# Subspace Rings: A New Tool in Communication Trellis Design

John Kieffer
Dept. of Electrical & Computer Engineering
University of Minnesota
Minneapolis, MN 55455
Email: kieffer@umn.edu

***Abstract*— By a "communication trellis", we mean a trellis whose edges are labelled in order that the trellis may be used for any of the following three communication tasks: (i) error-correction encoding, (ii) modulation, or (iii) quantization for source coding. We view the design of such a labelled trellis as a two-step procedure: Step 1 consists of the design of one or more patterns which potential sequences of trellis edge labels must conform to; in Step 2, an objective function suited to whichever of communication tasks (i)-(iii) is the design goal is used to select a sequence of trellis edge labels conforming to the pattern(s) chosen in Step 1. In this paper, we introduce the concept of subspace ring, which will be useful to us in the future in performing Step 1. To this end, each possible pattern of trellis edge labels is specified via a subspace $S$ of some fixed dimension of a vector space $V$; by letting $S$ vary, we obtain all possible allowable patterns. We show how some of these subspaces can be grouped together to form our subspace rings. We illustrate how some properties useful in communication trellis design can be characterized in terms of the subspace ring concept.**

## I. Introduction

In Fig. 1, the lefthand trellis stage is labelled to yield a rate $1/2$ convolutional encoder ([1], p. 277), the middle trellis stage is labelled to yield a trellis-coded modulation scheme ([2], p. 117), and the righthand trellis stage is labelled to yield a trellis-coded quantizer ([4], p. 133; the labels $D_i$ are called "subcodebooks", but the significance of the subcodebook concept need not concern us here).

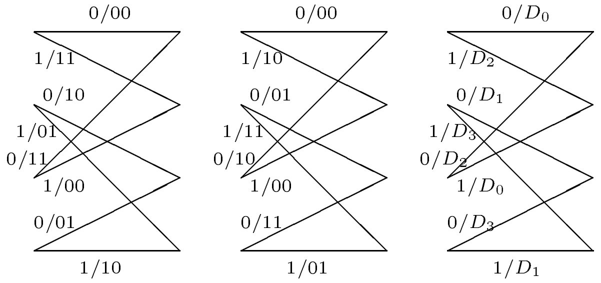

**Fig. 1: Trellis-based convolutional encoder, trellis-coded modulator, and trellis-coded quantizer (left to right)**

Each trellis edge label in the Fig. 1 trellises is of the form $i/j$, where $i$ is an "input label" and $j$ is an "output label". The input labels are trivial because of the convention in trellis labelling that the upper of the two edges from each trellis state receives input label $0$ and the lower of the two edges receives input label $1$. It is only the output labels that are significant in Fig. 1. In each of the Fig. 1 trellises, suppose we read off the sequence of edge output labels, traversing the edges from top to bottom. Doing this, we obtain the following three sequences of length 8:

$$00, 11, 10, 01, 11, 00, 01, 10 \tag{1}$$

$$00, 10, 01, 11, 10, 00, 11, 01 \tag{2}$$

$$D_0, D_2, D_1, D_3, D_2, D_0, D_3, D_1 \tag{3}$$

The *pattern* corresponding to each of the three sequences of edge labels (1)-(3) is

$$a, b, c, d, b, a, d, c. \tag{4}$$

For example, reading (1) from left to right, the first distinct label we see is assigned the pattern symbol "$a$", the second distinct label we see is assigned the pattern symbol "$b$", etc., resulting in the sequence (4).

Our point in this introduction is that whether a trellis is used in communications for (i) error-correction encoding (leftmost trellis in Fig. 1), (ii) modulation (middle trellis in Fig. 1), or (iii) quantization in lossy source coding (rightmost trellis in Fig. 1), the *pattern* followed by the trellis edge labels may be independent of whichever of the three types of communication trellises (i)-(iii) that one is designing. Therefore, it makes sense to find a procedure to select good patterns for trellis label sequences. Once such patterns have been isolated (Step 1), one can address as a separate problem (considered elsewhere) the problem (Step 2) of selecting a particular trellis label sequence suited to one of the applications (i)-(iii), whose pattern will be one of those selected in Step 1.

## II. Subspaces and Patterns

In most trellis codes, the cardinality of the pattern symbol alphabet is a power of two $> 2$. Due to limited space, we concentrate throughout on the simplest (but nontrivial) case in which this cardinality is equal to $4$; accordingly, we take the pattern symbol alphabet to be $\{a, b, c, d\}$. Let $V_n$ denote the vector space of dimension $n$ over the binary field, realized as the set of all binary $n$-tuples. Suppose we have a subspace $S$ of $V_n$ of codimension two. Then there are four cosets of $S$, including $S$ itself. As in [3], we assign symbol $a$ to each member of $S$ and assign symbols $b, c, d$ (respectively) to each member of each of the other three cosets (respectively), thereby obtaining a pattern for labelling the edges of the usual de Bruijn graph based trellis; this trellis has $2^{n-1}$ states and $2^n$ edges (two outgoing edges per state), and its edges are in natural one-to-one correspondence with the vectors in $V_n$.

*Example 1.* We represent each vector in $V_4$ as the 4-bit expansion of an integer between $0$ and $15$, inclusively. We show how to label the edges of an $8$-state trellis using

$$S = \{0 = (0000), 5 = (0101), 11 = (1011), 14 = (1110)\}, \tag{5}$$

a subspace of $V_4$ of codimension two. The 16-bit "indicator sequence" of $S$ is

$$1000010000010010, \tag{6}$$

where the 1's in (6) occur in positions $0, 5, 11, 14$, corresponding to the vectors in $S$ in (5). (We number positions in indicator sequences of subspaces of $V_4$ from left to right, starting with position $0$ and ending with position $15$.) Consider the set of four sequences

$$\{abcd, badc, cdab, dcba\}. \tag{7}$$

Partition indicator sequence (6) into four 4-bit blocks and then convert these blocks according to the following algorithm:

$$\begin{array}{ccc} 1000 & \rightarrow & abcd \\ 0100 & \rightarrow & badc \\ 0010 & \rightarrow & cdab \\ 0001 & \rightarrow & dcba \end{array}$$

Following this procedure, the indicator sequence (6) is converted into the pattern $abcdbadcdcbacdab$. We have used this pattern to label the edges of the lefthand trellis in Fig. 2.

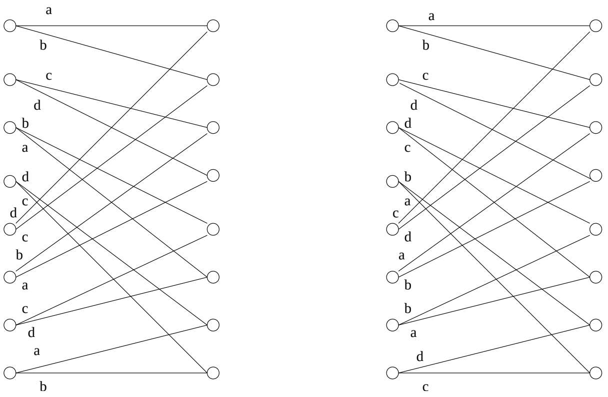

**Fig. 2: Reverse Pair of Trellis Codes For 8-State Trellis**

Consider the permutation

$$\begin{array}{cccccccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 0 & 8 & 4 & 12 & 2 & 10 & 6 & 14 & 1 & 9 & 5 & 13 & 3 & 11 & 7 & 15 \end{array}$$

obtained by reversing each $4$-bit expansion. Applying this permutation to the pattern in the lefthand trellis in Fig. 2, one obtains the pattern imposed on the righthand trellis in Fig. 2. The two labelled trellises in Fig. 2 are called a *reverse pair* of trellis codes (or each of the codes is said to be the *reverse code* of the other code). What this concept means physically is that you obtain either code in Fig. 2 from the other one by running the trellis backward in time instead of forward in time.

With a bit of work, it can be shown that the $35$ subspaces of $V_4$ of codimension two yield $21$ reverse pairs of trellis codes (some code pairs consist of two identical codes). The patterns of these pairs are (row by row, with the patterns of the Fig. 2 pair of codes in the first row)

| | |
|---|---|
| $abcdbadcdcbacdab$ | $abcddcbacdabbadc$ |
| $aaaabbbbccccdddd$ | $abcdabcdabcdabcd$ |
| $aabbaabbccddccdd$ | $ababcdcdababcdcd$ |
| $aabbbbaaccddddcc$ | $abcdcdababcdcdab$ |
| $aabbccddaabbccdd$ | $aabbccddaabbccdd$ |
| $aabbccddbbaaddcc$ | $abcdbadcabcdbadc$ |
| $aabbccddccddaabb$ | $abbacddcabbacddc$ |
| $aabbccddddccbbaa$ | $abcddcbaabcddcba$ |
| $ababababcdcdcdcd$ | $ababababcdcdcdcd$ |
| $ababbabacdcddcdc$ | $abcdabcdcdabcdab$ |
| $ababcdcdbabadcdc$ | $abcdabcdbadcbadc$ |
| $ababcdcdcdcdabab$ | $abbaabbacddccddc$ |
| $ababcdcddcdcbaba$ | $abcdabcddcbadcba$ |
| $abbabaabcddcdccd$ | $abcdcdabcdababcd$ |
| $abbacddcbaabdccd$ | $abcdbadcbadcabcd$ |
| $abbacddccddcabba$ | $abbacddccddcabba$ |
| $abbacddcdccdbaab$ | $abcddcbadcbaabcd$ |
| $abcdbadccdabdcba$ | $abcdbadccdabdcba$ |
| $abcdcdabbadcdcba$ | $abcdcdabbadcdcba$ |
| $abcdcdabdcbabadc$ | $abcdcdabdcbabadc$ |
| $abcddcbabadccdab$ | $abcddcbabadccdab.$ |

## III. Subspaces and Parity Check Matrices

Let $S$ be a subspace of $V_n$ of codimension two. Then $S^\perp$, the orthogonal complement of $S$, is a subspace of $V_n$ of dimension 2. Forming a matrix consisting of the three nonzero vectors in $S^\perp$, we define this matrix to be the *parity check matrix* of $S$. The parity check matrix is uniquely defined up to ordering of the rows (the way in which the rows are ordered makes no difference). Given a trellis code, by which we mean a labelling of the edges of the $2^{n-1}$-state de Bruijn graph with labels from $\{a, b, c, d\}$ corresponding to a subspace $S$ of $V_n$, we define the parity check matrix of this code to be the parity check matrix of $S$.

The parity check matrix $H$ of a subspace $S$ of $V_n$ of codimension two obeys the following rules:

(i): There are three rows of $H$, which are each nonzero vectors in $V_n$.

(ii): The three rows of $H$ sum to the zero vector.

(iii): Any two of the rows of $H$ are linearly independent.

Conversely, suppose we have any matrix $H$ satisfying properties (i)-(iii). Then there is a unique subspace $S$ of $V_n$ of codimension two whose parity check matrix is $H$. There is thus a one-to-one correspondence between subspaces $S$ of $V_n$ of codimension two and matrices $H$ satisfying (i)-(iii).

*Example 2.* The parity check matrices of the reverse pair of trellis codes in Fig. 2 are

$$\begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

respectively. In general, reverse pairs of codes always have parity check matrices which are left-to-right reversals of each other.

## IV. Subspace Rings

It is possible to form sets of subspaces which we can make into rings by appropriately defining the ring addition/multiplication operations; the parity check matrices of the subspaces in a subspace ring will all have some fixed row in common.

We will define three types of subspace rings.

### *A. Type 1 Subspace Rings*

We will call a binary sequence $v$ a Type 1 sequence if it begins and ends with 1. Let $v \in V_n$ be a Type 1 sequence. Then the Type 1 ring $R(v)$ consists of all subspaces $S$ of $V_n$ of codimension 2 whose parity check matrix has $v$ as one of its rows together with the subspace of $V_n$ of codimension 1 whose orthogonal complement is the subspace spanned by $v$. We discuss how we make $R(v)$ into a ring. Suppose we index the coordinates of $v$ as

$$v = (v_{n-1}, v_{n-2}, \cdots, v_1, v_0),$$

where $v_{n-1} = v_0 = 1$. Let $p_v(x)$ be the polynomial

$$p_v(x) = v_{n-1}x^{n-1} + v_{n-2}x^{n-2} + \cdots + v_2x^2 + v_1x + v_0, \quad (8)$$

a polynomial of degree $n-1$. Consider the residue class ring

$$R = GF(2)[x]/ < p_v(x) > .$$

We can consider the members of $R$ to be the $2^{n-1}$ polynomials of degree at most $n-2$ which have binary coefficients. If we multiply two such polynomials and then compute the remainder modulo $p_v(x)$, we obtain the multiplication operation in ring $R$. If we add two such polynomials in the usual way we add polynomials, we obtain the addition operation in ring $R$. We will make use of a one-to-one correspondence between $R$ and $R(v)$ to carry over the ring operations on $R$ into ring operations on $R(v)$. Suppose $q(x) = \sum_{i=0}^{n-2} a_i x^i$ is a polynomial in $R$. The subspace in $R(v)$ corresponding to $q(x)$ is the subspace of $V_n$ whose parity check matrix is

$$\begin{bmatrix} 1 & v_{n-2} & \cdots & v_1 & 1 \\ 0 & a_{n-2} & \cdots & a_1 & a_0 \\ 1 & v_{n-2}+a_{n-2} & \cdots & v_1+a_1 & 1+a_0 \end{bmatrix}.$$

The numbers $a_0, a_1, \cdots, a_{n-2}$ are arbitrary binary parameters.

*Example 3.* We consider the Type 1 ring $R(1100111)$, which consists of certain subspaces of $V_7$. For each subspace $S$ in $R(1100111)$, let $H(S)$ denote the parity check matrix of $S$. Let us denote the additive identity of ring $R(1100111)$ by $O$. Then

$$H(O) = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

This is one instance in which we depart from our restriction that the three rows of a parity check matrix be of rank two. In this case, we have a zero row, which means that the subspace $O$ is of codimension 1 instead of codimension 2. The parity check matrix of any other member of $R(1100111)$ will be of the usual form, i.e., corresponding to a subspace of codimension 2. For example, the identity element $I$ of $R(1100111)$ (the multiplicative identity) is the subspace of $V_7$ for which

$$H(I) = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}.$$

### *B. Type 2 Subspace Rings*

We will call a binary sequence $v$ a Type 2 sequence if either

(a): it begins with 01 and ends in 1; or

(b): it begins with 1 and ends with 10.

Corresponding to each Type 2 sequence $v \in V_n$, we will define a ring $R(v)$ of $2^{n-2}$ subspaces of $V_n$ of codimension two. We call $R(v)$ a Type 2 ring. The Type 2 ring $R(v)$ is defined differently depending upon whether (a) or (b) above is satisfied. In the following, we treat these cases separately.

*1) Type 2 Ring $R(v)$ under assumption (a):* Throughout this subsection, we take $v \in V_n$ to be a Type 2 sequence satisyfing (a) above. Then Type 2 subspace ring $R(v)$ consists of the $2^{n-2}$ subspaces of $V_n$ of codimension two whose parity check matrices each have $v$ as a row and have linearly independent leftmost and next to leftmost columns. Writing $v = (v_{n-1}, v_{n-2}, v_{n-3}, \cdots, v_1, v_0)$, with $v_{n-1} = 0$ and $v_{n-2} = v_0 = 1$, the general form of the parity check matrix for subspaces in $R(v)$ is

$$\begin{bmatrix} 0 & 1 & v_{n-3} & \cdots & v_1 & 1 \\ 1 & 0 & a_{n-3} & \cdots & a_1 & a_0 \\ 1 & 1 & v_{n-3}+a_{n-3} & \cdots & v_1+a_1 & 1+a_0 \end{bmatrix}. \quad (9)$$

The $n-2$ binary numbers $a_0, a_1, \cdots, a_{n-3}$ are arbitrary.

In order to define the addition and multiplication in $R(v)$ that will make $R(v)$ a ring, we make use of a one-to-one correspondence between $R(v)$ and the residue ring $R = GF(2)[x]/ < p_v(x) >$, where $p_v(x)$ is the polynomial form of $v$ defined by (8). (Here, $p_v(x)$ is a polynomial of degree $n-2$.) The members of $R$ can be regarded as being all polynomials in $GF(2)[x]$ of degree less than or equal to $n-3$. Let us refer to the general form of parity check matrix in (9). We explain how to form the polynomial in $R$ corresponding to this. First, we remark that $(1, v_{n-3})$ coincides either with the first two entries of the second row of (9) or the third row of (9). Pick whichever of these two rows for which this is true, and let $q(x)$ be the polynomial form of this row. Then the polynomial in $R$ corresponding to (9) is $q(x) + xp_v(x)$ (which is of degree at most $n-3$ since the $x^{n-1}, x^{n-2}$ terms in $q(x)$ must cancel with the $x^{n-1}, x^{n-2}$ terms in $xp_v(x)$). Now that we have a one-to-one correspondence between $R(v)$ and $R$, we use this correspondence to carry over the addition/multiplication in $R$ to addition/multiplication in $R(v)$.

*Example 4.* We examine the ring $R(0101001)$. The zero element $O$ is given by

$$H(O) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}.$$

Notice that the second row is just a cyclic shift of the first row. The identity element $I$ is given by

$$H(I) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}.$$

Suppose

$$\begin{aligned} H(S_1) &= \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}, \\ H(S_2) &= \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}. \end{aligned}$$

We compute $H(S_1 \oplus S_2)$ and $H(S_1 \otimes S_2)$, where $\oplus$ and $\otimes$ denote, respectively, the addition and multiplication operations in $R(0101001)$. Since the second and third entries of $(0101001)$ are 10, we focus on the second row of $H(S_1)$ and $H(S_2)$, which have polynomial forms

$$x^6 + x^4 + x^3 + x^2 + x + 1, \quad x^6 + x^3 + x^2 + 1. \tag{10}$$

For $v = (0101001)$, we have

$$xp_v(x) = x^6 + x^4 + x.$$

Adding $xp_v(x)$ to each of the polynomials in (10), we obtain the following polynomials in the residue ring $GF(2)[x]/ < x^5 + x^3 + 1 >$:

$$x^3 + x^2 + 1, \quad x^3 + x^2 + x + 1.$$

We add and multiply these two polynomials as members of the residue ring:

$$\begin{aligned} (x^3 + x^2 + 1) + (x^3 + x^2 + x + 1) &= x, \\ (x^3 + x^2 + 1) * (x^3 + x^2 + x + 1) &= x^4 + x^3 + 1. \end{aligned}$$

We can convert these two results to rows of $H(S_1 \oplus S_2)$ and $H(S_1 \otimes S_2)$, respectively, by adding $x^6 + x^4 + x$ to them, obtaining the following polynomials and their vector forms:

$$\begin{aligned} x + (x^6 + x^4 + x) &\leftrightarrow (1010000), \\ x^4 + x^3 + 1 + (x^6 + x^4 + x) &\leftrightarrow (1001011). \end{aligned}$$

Therefore,

$$\begin{aligned} H(S_1 \oplus S_2) &= \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}, \\ H(S_1 \otimes S_2) &= \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \end{aligned}$$

*2) Type 2 Ring $R(v)$ under assumption (b):* Throughout this subsection, we take $v \in V_n$ to be a Type 2 sequence beginning with 1 and ending with 10. Then Type 2 subspace ring $R(v)$ consists of the $2^{n-2}$ subspaces of $V_n$ of codimension two whose parity check matrices each have $v$ as a row and have linearly independent rightmost and next to rightmost columns. Writing $v = (1, v_{n-2}, \cdots, v_2, 1, 0)$, the general form of the parity check matrix for subspaces in $R(v)$ is

$$\begin{bmatrix} 1 & v_{n-2} & \cdots & v_2 & 1 & 0 \\ 0 & a_{n-2} & \cdots & a_2 & a_1 & 1 \\ 1 & v_{n-2} + a_{n-2} & \cdots & v_2 + a_2 & 1 + a_1 & 1 \end{bmatrix}, \tag{11}$$

where the $n-2$ binary numbers $a_1, a_2, \cdots, a_{n-2}$ are arbitrary.

Let $u$ be the sequence

$$u = (1, v_{n-2}, v_{n-3}, \cdots, v_2, 1),$$

consisting of all but the last term of $v$. Then $p_u(x)$, the polynomial form of $u$, is of degree $n-2$, and the residue class ring $R = GF(2)[x]/ < p_u(x) >$ consists of the $2^{n-2}$ polynomials in $GF(2)[x]$ of degree at most $n-3$. We define a natural one-to-one correspondence between $R(v)$ and $R$. Given the general form (11) of parity check matrix for subspaces in $R(v)$, form the vector

$$b = (a_{n-2}, a_{n-3}, \cdots, a_2, a_1) + (1, v_{n-2}, \cdots, v_3, v_2).$$

Writing the entries of $b$ as

$$b = (b_{n-3}, b_{n-2}, \cdots, b_1, b_0),$$

the polynomial $p_b(x) = \sum_{i=0}^{n-3} b_i x^i$ is the element of $R$ corresponding to (11). Using the one-to-one correspondence just defined, we can carry over addition/multiplication in $R$ to addition/multiplication in $R(v)$, making $R(v)$ into a ring.

*Example 5.* In the Type 2 ring $R(1001010)$, the zero element $O$ and the identity element $I$ are given by

$$\begin{aligned} H(O) &= \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}, \\ H(I) &= \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}. \end{aligned}$$

### *C. Type 3 Rings*

A Type 3 sequence is a binary sequence that starts with 01 and ends with 10. Let $v \in V_n$ be a Type 3 sequence. Then we will define a ring $R(v)$ called a Type 3 ring. $R(v)$ will consist of the $2^{n-3}$ subspaces of $V_n$ of codimension two such that the corresponding parity check matrices each satisfy the properties

(1): $v$ is a row of the parity check matrix.

(2): The first two columns of the parity check matrix are linearly independent.

(3): The last two columns of the parity check matrix are linearly independent.

Letting $v = (0, 1, v_{n-3}, \cdots, v_2, 1, 0)$, we see that the general form of parity check matrix for a subspace in $R(v)$ is

$$\begin{bmatrix} 0 & 1 & v_{n-3} & \cdots & v_2 & 1 & 0 \\ 1 & 0 & a_{n-3} & \cdots & a_2 & a_1 & 1 \\ 1 & 1 & v_{n-3}+a_{n-3} & \cdots & v_2+a_2 & 1+a_1 & 1 \end{bmatrix}. \tag{12}$$

In the preceding, $a_1, a_2, \cdots, a_{n-3}$ are arbitrary binary parameters. Let

$$u = (1, v_{n-3}, v_{n-4}, \cdots, v_3, v_2, 1),$$

the sequence we obtain from $v$ by removing the first and last coordinates from $v$. The polynomial form $p_u(x)$ of $u$ is of degree $n-3$, and the elements of the residue class ring $R = GF(2)[x]/ < p_u(x) >$ are the $2^{n-3}$ polynomials in $GF(2)[x]$ of degree at most $n-4$. We define a natural one-to-one correspondence between $R(v)$ and $R$. Starting with (12), the general form of parity check matrix for subspaces in $R(v)$, the corresponding member of $R$ is simply the polynomial form of the vector $(a_{n-3}, a_{n-4}, \cdots, a_1)$. Using this correspondence, we carry addition/multiplication in $R$ over to addition/multiplication on $R(v)$. This makes $R(v)$ into a ring.

## V. Trellis Code Properties Via Subspace Rings

In the design of communication trellises, various properties of trellis codes are commonly employed as restrictions on the type of trellis code that the designer must choose from. In this our concluding section, we present two such properties which each have a nice characterization in terms of subspace rings:

- The property that a trellis code be *systematic*.
- The property that a trellis code satisfy the *set-partitioning principle*.

*Definition.* We define a trellis code to be *systematic* if and only if the two leftmost rows of its parity check matrix are linearly independent and the two rightmost rows of its parity check matrix are linearly independent.

**Theorem 1.** *A trellis code belongs to at least one subspace ring if and only if the left and right column of the parity check matrix are both nonzero vectors. A trellis code belongs to exactly three subspace rings if and only if it is systematic.*

*Example 6.* The systematic code on the 128-state trellis whose parity check matrix is

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{13}$$

belongs to the three subspace rings $R(11111011)$, $R(10110101)$, $R(01001110)$.

*Definition.* We say that a trellis code for a $2^k$-state trellis satisfies the set-partitioning principle if and only if there is only one $k$-edge path in the trellis whose $k$ edge labels are all equal to $a$.

*Example 7.* For the 8-state trellises in Fig. 2, let us number the eight states $1, 2, 3, 4, 5, 6, 7, 8$ from the top down. Then for each of the two trellis codes in Fig. 2, the only 3-edge path with edge labels $a, a, a$ is the path

$$1 \to 1 \to 1 \to 1. \tag{14}$$

Therefore, each of these codes satisfies the set-partitioning principle. On the other hand, consider the 8-state trellis code whose labels on the 16 edges in Fig. 2 (top down) are $abcdbadccdabdcba$. Then the path

$$8 \to 8 \to 8 \to 8$$

has edge labels $a, a, a$ and so does the path (14); therefore, this trellis code does not satisfy the set-partitioning principle.

*Definition.* We say that a code in a subspace ring $R(a)$ is a *unit* in $R(a)$ if and only if it possesses a multiplicative inverse. The set of units in $R(a)$ forms a multiplicative group (which is sometimes a cyclic group).

**Theorem 2.** *A trellis code satisfies the set-partitioning principle if it belongs to a subspace ring in which it is a unit. Conversely, if a trellis code obeys the set-partitioning principle, it belongs to at least one subspace ring and is a unit in each subspace ring containing it.*

*Example 8.* Consider the systematic code $\beta$ whose parity check matrix is (13). We show that $\beta$ satisfies the set-partitioning principle by showing that $\beta$ is a unit in the ring $R(11111011)$. From the factorization

$$p(x) = x^7+x^6+x^5+x^4+x^3+x+1 = (x^2+x+1)(x^5+x^2+1),$$

it follows that the ring $R(11111011)$ is isomorphic to the direct product ring $R(111) \times R(100101)$. $R(111)$ is the Galois field $GF(4)$ and $R(100101)$ is the Galois field $GF(32)$. Since $GF(4)$ possesses 3 units and $GF(32)$ possesses 31 units, it follows that $R(11111011)$ possesses $93 = 3 * 31$ units. This set of units forms a cyclic group $G$ whose generator is the code $\alpha \in R(11111011)$ corresponding to the polynomial $x \in GF(2)[x]/ < p(x) >$. The elements of $G$ consist of all powers $\alpha^i$, $i = 0, 1, 2, \cdots, 92$. With the aid of a computer, one can show that $\beta$ is the unit $\alpha^{33}$. (The multiplicative inverse of $\beta$ is $\alpha^{93-33} = \alpha^{60}$.) Similarly, one can also show that $\beta$ is a unit in the ring $R(10110101)$ and a unit in the ring $R(01001110)$. Indeed, any systematic code satisfying the set-partitioning principle (such as $\beta$) will be a unit in each of the three subspace rings to which it belongs.

**Final Remark.** It has been our purpose in this paper merely to introduce the reader to the new concept of subspace ring. In a subsequent paper, we will push further, designing communication trellises using subspace rings as a powerful tool.

## References

[1] R. E. Blahut, *Algebraic Codes for Data Transmission.* Cambridge University Press, Cambridge, United Kingdom, 2003.

[2] E. Biglieri, D. Divsalar, P. McLane, and M. Simon, *Introduction to Trellis-Coded Modulation with Applications.* MacMillan Publishng Company, New York NY, 1991.

[3] J. Kieffer, "Perturbations in Optimal Trellis Source Code Design," *Proc. 2005 IEEE Intl. Symp. Inform. Theory (Adelaide, Australia).*

[4] D. S. Taubman and M. W. Marcellin, *JPEG 2000: Image Compression Fundamentals, Standards and Practice.* Kluwer Academic Publishers, Boston, MA, 2002.