Codes Over Rings for Rayleigh Fading Channel

Shahram Yousefi[0]
Dept. of Elec. and Comp. Eng.
University of Waterloo
Waterloo, Ont., Canada N2L 3G1
shahram@shannon.uwaterloo.ca

Amir K. Khandani
Dept. of Elec. and Comp. Eng.
University of Waterloo
Waterloo, Ont., Canada N2L 3G1
khandani@shannon.uwaterloo.ca

Abstract — **In this paper, we investigate the algebraic structure of a new Block-Coded Modulation (BCM) scheme suitable for Rayleigh fading applications. The construction method is based on extending good binary block codes from GF(2) to Z_q to be used with a q-PSK constellation. We use well-known binary block codes because the Minimum Time Diversity (MTD) and Bandwidth-Efficiency (BWE) of the resulting BCM scheme are determined by the minimum Hamming distance and rate of the underlying binary code. Also a 2-level decoding method for the maximum-likelihood decoding of the code is proposed.**

I. Introduction

The primary advantage of bandwidth-efficient coded modulation schemes over modulation schemes using traditional error-correcting codes is their capability to improve the coding performance without bandwidth expansion. Such a property makes them appropriate for channels that are limited in both power and bandwidth. Coded modulation goes back to 1974, when Massey suggested the notion of improving system performance by looking at coding and modulation as a combined entity [1]. Perhaps the most remarkable contribution towards implementing Massey's thoughts was the invention of *Trellis-Coded Modulation* (TCM) by Ungerboeck [2]. Almost at the same time and independently, Imai and Hirakawa [3] proposed *Block-Coded Modulation* (BCM). The difference between the two being that in TCM one uses a convolutional code to choose a sequence of points from an expanded signal constellation while in BCM a block code is used for this purpose. It is generally believed that BCM is less power-efficient than TCM for Additive White Gaussian Noise (AWGN) channels. For fading channels, however, because of its shorter decoding depth and hence more effective interleaving[1], BCM has the potential to compete with TCM [4].

It has been established in the literature that the appropriate criterion for coded modulation design on an AWGN channel is the maximization of *Minimum Squared Euclidean Distance* (MSED) between any two valid sequences of symbols. It has been shown in [5] that the performance of coded modulation over a fading channel with interleaving/deinterleaving is dominated by: (i) the length of the shortest error event path or the so-called *Minimum Time Diversity* (MTD) between any two valid sequences of symbols; and (ii) the *Minimum Squared Product Distance* (MSPD) between any two valid sequences of symbols with the MTD. Although the literature of formulation, design and analysis of modulation schemes suitable for AWGN interference is very rich, there is considerably less effort devoted to fading channel.

In this paper, a new BCM scheme suitable for fading channel is considered. Our method is based on extending good binary block codes from GF(2) to Z_q to be used with a q-PSK constellation. We use well-known binary block codes because the MTD and BWE of the resulting BCM scheme are determined by the minimum Hamming distance and rate of the underlying binary code. The schemes fall into the category of codes over rings and groups which recently have received a lot of attention among coding theorists [6]. An appropriate graphical representation of the code is used in its analysis and decoding.

II. The structure of codes over rings

There are two important components to the structure of the proposed schemes. One is the block code used to introduce the redundancy and the other is the underlying signal constellation. For the former, we use an special extension of well-known binary block codes. We give example for the class of Reed-Muller (RM) codes extended (as it will be explained later) from GF(2) to Z_q. An rth-order Reed-Muller code of length 2^m denoted by RM(r,m) is an (n,k,d) binary block code where $(n,k,d) = (2^m, \sum_{i=0}^{r} \binom{m}{i}, 2^{m-r})$. These are the best binary block codes of length $n = 2^m$ with minimum distance 2^{m-r} for $n \leq 32$ [7].

We assume a q-PSK signal constellation where the components of the q-ary code are directly mapped to the q-PSK points using an appropriate labeling. The extension of the binary linear code to a q-ary linear code is based on extending the kernel space of the binary code (i.e., the parity-check equations or the H matrix) to $\{0,1,...,q-1\}$, mod q constraints. In this case, the encoder inputs $\log(q^k) = k \cdot \log(q)$ bits and outputs a length n codeword of elements of $Z_q = \{0,1,...,q-1\}$ which are each mapped to the points of a q-PSK constellation. The resulting scheme is $2n$-dimensional with a time diversity of TD$= d$, and bandwidth-efficiency of $\eta = k \cdot \log(q)/n = R \cdot \log(q)$ bits/2-D symbol ($R = k/n$ is the binary code rate).

The matching of Z_q and q-PSK is motivated by the notion of *matched labeling* in the sense of Loeliger [8]. Although the algebraic structure used here is the ring of integers mod q (Z_q), the reader should note that at least in this context, we will only need the additive group of the ring. In fact rings are very restrictive and only in the case of generator matrix and other issues such as trellis representation that might require a generator of some form we prefer to work on rings in

[0]This work was supported by Natural Sciences and Engineering Research Council of Canada (NSERC).

[1]An Interleaver or scrambler is an essential part of almost all digital communication systems. It performs a permutation of the transmitted symbols to break up burst errors caused by amplitude fades of duration greater than one symbol time.

which case we can use the convenient concept of linearity[2]. Therefore, on a more general algebraic side of the problem, we have a finite group $(G,*)$, the alphabet over which the code is defined. The connection between the group $(G,*)$ and the Geometrically Uniform [9] (GU) signal set $\mathcal{S}$ is made by a mapping m from G to $\mathcal{S}$ which for the sake of brevity we require to be a bijection.

Definition [8]: A group $(G,*)$ is said to be matched to a signal set $\mathcal{S} \subset \mathcal{R}^N$ in the AWGN sense if there exists a bijection m from G to $\mathcal{S}$ such that:

$$\forall g, g' \in G : d\left(m(g), m(g')\right) = d(m(g^{-1} * g'), m(e))$$

where $d(.,.)$ is the Euclidean distance function and e is the identity element of the group. The mapping m is called an AWGN *matched labeling*.

Starting from a low-dimensional GU signal set, we have the following recipe to construct high-dimensional GU signal sets: let I be an index set[3] and C be a subgroup of the Cartesian or n-fold direct product group G^I ($\mathrm{C} < G^I$), then a GU signal constellation can be obtained by mapping the elements of C component-wise, i.e.,

$$S = \mathbf{m}(\mathrm{C}) = \{s \in \mathcal{S}^I : s = m(c), c \in \mathrm{C}\}$$

The ordering of the signal points around the q-PSK constellation (labeling of points) influences the MSPD and the error coefficient of the code and thereby affects the performance.

For a Rayleigh fading environment where TD and SPD are the important performance factors we have the following:

Theorem: If for the group $(G,*)$, $m : G \longrightarrow \mathcal{S}$ is an AWGN matched labeling, then, $\forall \mathbf{g}, \mathbf{g}' \in C < G^I, I = Z_n$,

$$\begin{cases} \mathrm{TD}(\mathbf{m(g)}, \mathbf{m(g')}) = \mathrm{TD}\left(\mathbf{m(g^{-1} * g')}, \mathbf{m(e)}\right) \\ \mathrm{SPD}(\mathbf{m(g)}, \mathbf{m(g')}) = \mathrm{SPD}(\mathbf{m(g^{-1} * g')}, \mathbf{m(e)}) \end{cases}$$

in which case we would say S is matched to $(C,*)$ in the Rayleigh sense.

Proof: Denote $\mathbf{g} = [g_1, g_2, ..., g_n]$, and $\mathbf{g'} = [g'_1, g'_2, ..., g'_n]$. then,

$$\mathrm{TD}(\mathbf{m(g)}, \mathbf{m(g')}) = \sum_{i=1}^{n} [g_i \neq g'_i]$$

where the function $[P]$ returns 1 if P is true and 0 if it is false. Then,

$$\mathrm{TD}(\mathbf{m(g)}, \mathbf{m(g')}) = \sum_{i=1}^{n} [g_i^{-1} * g'_i \neq e] = \mathrm{TD}(m(\mathbf{g^{-1} * g'}), m(\mathbf{e}))$$

and,

$$\mathrm{SPD}(\mathbf{m(g)}, \mathbf{m(g')}) = \prod_{i \in \zeta} d^2(m(g_i), m(g'_i))$$

where the time diversity set ζ is defined as, $\zeta = \{i \mid m(g_i) \neq m(g'_i)\}$. Then,

$$\begin{aligned} \mathrm{SPD}(\mathbf{m(g)}, \mathbf{m(g')}) &= \textstyle\prod_{i \in \zeta} d^2(m(g_i^{-1} * g'_i), m(e)) \\ &= \mathrm{SPD}(\mathbf{m(g^{-1} * g')}, \mathbf{m(e)}) \end{aligned}$$ ■

When the group G is taken as Z_q and I as a simple index set of $\{1, 2, ..., n\}$, $\mathrm{C} < Z_q^n$ will be a linear code over Z_q, in other words a submodule in which case the concept of linear independence can be used to represent the scheme in terms of a generator. Caire and Biglieri [6] considered that problem for the special case of $q = p^\ell$ with p a prime. They use the invariant factor theorem [10] for a principal ideal domain (pid) to find a basis and thus a generator. Then, having the generator in hand, one can use the method of Vazirani et al. [11] to produce a trellis-oriented generator which will result in the minimal trellis representation [12, 7] of the scheme. This generator-based method of Vazirani et al. uses the so-called p-linear combination of vectors of the module. We extend that concept to p-constraints resulting from the extension of the H matrix of the underlying binary code and therefore construct the minimal trellis using the p-constraints rather than the generator matrix. In this correspondence, we will only consider $p = 2$.

As an example of the code construction, consider the $\mathrm{RM}(1,3) = (8,4,4)$ binary code with the parity check matrix,

$$H = [h_{ij}] = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}. \qquad (1)$$

the resulting 8-ary code is composed of codewords $(x_1, x_2, ..., x_8)$ satisfying:

$$\begin{cases} x_1 + x_2 + x_3 + x_4 = 0 \mod 8 \\ x_3 + x_4 + x_5 + x_6 = 0 \mod 8 \\ x_5 + x_6 + x_7 + x_8 = 0 \mod 8 \\ x_1 + x_3 + x_5 + x_7 = 0 \mod 8 \end{cases}. \qquad (2)$$

This 8-ary 16-dimensional scheme bears a TD and BWE of 4 symbol and 1.5 bits/2-D symbol, respectively. The Rayleigh fading simulation of this scheme indicates a bit error probability of 10^{-5} at a bit SNR of 17.9 dB which shows improvement with respect to many comparable schemes in the literature in terms of both power- and bandwidth-efficiency (e.g., with respect to those of L.-F. Wei [4]).

III. Maximum-likelihood decoding

The maximum likelihood (ML) soft-decoding complexity of a code C is one of the major concerns in its application. The main known approaches to the soft decision decoding of block codes are: (i) Viterbi algorithm, which is a trellis-based approach; and (ii) Coset decoding. In Coset decoding, the basic idea is to decompose the set of codewords into a number of similar subsets (a subcode and its cosets) which have some nice properties that simplify the soft decision decoding of the code. It is believed that appropriate graphical representations of codes will contribute in their decoding complexity. The well-known graphical models presented for linear codes are the trellis diagram , the factor or Tanner graph (TG) , the Tanner-Wiberg-Loeliger graph , and the Bayesian network.

A trellis diagram is a regular, directed, and finite-state graph reminiscent of a garden trellis, a concept which is very well-understood today.

A Tanner graph representing a linear block code of length n and dimension k with a parity check matrix $H_{(n-k)\times n} = [h_{ij}]$,

[2] A linear code C over a ring R is a submodule of the Z-module over R.

[3] Two simple choices for the index set are $I = Z$ and $I = Z_n$ which correspond to trellis and block codes, respectively.

is a bipartite graph[4] in which one of the two sets of vertices denote the parity nodes (the rows of H) and the other set denotes the symbol nodes (the columns of H). A parity node u_i is connected to a symbol node v_j, iff $h_{ij} \neq 0$.

A Bayesian network is a directed acyclic graph which can be used similar to a factor graph to show how a global function factors to local functions.

These graphical models can be used to suggest efficient ML trellis-based or even iterative decoding methods for our construction.

In our approach, to decode the constructed BCM scheme, we will decompose the codebook which has a cyclic Tanner graph (TG), to a subcode with an Acyclic Tanner Graph (ATG), and its cosets. The significance of representing each component code by an ATG is that, one can use a generalization of the well-known *Wagner* rule [13] for their decoding.

Minimal Tanner Graph[5] (MTG) and Bayesian net for the example of the previous section are shown in Figure 1. These are exactly the same graphs as those for the underlying binary code RM$(1,3)$ with the only difference of having different local functions in them (mod 8 addition as compared to mod 2 addition). There is a great deal of similarity between the underlying binary code representations and the q-ary code as well as some discrepancies which will be revealed in more detail. For instance, a multilevel coset decomposition of these schemes provides a very clear minimal trellis representation for them which for the binary case (over a field) agrees with methods of others such as Forney [7]. In this correspondence, we will use a 2-level decoding method.

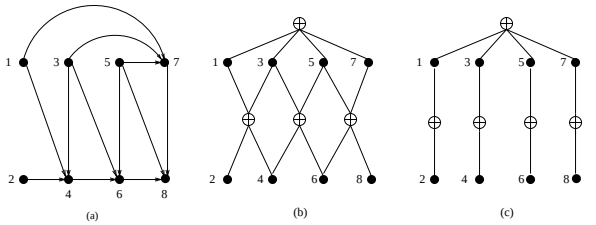

Figure 1: **a**) Bayesian net, **b**) Tanner graph, and **c**) Tanner graph of the acyclic subcode for the $(8,4,4)$ RM code.

IV. Maximum-likelihood decoding of the scheme

One of our important motivations in using well-known binary block codes in the construction of our BCM schemes has been their ease of decoding. The literature of the binary block codes is very rich since much research has been devoted to this area up to date. Thus, using such codes will enable us to exploit the knowledge about the structure of these codes in the decoding algorithm of the corresponding BCM scheme.

In our approach, to decode the constructed BCM scheme, we will decompose the code which has a cyclic Tanner graph, to one of its subcodes with an ATG, and its cosets. This subcode is referred to as the *base subcode*. This is done by adding new constraints to the parity check matrix which will result in the removal of the cycles. Then the ATG of the base subcode and a trellis representation for the so-called *parity space* will represent the whole code.

As the constraints on the code symbols are the same as parity equations of a binary code extended to q-ary, it is trivial to see that the TG of the resulting q-ary scheme is the same as that of the underlying binary code with changing the local function from addition mod 2 to addition mod q.

In order to remove the cycles in a cyclic Tanner graph, one needs to add some extra constraints to the existing parity-check constraints. The necessary constraints for the removal of the cycles and the resulting base subcode ATG are also the same for both binary and q-ary cases. So given a binary block code to be used for our construction, first, one needs to find its base subcode and its ATG and second, obtain the set of equations governing the structure of the resulting parity space for mod q.

To address the first problem, let us consider $C = (n,k,d)$ over GF(2), with generator matrix $G_{k\times n}$ and parity-check matrix $H_{(n-k)\times n}$ as the underlying binary code. As cycle-free TGs cannot support good codes [**?**], this code will have a cycle space with nonzero rank. Suppose $C_0 = (n,k_0,d_0)$ is the acyclic base subcode. Then the generator G of C can be partitioned to

$$G = \begin{bmatrix} G_0 \\ G_c \end{bmatrix}$$

where G_0 is the generator for C_0. Thus C is just the sum of C_0 and C_c as follows: $C = C_0 + C_c = \{c_1 + c_2 | c_1 \in C_0, c_2 \in C_c\}$. Having the generators G and G_0 in hand and accordingly the kernel representations H and H_0, one can use the elementary row operations to put H_0 in the form

$$H_0 = \begin{bmatrix} H \\ H_e \end{bmatrix}.$$

Here H_e gives the set of extra constraints that should be added to H to remove the cycles. An alternative method to remove the cycles is the graph-theoretic approach that we will not get into.

Over GF(2), it is also very easy to find the generator for the parity space and thus answer the second problem as well. If c is an arbitrary codeword in C, i.e., $c \in C$, then there exists an information vector $u_{1\times k}$ such that, $c = u \cdot G$. If c is also in C_0, then $c \cdot H_0^T = s_{1\times(n-k_0)} = 0$, otherwise $s \neq 0$. In other words, the syndrome s in this case is the parity vector for the base subcode and its cosets. Our objective is to find the generator for s. We have,

$$\begin{aligned} s = c \cdot H_0^T &= u \cdot G \cdot H_0^T = u \cdot \begin{bmatrix} G_0 \\ G_c \end{bmatrix} \cdot H_0^T \\ &= [u_0 \;\; u'] \cdot \begin{bmatrix} G_0 \cdot H_0^T \\ G_c \cdot H_0^T \end{bmatrix} \\ &= u_0 \cdot G_0 \cdot H_0^T + u' \cdot G_c \cdot H_0^T = u' \cdot G_c \cdot H_0^T. \end{aligned}$$

The $(k-k_0)\times(n-k_0)$ matrix $G_{PS} = G_c \cdot H_0^T$ is the generator for the parity space.

For codes of practical interest (e.g., Reed-Muller or Hamming codes), the maximal[6] base subcode is a *Generalized Sin-*

[4] A bipartite graph G, is a graph whose vertex set V can be partitioned into two disjoint nonempty sets, V_1 and V_2, such that every edge in the graph connects a vertex in V_1 and a vertex in V_2 (so that no edge in G connects either two vertices in V_1 or two vertices in V_2).

[5] MTG is the Tanner graph with the minimum number of edges in the graph which corresponds to the parity-check matrix with the minimum number of nonzero elements.

[6] This is the subcode with the minimum index.

gle Parity (GSP) code. A linear code having a tree[7] Tanner graph is defined to be a generalized single parity code if at most one of the parity nodes, denoted as *root parity*, is of degree[8] more than 2. It is obvious that the generator matrix G of a GSP code C with a root parity of degree $b > 2$ is obtained from the generator matrix of even weight, or single-parity (SP) code $\mathcal{E}_b = (b, b-1, 2)$, by replacing the nonzero entries of each column by a repetition code that has the same length as the corresponding branch of TG. In the trivial repetition code, $\mathcal{R}_\ell = (\ell, 1, \ell)$, all the parity nodes have the same degree of 2.

If all the branches connected to the root parity are of the same length, the linear code is called a *Uniform Generalized Single Parity* (UGSP) code. This means that the code is of the form of a product code, namely, $\mathcal{E}_b \otimes \mathcal{R}_\ell$, where "$\otimes$" is the Kronecker[9] product of codes. Usually, $\ell = \lceil d/2 \rceil$.

For instance for the class of Reed-Muller codes, the maximal acyclic subcode is a $(2^{r+1} - 1)$-dimensional UGSP of the form:

$$\mathcal{E}_b \otimes \mathcal{R}_\ell = (b, b-1, 2) \otimes (\ell, 1, \ell) = (b\ell, b-1, 2\ell) \qquad (3)$$

where, $b = 2^{r+1}$, and $\ell = 2^{m-r-1} = d/2$. The corresponding

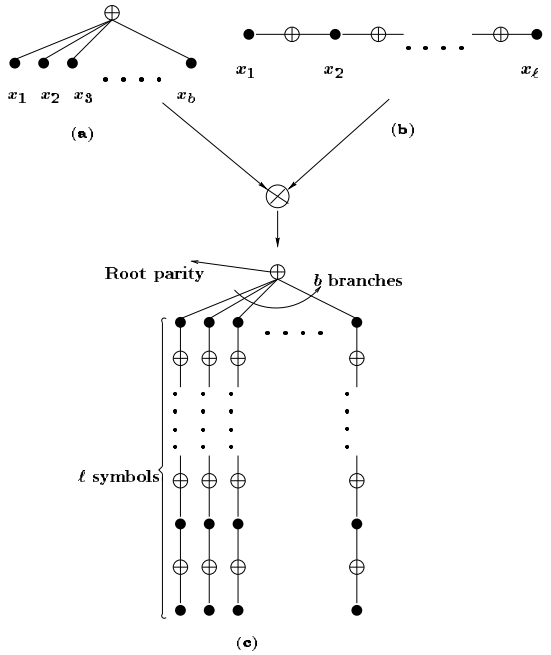

Figure 2: Acyclic Tanner Graphs for **a**) Single Parity Code, $\mathcal{E}_b$, **b**) Repetition Code, $\mathcal{R}_\ell$, **c**) The Product Code, $\mathcal{E}_b \otimes \mathcal{R}_\ell$.

ATG is composed of b branches of length ℓ connected to a root parity as shown in Figure 2.

[7]We concentrate on codes which have a tree (connected) TG. If TG is disconnected, then obviously each component of the TG represents a subcode of the code such that the direct-sum of them adds to the code itself.

[8]The degree of a node is defined as the number of the edges connected to it.

[9]The Kronecker product (also called the direct product or simply the product) of two linear codes $C_1 = (n_1, k_1, d_1)$ and $C_2 = (n_2, k_2, d_2)$ is defined as the code $C = (n_1 n_2, k_1 k_2, d_1 d_2)$ whose codewords consist of all $n_1 \times n_2$ arrays constructed such that the rows of the array are codewords of the first code and its columns are the codewords of the second code [?].

If a code is obtained by the shortening of another code, then it inherits many of the properties of the original code. As an example of such inheritance, an ATG for the shortened code can be obtained by shortening the ATG of the original code. Thus, the ATG for the Hamming code $\mathcal{H}_m = (2^m - 1, 2^m - 1 - m, 3)$, which is the shortened Reed-Muller code $\mathcal{RM}(m-2, m)$, can be obtained by deleting one ending symbol node from the ATG of the Reed-Muller code.

In the q-ary case finding the generator or constraints governing the structure of the parity space is not as straightforward as it is for the binary case. In order to describe our method for non-binary cases, let us consider the $(8, 4, 4)$ code over Z_8 in (2). Using the method described before, one will see that only one constraint is needed to remove all the cycles at once. That equation that we have to add to the system of equations in (2) is, $x_1 + x_2 = 0 \mod 8$, and therefore the base subcode C_0 is an $(8, 3, 4)$ code consisting of codewords $(x_1, x_2, ..., x_8)$ satisfying:

$$\begin{cases} x_1 + x_2 + x_3 + x_4 = 0 \mod 8 \\ x_3 + x_4 + x_5 + x_6 = 0 \mod 8 \\ x_5 + x_6 + x_7 + x_8 = 0 \mod 8 \\ x_1 + x_3 + x_5 + x_7 = 0 \mod 8 \\ x_1 + x_2 = 0 \mod 8 \end{cases} . \qquad (4)$$

If the added equation is changed to $x_1 + x_2 = p_1 \mod 8$, when p_1 takes different values from Z_8 we will produce different cosets of the base subcode including the C_0 itself for $p_1 = 0$. Now it is trivial to see that if we name,

$$\begin{cases} x_1 + x_2 = p_1 \mod 8 \\ x_3 + x_4 = p_2 \mod 8 \\ x_5 + x_6 = p_3 \mod 8 \\ x_7 + x_8 = p_4 \mod 8 \end{cases}$$

the set of equations in (4) can be reduced to:

$$\begin{cases} p_1 + p_2 = 0 \mod 8 \\ p_2 + p_3 = 0 \mod 8 \\ p_3 + p_4 = 0 \mod 8 \\ x_1 + x_3 + x_5 + x_7 = 0 \mod 8 \end{cases} \qquad (5)$$

The last equation in (5) corresponds to the root parity which remains constant at zero for all the cosets. The other three constraints are the ones regulating the structure of the parity space. A trellis diagram for this parity space and the TG of the base subcode comprise a composite graphical representation for the corresponding BCM scheme. This trellis-Tanner graph (T-TG) representation is used for a very efficient ML decoding, as a lot of the metric computations are repeated and need to be calculated only once.

For the example under consideration, the parity space trellis is shown in Figure 3. The root parity need not be considered as it does not have any dynamics. The parity space for p_1, p_2, p_3, and p_4 is itself a $(4, 1, 4)$ code over Z_8.

Now let us show in more detail, the amount of complexity and memory requirement involved in the ML decoding of the designed schemes. We design the parity space trellis such that each section represents the parities in one branch of the base subcode TG, i.e., for a code with a base subcode TG as in Figure 2c we will have a b-section parity space trellis. Each path in this trellis from the initial state (root) to the final state

Figure 3: **a**) The parity space trellis for the $(8, 4, 4)$ code over Z_8.

(toor or goal) corresponds to one coset of the decomposition. For the sake of brevity and without loss of generality let us consider the TG in Figure 2c as our subcode ATG. Then in the parity space trellis each branch actually corresponds to $\ell - 1$ parity nodes and ℓ code symbols and accordingly it represents q parallel paths. We assign to each path a q-tuple, $(\mathcal{M}_1, \mathcal{M}_2, ..., \mathcal{M}_q)$, where $\mathcal{M}_i$ is the cost of that branch contributing i units to the root parity of the ATG. Also to each state of the trellis we associate a similar q-tuple whose ith element denotes the minimum cost of contributing i unit to the root parity of the ATG from the root up to that state. Therefore, one stage of the decoding is to determine the cost assocaited with the q parallel paths and one other stage involves finding a survivor path and metric for each state and $1 \leq i \leq q$. In fact like the binary case we can normalize each q-tuple to one of its elemnts consider the ratios. In this way we only need to store a a $(q-1)$-tuple for each survivor. Of course for the final state we only require the minimum cost of contributing 0 to the root parity which will be our decoding decision.

V. Summary and Conclusion

References

[1] J. L. Massey, "Coding and modulation in digital communication," in Proc. 1974 *International Zürich Seminar on Digital Communication*, Zürich, Switzerland, pp. E2(1)-E2(4), Mar. 1974.

[2] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory*, Vol. IT-28, No. 1, pp. 55-67, Jan. 1982.

[3] H. Imai, and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Trans. Inform. Theory*, Vol. IT-23, pp. 371-377, May 1977.

[4] L.-F. Wei, "Coded M-DPSK with built-in time diversity for fading channels," *IEEE Trans. Inform. Theory*, Vol. IT-39, no. 6, pp. 1820-1839, Nov. 1993.

[5] E. D. Biglieri, D. Divsalar, P. J. McLane, and M. K. Simon, *Introduction to Trellis-Coded Modulation with Applications*. New York: Macmillan, 1991.

[6] G. Caire, and E. Biglieri, "Linear block codes over cyclic groups," *IEEE Trans. Inform. Theory*, vol. IT-41, Sept. 1995, pp. 1246-1256.

[7] G. D. Forney, Jr., "Coset codes-Part I: Introduction and geometrical classification and Part II: Binary lattices and related codes," *IEEE Trans. Inform. Theory*, vol. IT-34, pp. 1123-1187, Sept. 1988.

[8] H.-A. Loeliger, "Signal sets matched to groups," *IEEE Trans. Inform. Theory*, vol. IT-37, Nov. 1991, pp. 1675-1682.

[9] G. D. Forney, Jr., "Geometrically uniform codes," *IEEE Trans. Inform. Theory*, vol. IT-37, pp. 1241-1260, Sept. 1991.

[10] C. W. Curtis, and I. Reiner, *Representation Theory of Finite Groups and Associative Algebras*. New York: Wiley Edition, 1988.

[11] V. V. Vazirani, H. Saran, and B. S. Rajan, "An efficient algorithm for constructing minimal trellises for codes over finite Abelian groups," *IEEE Trans. Inform. Theory*, vol. IT-42, Nov. 1996, pp. 1839-1854.

[12] F. R. Kschischang, and V. Sorokine, "On the trellis structure of block codes," *IEEE Trans. Inform. Theory*, vol. IT-41, Nov. 1995, pp. 1924-1937.

[13] R. A. Silverman, and M. Balser, "Coding for a constant data-rate source," *IRE Trans. Inform. Theory*, vol. PG IT-4, pp. 50-63, 1954.