{\Delta_i d_i, \;\; 0 \leq i \leq m-1\}. \tag{13}$$

Therefore, to construct a code, we must choose a suitable set partition chain and a set of component codes. The two classes of multi-dimensional signals introduced in Section 2 provide good choices for set partition chains, and Reed-Solomon codes, extended Reed-Solomon codes, and shortened extended Reed-Solomon codes can be used as component codes. Now we discuss these codes in detail.

Construction A. This class of codes is based on the set partition chain $RS(0)/RS(1)/\ldots/RS(q-2)/\{0\}$. To construct a code with design distance d ($d \leq q-1$), we use component codes $C_i = RS(\lceil \frac{d}{i+1} \rceil - 1)$, for $i = 0, 1, \ldots, d-2$, and $C_i = RS(0)$, for $d-1 \leq i \leq q-2$, where $\lceil x \rceil$ is the smallest integer larger than or equal to x. According to (13), this code has minimum distance $d_{\min} \geq \underline{d} \geq d$. A codeword in this code can be expressed as follows:

$$\begin{aligned} Y_i(x) &= I_i(x) g_{\lceil \frac{d}{i+1} \rceil - 1}(x) \\ &= y_0^{(i)} + y_1^{(i)} x + \ldots + y_{q-2}^{(i)} x^{q-2}, \text{ for } i = 0, 1, \ldots, d-2, \end{aligned} \tag{14}$$

$$\text{and } Y_i(x) = I_i(x) = y_0^{(i)} + y_1^{(i)} x + \cdots + y_{q-2}^{(i)} x^{q-2}, \text{ for } i = d-1, \ldots, q-2, \tag{15}$$

where $y_j^{(i)} \in GF(q)$, for $i = 0, 1, \ldots, q-2, \;\; j = 0, 1, \ldots, q-2$. It follows from Lemma 2 that the j^{th} $(q-1)$-tuple in a multi-level codeword can be written as

$$S^{(j)}(x) \triangleq \sum_{i=0}^{q-2} P_i(x) y_j^{(i)}. \tag{16}$$

Thus, a $(q-1)^2$-tuple codeword can be expressed as

$$S(x) \triangleq \sum_{j=0}^{q-2} x^{j(q-1)} S^{(j)}(x)$$

$$= \sum_{j=0}^{q-2}\sum_{i=0}^{q-2} x^{j(q-1)} P_i(x) y_j^{(i)}. \tag{17}$$

In this case, $n = (q-1)^2$, $k = n - \sum_{i=1}^{d-1} \lceil \frac{d}{i} \rceil - (d-1)$, and the minimum Hamming distance $d_{\min} \geq d$.

To construct codes with design distance d $(q-1 < d \leq (q-1)^2)$, let $C_i = \{0\}$, i.e., $k_i = 0$, for $i < \lceil \frac{d}{q-1} \rceil - 1$, and $C_i = RS\left(\lceil \frac{d}{i+1} \rceil - 1\right)$, for $i \geq \lceil \frac{d}{q-1} \rceil - 1$. In this case, $Y_i(x) = 0$ for $i < \lceil \frac{d}{q-1} \rceil - 1$, and $Y_i(x)$ can be written as in (14) for $i \geq \lceil \frac{d}{q-1} \rceil - 1$. The number of information symbols in a codeword is

$$k = \sum_{i=\lceil \frac{d}{q-1} \rceil - 1}^{q-2} \left[(q-1) - (\lceil \frac{d}{i+1} \rceil - 1) \right] = q(q - \lceil \frac{d}{q-1} \rceil) - \sum_{i=\lceil \frac{d}{q-1} \rceil - 1}^{q-2} \lceil \frac{d}{i+1} \rceil. \tag{18}$$

Other block codes over $GF(q)$, such as extended Reed-Solomon codes, shortened extended Reed-Solomon codes, and BCH codes over $GF(q)$ can also be used as component codes. In general, if a code polynomial in C_i is

$$Y_i(x) = y_0^{(i)} + y_1^{(i)} x + \cdots + y_{n_2-1}^{(i)} x^{n_2-1}, \quad \text{for } i = 0, 1, \ldots, q-2, \tag{19}$$

then a codeword in the multi-level code can be expressed as

$$S(x) = \sum_{j=0}^{n_2-1}\sum_{i=0}^{q-2} x^{j(q-1)} P_i(x) y_j^{(i)}. \tag{20}$$

If we take only Reed-Solomon codes, shortened extended Reed-Solomon codes, and extended Reed-Solomon codes as component codes, we obtain block codes over $GF(q)$ having the following parameters:

$$\text{block length } n = (q-1)^2, \;\; q(q-1), \text{ or } q^2 - 1,$$

number of information symbols

$$k = n - \sum_{i=1}^{d-1} \lceil \frac{d}{i} \rceil - (d-1), \quad (d \leq n/(q-1))$$

or

$$k = (\frac{n}{q-1} + 1)\left(q - \lceil \frac{d(q-1)}{n} \rceil\right) - \sum_{i=\lceil \frac{d(q-1)}{n} \rceil}^{(q-1)} \lceil \frac{d}{i} \rceil, \quad (d > n/(q-1))$$

and minimum Hamming distance $d_{\min} \geq d \ \ (d \leq n)$.

Construction B. This class of codes is based on the partition chain $RS'(0)/RS'(1)/\ldots/RS'(q-1)/\{0\}$. Similar to Construction A, a code polynomial in component code C_i can be expressed as

$$Y_i(x) = y_0^{(i)} + y_1^{(i)} x + \cdots + y_{n_2-1}^{(i)} x^{n_2-1} \quad \text{for } i = 0, 1, \ldots, q-1. \tag{21}$$

From Lemma 4, the j^{th} q-tuple in a multi-level codeword can be written as

$$S^{(j)}(x) \triangleq y_j^{(0)} + \sum_{i=1}^{q-1} \sum_{k=1}^{i-1} p_k^{(i)} \partial^{-k} y_j^{(i)} + x \sum_{i=1}^{q-1} P_{i-1}(x) y_j^{(i)}. \tag{22}$$

Thus a codeword in the multi-level code can be expressed as

$$S(x) = \sum_{j=0}^{n_2-1} x^{jq} \left[y_j^{(0)} + \sum_{i=1}^{q-1} \sum_{k=o}^{i-1} p_k^{(i)} y_j^{(i)} \partial^{-k} + x \sum_{i=1}^{q-1} P_{i-1}(x) y_j^{(i)} \right]. \tag{23}$$

Using Reed-Solomon codes, shortened extended Reed-Solomon codes, and extended Reed-Solomon codes as component codes, we can obtain block codes over $GF(q)$ having the following parameters:

block length $n = q(q-1), q^2$, or $q(q+1)$

number of information symbols $k = n - \sum_{i=1}^{d-1} \lceil \frac{d}{i} \rceil - (d-1), \quad (d \leq n/q)$

or

$$k = (\frac{n}{q} + 1)\,(q - \lceil \frac{dq}{n} \rceil + 1) - \sum_{i=\lceil \frac{dq}{n} \rceil}^{q} \lceil \frac{d}{i} \rceil, \quad (d > n/q)$$

and minimum Hamming distance $d_{\min} \geq d \ (d \leq n)$.

In summary, we have constructed two classes of block codes over $GF(q)$ having the following parameters:

a) block length $n = n_1 n_2$, where $n_1 = q-1$ for construction A and q for construction B, and $n_2 = q-1, q,$ or $q+1$, which is the block length of Reed-Solomon codes, shortened extended Reed-Solomon codes, and extended Reed-Solomon codes, respectively.

b) number of information symbols $k = n - \sum_{i=0}^{n_1-1} \min\left\{\lceil \frac{d}{i+1} \rceil - 1, n_2\right\}$.

c) minimum Hamming distance $d_{\min} \geq d$.

Table 1 shows all the codes in the above two classes over $GF(4)$ with $d_{\min} \geq 3$. Table 2 shows some construction B codes over $GF(8)$ with block length 72 and minimum distance from 3 to 15, where the redundancy of the component codes is $\rho_i = \min\left\{\lceil \frac{d}{i+1} \rceil - 1, 9\right\}, i = 0, 1, \ldots, 7$, and the total redundancy is $\rho = \sum_{i=0}^{7} \rho$.

Example 1. The (20, 9, 8) construction B code shown in Table 1 is based on the set partition chain $GF^4(4) = RS'(0)/RS'(1)/RS'(2)/RS'(3)/RS'(4) = \{0\}$ with distances $1/2/3/4/\infty$. It contains four extended Reed-Solomon codes as component codes: $C_0 = (5, 0, \infty), C_1 = (5, 2, 4), C_2 = (5, 3, 3)$, and $C_3 = (5, 4, 2)$. It has a higher information rate ($\frac{9}{20}$ vs. $\frac{6}{15}$) and a larger minimum distance (8 vs. 7) than the (15, 6, 7) BCH code over $GF(4)$.

Table 3 presents a comparison between BCH codes and the new codes over $GF(4)$. Although some codes shown in Table 1 are not as good as BCH codes, there exist many new codes better than BCH codes. Moreover, the decoding complexity of these new codes is less than the BCH codes.

Table 1. Codes over $GF(4)$ ($d \geq 3$)

$(9,6,3)^1$	$(9,4,4)^1$	$(9,3,6)^1$	$(9,1,9)^1$
$(15,12,3)^1$	$(15,10,4)^1$	$(15,8,5)^1$	$(15,7,6)^1$
$(15,5,8)^1$	$(15,4,9)^1$	$(15,3,10)^1$	$(15,2,12)^1$
$(15,1,15)^1$	$(12,9,3)^2$	$(12,7,4)^2$	$(12,5,6)^2$
$(12,3,8)^2$	$(12,2,9)^2$	$(12,1,12)^2$	$(16,13,3)^3$
$(16,11,4)^3$	$(16,8,6)^3$	$(16,6,8)^3$	$(16,4,9)^3$
$(16,3,12)^3$	$(16,1,16)^3$	$(20,17,3)^3$	$(20,15,4)^3$
$(20,12,5)^3$	$(20,11,6)^3$	$(20,9,8)^3$	$(20,7,9)^3$
$(20,6,10)^3$	$(20,5,12)^3$	$(20,3,15)^3$	$(20,2,16)^3$
$(20,1,20)^3$			

1 Construction A; 2 Construction A or B; 3 Construction B

Table 2. $(72, 72-\rho, d)$ Codes over $GF(8)$ $(3 \leq d \leq 15)$

d	3	4	5	6	7	8	9	10	11	12	13	14	15
ρ_0	2	3	4	5	6	7	8	9	9	9	9	9	9
ρ_1	1	1	2	2	3	3	4	4	5	5	6	6	6
ρ_2	0	1	1	1	2	2	2	3	3	3	4	4	4
ρ_3	0	0	1	1	1	1	2	2	2	2	3	3	3
ρ_4	0	0	0	1	1	1	1	1	2	2	2	2	2
ρ_5	0	0	0	0	1	1	1	1	1	1	2	2	2
ρ_6	0	0	0	0	0	1	1	1	1	1	1	1	2
ρ_7	0	0	0	0	0	0	1	1	1	1	1	1	1
ρ	3	5	8	10	14	16	20	22	24	24	28	28	29

Table 3. Comparison between BCH codes and the new codes over $GF(4)$

BCH [12]	(15, 11, 3)	(15, 9, 5)	(15, 6, 7)	(15, 4, 9)	(15, 3, 11)
New	(20, 17, 3)	(20, 12, 5)	(20, 9, 8)	(20, 7, 9)	(20, 5, 12)

Note that sometimes the actual lower bound distance $\underline{d}$ is larger than the design distance d. For example, in Table 2, both $d = 11$ and $d = 12$ lead to the same code ($\underline{d} = 12$). The following theorem gives the relationship between d and $\underline{d}$.

Theorem 3 *For the above two classes of codes,*

(1) for $d \leq n_1$, $\underline{d} = d$.

(2) for $d > n_1$, if there exists an $i \in I \triangleq \left\{\lceil \frac{d}{n_1} \rceil, \lceil \frac{d}{n_1} \rceil + 1, \ldots, N_2\right\}$ so that i divides d, then $d = \underline{d}$, and if for all $i \in I$, i does not divide d, then $\underline{d} > d$.

Proof: (1) This follows from the fact that $\lceil \frac{d}{1} \rceil = d$ and from (13).

(2) Suppose $i_0 \in I$ divides d. Then $\underline{d} = \lceil \frac{d}{i_0} \rceil i_0 = d$. If for all $i \in I$, i does not divide d, then $\lceil \frac{d}{i} \rceil i \geq d + 1$ for all $i \in I$. By (13), $\underline{d} \geq d + 1$. QED

Corollary 1. For $d > n_1$, if d is a prime, then the above (n, k, d) codes have a lower bound distance $\underline{d} \geq d + 1$.

4 Constructions of Trellis Codes Over $GF(q)$

The idea of applying set partitioning to trellis coding over finite fields was discussed in references [13, 14]. Convolutional codes have the advantage of achieving large free distances by increasing complexity without decreasing information rate. For this reason, with q-ary convolutional codes, we can construct q-ary multi-level trellis codes with the same minimum distance and higher information rates than the above block codes.

High rate convolutional codes over $GF(q)(q > 2)$ are difficult to find, and their decoding complexities are high because they are powers of q. Most of the work on constructing convolutional codes over $GF(q)$ has focused on low rate codes. Recently, Ryan and Wilson [15] have constructed some optimal low rate convolutional codes over $GF(q)$. In this section, we will show how to construct high rate multi-level trellis codes over $GF(q)$ by using good known low rate convolutional codes with reasonable decoding complexities as component codes. We should point out that, from the multi-level coding point of view, the finite state codes found in [13] can be viewed as one level codes based on set partitioning of multi-dimensional signal spaces over $GF(q)$.

To construct high rate multi-level trellis codes, we use low rate q-ary convolutional codes to replace some of the low rate q-ary block codes as component codes. It is not necessary to replace every block component code with a convolutional code. The idea of using a mixture of block and convolutional codes at different levels has appeared in [4, 6, 11].

Let R_i be the rate of component code C_i $(i = 0, 1, \ldots, m-1)$. Then the rate of the overall code is

$$R = \frac{1}{m} \sum_{i=0}^{m-1} R_i \qquad (24)$$

Since some component codes are convolutional codes, we will use the free Hamming distance (d_{free}) instead of the minimum Hamming distance $(d_{\min})$ over a block. The free Hamming distance of a trellis code is defined to be the minimum Hamming distance between all pairs of distinct code sequences. In this case, we have a lower bound on the free distance:

$$d_{free} \geq \underline{d} = \min \{\Delta_i d_f(i), \;\; 0 \leq i \leq m-1\} \tag{25}$$

where $d_f(i)$ is the free (or minimum) distance of the convolutional (or block) code C_i, for $i = 0, 1, \ldots, m-1$.

Example 2. We will take the (20, 9, 8) code from Table 1 and show how to improve its information rate by using convolutional codes. As shown in example 1, the component codes are $C_0 = (5, 0, \infty)$, $C_1 = (5, 2, 4)$, $C_2 = (5, 3, 3)$, and $C_3 = (5, 4, 2)$. Now replace C_0 by a rate 1/2 4-ary convolutional code with free distance 8 and 64 states and C_1 by a rate 1/2 4-ary convolutional code with free distance 6 and 16 states. Thus the trellis code has rate $R = \frac{1}{4}\left(\frac{1}{2} + \frac{1}{2} + \frac{3}{5} + \frac{4}{5}\right) = \frac{3}{5}$ and free distance 8, whereas the (20, 9, 8) block code also has minimum distance 8, but its rate is only 9/20.

From this example, we can give the following principles for constructing multi-level trellis codes with design distance d from the above block codes. If a block component code C_i has rate less than 1/2, we consider a rate 1/2 convolutional code with free distance larger than or equal to d/Δ_i instead of the block component code; otherwise, retain the block component code. If the constraint length of the rate 1/2 convolutional code is too large, implying that the decoding complexity is too great, and if the rate of the block component code C_i is less than 1/3, we can consider a rate 1/3 convolutional code with free distance larger than or equal to d/Δ_i as a candidate to replace the block component code. Other replacement codes can be found in a similar way.

Table 4 lists some codes obtained from the block length 20 codes listed in Table 1, where R_i is the rate of component code C_i, K_i is the constraint length of convolutional code C_i, and $d_f(i)$ is the free (or minimum) distance of convolutional (or block) code C_i, for $i = 0, 1, 2, 3$. The memory order of convolutional code C_i is $m_i = K_i - 1$, i.e., the number of states of code C_i is $4^{K_i - 1}$. To compare with block codes, we also use the notation (n, k, d) for the multi-level trellis codes constructed, where n is the block length of the block component codes and

$k = nR$ is not necessarily an integer. Compared to the construction B block codes listed in Table 1, the codes in Table 4 exhibit a clear improvement in information rate. For example, the block code of block length 20 and minimum Hamming distance 12 has information rate 1/4, whereas the trellis code with free Hamming distance 12 achieves an information rate of 29/60, or close to 1/2.

Table 4. Trellis codes over $GF(4)$ $(6 \leq d \leq 12)$

Trellis code	R_i	K_i	$d_f(i)$
$(20, 13\frac{1}{2}, 6)$	1/2-C	3	6
	3/5-B	–	3
	4/5-B	–	2
	4/5-B	–	2
(20, 12, 8)	1/2-C	4	8
	1/2-C	3	6
	3/5-B	–	3
	4/5-B	–	2
(20, 11, 9)	1/2-C	5	9
	1/2-C	3	6
	3/5-B	–	3
	4/5-B	–	2
$(20, 10\frac{1}{6}, 9)$	1/3-C	3	9
	1/2-C	3	6
	3/5-B	–	3
	4/5-B	–	2
$(20, 9\frac{2}{3}, 12)$	1/3-C	4	12
	1/2-C	3	6
	1/2-C	3	6
	3/5-B	–	3

Note: In the table, the letter B denotes a block code over $GF(4)$ (an extended Reed-Solomon code), and C denotes a convolutional code over $GF(4)$ from [15].

5 Constructions of Binary-to-q-ary Trellis Codes

In many practical systems, the output of the information source is binary symbols, and the channel signals can be viewed as symbols in $GF(2^n)$. Here are two examples:

1. For multi-level codes based on multi-dimensional signal constellations [10, 16-18], suppose H_i and H_{i+1} are subsets of the signal set H_0 and $H_i \supset H_{i+1}$. Then the cosets of H_i/H_{i+1} are usually isomorphic to $GF(q)$, where q is the number of cosets in H_i/H_{i+1}. So the outputs of the component encoder E_i can be viewed as elements in $GF(q)$, but the inputs of the encoder are binary symbols.

2. Piret [19] suggested a class of convolutional codes called word error-correcting codes. His word-error-correcting codes use word weight instead of Hamming weight as the distance measure. Consider an (n, k) convolutional code with k input bits and n output bits at each time interval. The n output bits are called a word, and if they are not all zeros, the word weight is 1. The word distance between any two code sequences is the word weight of the difference between these two code sequences. The minimum free word distance of a binary convolutional code is defined to be the minimum word distance between all pairs of distinct code sequences. Alternately, if we view a word as a symbol over $GF(2^n)$, an (n, k) convolutional code is actually a k-input, 1-output binary-to-2^n-ary convolutional code, i.e., it is a special class of binary-to-q-ary convolutional codes.

Ryan and Wilson [15] presented some optimal low rate binary-to-q-ary convolutional codes for $q = 4, 8$, and 16. In their paper, the rate was defined as the number of input bits divided by the number of output symbols. We call this the binary-to-q-ary rate, denoted by $R_{b,q}$. If the binary-to-q-ary rate of a k input bit, n output symbol convolutional code is k/n, then the normalized rate is defined as $R \triangleq k/(n \log_2 q)$. We will use the normalized rate to compare with the codes of the previous section.

The encoder structure of binary-to-q-ary trellis codes is still as shown in Figure 1. Many kinds of codes, including binary-to-q-ary codes, word-error-correcting codes, and codes over $GF(q)$ can be used as component codes. (Codes over $GF(q)$ can be chosen as component codes because any code over $GF(q)$ can be viewed as a binary-to-q-ary code as long as each input symbol is viewed as $\log_2 q$ bits. In this sense, trellis codes over $GF(q)$ are a special case of binary-to-q-ary trellis codes.) Because the free distance of a code depends only on the structure of the code sequences rather than on the input sequences, the inequality of (25) also holds for binary-to-q-ary trellis codes.

As in the previous section, we list codes corresponding to the block length 20 codes listed in Table 1. Note that here the number of state is $2^{K_i - 1}$ rather than $4^{K_i - 1}$. Comparing Table 4 to Table 5, one finds that the codes listed in Table 5 have less decoding complexity and a lower information rate. Therefore the codes in Table 5 offer additional trade-offs between information rate and decoding complexity.

Note that the two $(20, 13\frac{1}{2}, 6)$ codes have the same parameters in both tables, but the Table 5 code is better because the binary-to-4-ary convolutional code C_0 has a better distance distribution (fewer nearest neighbors) than the convolutional code over $GF(4)$.

Table 5. Binary-to-4-ary Trellis Codes ($6 \leq d \leq 12$)

Trellis codes	R_i	K_i	$d_f(i)$
$(20, 13\frac{1}{2}, 6)$	1/2-C	6	6
	3/5-B	–	3
	4/5-B	–	2
	4/5-B	–	2
$(20, 10\frac{3}{4}, 8)$	1/4-C	4	8
	1/2-C	4	4
	3/5-B	–	3
	4/5-B	–	2
$(20, 9\frac{1}{4}, 10)$	1/4-C	5	10
	1/2-C	5	5
	1/2-C	4	4
	3/5-B	–	3
$(20, 9\frac{1}{4}, 12)$	1/4-C	6	12
	1/2-C	6	6
	1/2-C	4	4
	3/5-B	–	3

Note: In the table, the letter B denotes a block code over $GF(4)$ (an extended Reed-Solomon code), and C denotes a binary-to-4-ary convolutional code from [15].

6 Fast Coding and Decoding

Coding and decoding schemes for multi-level codes based on two way partition chains were first presented by Imai and Hirakawa [1]. These were later generalized by Pottie and Taylor [5], Tanner [4], and Wu [11]. For simplicity, we take the construction A block codes as an example to illustrate the principles of encoding and decoding, which also apply to the other codes discussed above.

From the structure of the encoder shown in Figure 1, the encoder consists of m component encoders and a mapper. Since known encoders can be used as component encoders, the major problem of encoding is to decrease the complexity of the mapping. From (16), (17), (2), and (3), the mapping can be implemented by computing $P_i(x)$ in advance. Then the coefficients of a codeword $S(x)$ can be obtained by taking the product of $y_j^{(i)}$ and $P_i(x)$ for $i = 0, .1, \ldots, q-2$ and $j = 0, 1, \ldots, q-2$. Therefore, we must store the coefficients of $P_i(x)$ in advance, for $i = 1, 2, \ldots, q-2$ (note that $P_0(x) = 1$). Since the degree of $P_i(x)$ is i, only i symbols are required to store the coefficients of $P_i(x)$. The total number of symbols to be stored in the encoder is therefore $\frac{1}{2}(q-2)(q-1)$. From the following decoding procedure, we will see that the coefficients of $P_i(x)$ should also be stored in the decoder.

Assume that the receiver makes hard decisions and let $\tilde{S}(x)$ be a received codeword over $GF(q)$. Then the decoding procedure is as follows:

Step 1. For $j = 0, 1, \ldots, n_2 - 1$, let $\tilde{S}_1^{(j)}(x) = \tilde{S}^j(x)$ and set $\tilde{y}_j^{(0)} = \tilde{S}_1^{(j)}(1)$.
Then $\left(\tilde{y}_0^{(0)}, \tilde{y}_1^{(0)}, \ldots, \tilde{y}_{n_2-1}^{(0)}\right)$ is the decoder input for code C_0, and the output is denoted by $\left(\hat{y}_0^{(0)}, \hat{y}_1^{(0)}, \ldots, \hat{y}_{n_2-1}^{(0)}\right)$.

Step 2. $(2 \leq i \leq q-1)$ For $j = 0, 1, \ldots, n_2 - 1$, let

$$\tilde{S}_i^{(j)}(x) = \tilde{S}_{i-1}^{(j)}(x) - P_{i-2}(x)\hat{y}_j^{(i-2)} \tag{26}$$

and set

$$\tilde{y}_j^{(i-1)} = \tilde{S}_i^{(j)}(\partial^{i-1}). \tag{27}$$

Then $\left(\tilde{y}_0^{(i-1)}, \tilde{y}_1^{(i-1)}, \ldots, \tilde{y}_{n_2-1}^{(i-1)}\right)$ is the decoder input for code C_{i-1}, and the output is denoted by
$\left(\hat{y}_0^{(i-1)}, \hat{y}_1^{(i-1)}, \ldots, \hat{y}_{n_2-1}^{(i-1)}\right)$.

Finally, the estimated information sequences $\hat{I}_i(x)$ are obtained from the decoder estimates
$\left(\hat{y}_0^{(i)}, \hat{y}_1^{(i)}, \ldots, \hat{y}_{n_2}^{(i)}\right)$, for $i = 0, 1, 2, \ldots, q-2$, by applying the inverse of the encoder mapping.

As shown by Tanner [4], the above decoding procedure can achieve the lower bound distance $\underline{d}$ of (13). If the design distance $d \leq q-1$ in the above decoding procedure, only the first $d-1$ steps are needed, and for the remaining $q-d$ steps,

$$\left(\hat{y}_0^{(i-1)}, \hat{y}_1^{(i-1)}, \ldots, \hat{y}_{n_2-1}^{(i-1)}\right) = \left(\tilde{y}_0^{(i-1)}, \tilde{y}_1^{(i-1)}, \ldots, \tilde{y}_{n_2-1}^{(i-1)}\right), \quad \text{for } i = d-1, d-2, \ldots, q-1. \tag{28}$$

Also, if $d > q-1$, the first $\lceil \frac{d}{q-1} \rceil$ steps can be omitted.

From the decoding procedure, we can see that the decoding complexity is the sum of the complexity of computing (26) and (27) and the decoding complexity of each component code. The complexity of computing (26) and (27) is relatively small since the coefficients of the polynomials $P_i(x)$ are computed in advance, and in most cases the decoding complexity is dominated by the component code whose decoding complexity is the highest among all component codes.

7 Conclusions

We have applied set partitioning to multi-dimensional signal spaces over $GF(q)$ to construct powerful q-ary block and trellis codes. Many of these codes have a better trade-off of minimum distance, information rate, and decoding complexity than previously known q-ary codes. A fast decoding algorithm based on hard decision multi-stage decoding has been presented. (A decoding algorithm based on soft decisions appears to be quite complex at this time.)

Although only Reed-Solomon codes, shortened extended Reed-Solomon codes, and extended Reed-Solomon codes are used as component codes in the block code constructions, other q-ary codes, such as q-ary BCH codes, can also be used as component codes, possibly resulting in longer and better codes.

The trellis codes constructed have better performance, but more decoding complexity, than the block codes. A comparison of performance vs. decoding complexity between these new block and trellis codes will be an interesting subject for further study.

References

[1] H. Imai and S. Hirakawa, "A New Multi-Level Coding Method Using Error Correcting Codes", *IEEE Trans. on Inform. Theory*, Vol. IT-23, pp. 371-377, May 1977.

[2] V. V. Ginzburg, "Multi-Dimensional Signals for a Continuous Channel", *Problemy Peredachi Informatsii*, Vol. 20, No. 1, trans. from Russian by Plenum Publishing Co., 1984.

[3] T. Kasami and S. Lin, "On Construction of Bandwidth Efficient Block Codes", *IEEE Trans. on Inform. Theory*, to appear.

[4] R. M. Tanner, "Algebraic Construction of Large Euclidean Distance Combined Coding/Modulation Systems", *IEEE Trans. on Inform. Theory*, to appear.

[5] G. J. Pottie and D. P. Taylor, "Multi-level Codes Based on Partitioning", *IEEE Trans. on Inform. Theory*, Vol. 35, pp. 87-89, January 1989.

[6] A. R. Calderbank, "Multilevel Codes and Multistage Decoding", *IEEE Trans. on Commun.*, Vol. 37, pp. 222-229, March 1989.

[7] G. D. Forney, Jr., et al., "Efficient Modulation for Band Limited Channels", *IEEE J. Select. Areas Commun.*, Vol. SAC-2, pp. 659-686, Sept. 1984.

[8] G. D. Forney, Jr., "Coset Codes I: Introduction and Geometrical Classification", *IEEE Trans. Inform. Theory*, Vol. 34, pp. 1123-1151, September 1988.

[9] G. D. Forney, Jr., "Coset Codes II: Binary Lattices and Related Codes", *IEEE Trans. Inform. Theory*, Vol. 34, pp. 1152-1187, September 1988.

[10] J. Wu and X. Zhu, "Multi-level Multi-dimensional Trellis Codes", *1990 Int. Symp. on Inform. Theory*, San Diego, CA, Jan. 1990.

[11] J. Wu, "Efficient Coded Modulation for Improvement of Channel Utilization Ratio", Ph. D. Dissertation, (in Chinese) Tsinghua University, Dec. 1988.

[12] R. E. Blahut, Theory and Practice of Error Control Codes, Addison-Wesley Publishing Company, 1983.

[13] F. Pollara, et al, "Finite-state Codes", *IEEE Trans. on Inform. Theory*, Vol. 34, pp. 1083-1089, September 1988.

[14] J. Wu, X. Zhu and D. Lu, "Two Classes of High Rate Convolutional Codes", *J. China Inst. of Communications*, pp. 1-6, March, 1989.

[15] W. E. Ryan and S. G. Wilson, "Two Classes of Convolutional Codes over GF(q) for *q*-ary Orthogonal Signaling", *IEEE Trans. on Commun.*, to appear.

[16] G. D. Forney, Jr., "Algebraic Structure of Q-ary Lattices", in preparation.

[17] J. H. Conway and N. J. A. Sloane, Sphere Packings, Lattices, and Groups. New York: Springer-Verlag, 1988.

[18] D. J. Costello, Jr., J. Wu, and S. Lin, "Multi-level Trellis Coded Modulation and Multi-Stage Decoding", *1990 IEEE Inform. Theory Workshop*, Eindhoven, The Netherlands, June 1990.

[19] P. Piret, "Multiple-word Correcting Convolutional Codes", *IEEE Trans. on Inform. Theory*, Vol. 30, pp. 637-644, July 1984.

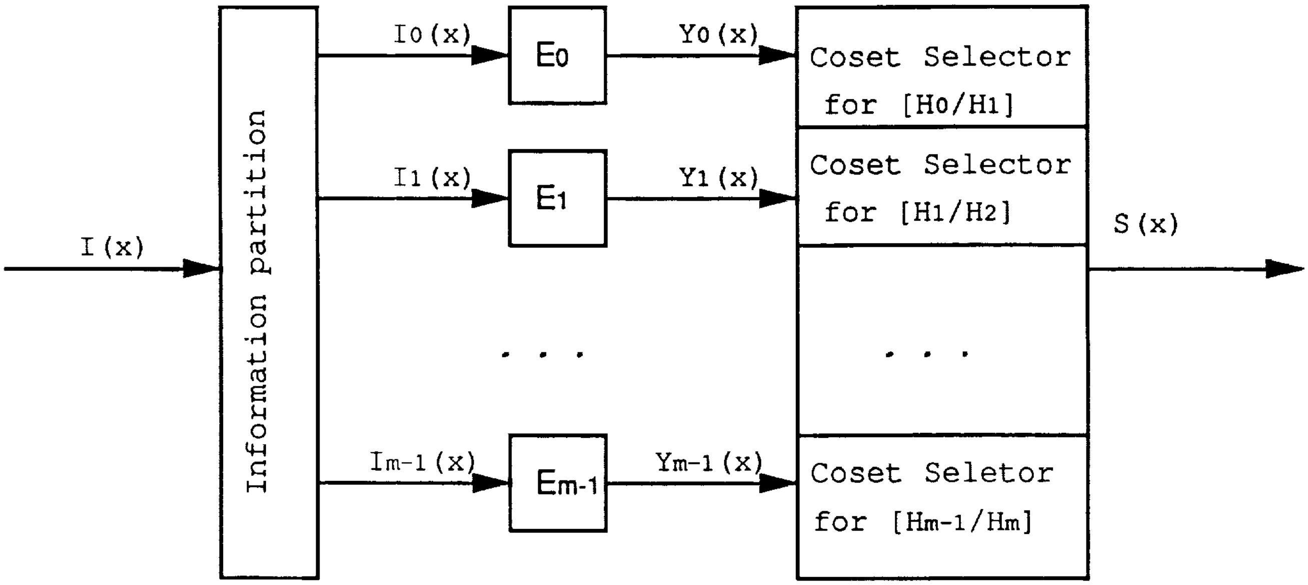
I(x)
Information partition
I0(x)
I1(x)
Im-1(x)
E0
E1
Em-1
Y0(x)
Y1(x)
Ym-1(x)
Coset Selector for [H0/H1]
Coset Selector for [H1/H2]
Coset Seletor for [Hm-1/Hm]
S(x)

Fig.1 The structure of an encoder for a multi-level code

Appendix C

A Hybrid M-Algorithm/Sequential Decoder for Convolutional and Trellis Codes