4×4 Perfect Space-Time Code Partition

Mireille Sarkiss , Ghaya Rekaya-Ben Othman and Jean-Claude Belfiore
École Nationale Supérieure des Télécommunications
46 Rue Barrault 75013 Paris France
Email:{sarkiss, rekaya, belfiore}@enst.fr

***Abstract*—In this paper, we partition the 4×4 Perfect Code in order to increase the minimum determinant between codewords, and hence improve the performance in slow fading channels. Using Construction A at the encoder, we construct the lattices corresponding to Perfect Code subcodes at different levels of the partition chain, and we find the convenient binary codes. This partition is made in the same manner as [3] for Golden Code. We propose also a new decoder, based on Stack decoder, which is valid for partitioning schemes. It achieves ML performance with significantly reduced complexity. Suboptimal versions are presented to reduce furthermore the complexity.**

I. Introduction

Perfect Codes were introduced for uncoded MIMO transmission in [1], [2] as full-rate, full-diversity, information lossless codes achieving diversity-multiplexing tradeoff. Thereafter, Golden space-time coded modulation were presented over slow fading channels by concatenating the Golden Code with an outer channel code [3], [4]. Since the minimum determinant of the Golden Code is constant, set partitioning is used to increase the minimum determinant then to design the trellis coded scheme in [3]. While in [4], it is used to increase the minimum hamming distances between codewords using Reed Solomon code. Motivated by the interesting gains, we apply in this paper the partitioning scheme of [3] on the 4×4 Perfect Code.

II. System Model

We consider a MIMO transmission system with $N_t = 4$ transmit antennas and $N_r = 4$ receive antennas over a slow fading channel where the channel coefficients remain constant over a frame of L codewords. Using the 4×4 Perfect Code for space-time coding, at each time slot, the complex received signal is defined by

$$\mathbf{Y} = \mathbf{HX} + \mathbf{N}; \quad \mathbf{Y}, \mathbf{X}, \mathbf{N} \in \mathbb{C}^{4\times4}. \tag{1}$$

$\mathbf{N}$ is the i.i.d. additive white gaussian noise $\sim \mathcal{N}_{\mathbb{C}}(0, N_0)$, and $\mathbf{H}$ the complex channel matrix with i.i.d. gaussian variables $\sim \mathcal{N}_{\mathbb{C}}(0, 1)$. The channel is assumed to be perfectly known at the receiver. The transmitted 4×4 signal matrices $\mathbf{X}$ contain 16 symbols and can be selected according to two options:

1) $\mathbf{X}$ are independent codeword matrices of the 4×4 uncoded Perfect Code $\mathcal{P}$.
2) $\mathbf{X}$ are independent codeword matrices chosen from a linear subcode $\mathcal{P}_k$ of the Perfect Code.

First, we represent the "Uncoded Perfect Code $\mathcal{P}$" system model. In this case, the codeword matrices $\mathbf{X}$ are linear combination of 16 information symbols $s_1, s_2, \ldots, s_{16}$ carved from Q-QAM constellation [2]. The minimum determinant of the finite Perfect Code $\mathcal{P}$ is given by

$$\delta_{min}(\mathcal{P}) = \min_{\mathbf{X}\neq\mathbf{0},\mathbf{X}\in\mathcal{P}} |\det(\mathbf{XX}^\dagger)|, \tag{2}$$

where $[.]^\dagger$ represents the hermitian operator.

In order to derive the real lattice representation, the codeword matrix is first vectorized then the real and imaginary parts are separated to obtain the real 32×32 rotation matrix $\mathbf{R}$. Thus, the Perfect Code can be seen as a rotated algebraic lattice of dimension $n = 2N_tT = 2N_rT = 32$ and the system model is redefined as following

$$\mathbf{R}\mathbb{Z}^{32} = \{\mathbf{x}; \mathbf{x} = \mathbf{Rs}, \mathbf{s} \in \mathbb{Z}^{32}\}, \tag{3}$$

$$\mathbf{y} = \mathcal{H}\mathbf{x} + \mathbf{n} = \mathcal{H}\mathbf{Rs} + \mathbf{n}, \tag{4}$$

where the real-valued vectors of $\mathbf{s}, \mathbf{y}, \mathbf{x}$ and $\mathbf{n}$ are obtained by applying $[\Re(v_i), \Im(v_i)], i = 1, \ldots, 16$ on each component v_i of the corresponding complex vector. The real channel matrix $\mathcal{H}$ is given by block diagonal of 4 real matrices $\mathcal{H}'$. $\mathcal{H}'$ is obtained by applying

$$\begin{bmatrix} \Re(h_{ij}) & -\Im(h_{ij}) \\ \Im(h_{ij}) & \Re(h_{ij}) \end{bmatrix} \tag{5}$$

on each coefficient h_{ij} of the complex channel matrix $\mathbf{H}$.

Decoding this lattice is then based on the search for the closest lattice point $\hat{\mathbf{s}}$ that minimizes the euclidian distance to the received vector $\mathbf{y}$, *i.e.,*

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s}\in\mathbb{Z}^{32}} \|\mathbf{y} - \mathcal{H}\mathbf{Rs}\|^2. \tag{6}$$

III. 4×4 Perfect Code Partition

An approach to increase the coding gain over the uncoded scheme can be achieved by partitioning the 4×4 Perfect Code which increases the minimum determinant between codewords. This technique consists first in choosing a good ideal to obtain the perfect subcodes, then identifying the lattices and the binary codes corresponding to these subcodes using construction A and lattice set partitioning by coset codes [5].

A. Choice of the Ideal

The 4×4 Perfect Code was constructed using the cyclic division algebra $\mathcal{A}(K/\mathbb{Q}(i), \sigma, i)$ of order 4, where $K = \mathbb{Q}(i, 2\cos(2\pi/15))$ [2] such that

$$\mathcal{A} = 1 \cdot K \oplus e \cdot K \oplus e^2 \cdot K \oplus e^3 \cdot K, \quad e^4 = i. \tag{7}$$

978-1-4244-2941-7/08/$25.00 ©2008 IEEE

In order to perform binary set partitioning, we need an ideal $\mathcal{I}_{\mathcal{A}}$ of $\mathcal{A}$ whose index in $\mathcal{P}$ is a power of 2 and norm is $1+i$. Note that the algebra $\mathcal{A}$ is seen as an extension of order 4 over K with minimal polynomial $X^4 - i$ as $e^4 = i$. But this cyclic algebra is not commutative and thus it will be difficult to find the ideal. So, let us consider the field extension $\mathbb{L} = \mathbb{Q}(\zeta_{16})$ of order 4 over $\mathbb{Q}(i)$ with $\zeta_{16} = \exp(\frac{i\pi}{8})$ is a primitive 16-th root of unity and $X^4 - i$ is the minimal polynomial. Using the software KANT, we factorize the prime $2\mathcal{O}_{\mathbb{L}}$, where $\mathcal{O}_{\mathbb{L}}$ is the ring of integers of $\mathbb{L}$. We find $2\mathcal{O}_{\mathbb{L}} = \mathcal{I}^4$ where $\mathcal{I}$ is a principal ideal generated by $\nu = 1 + \zeta_{16}^3$ whose norm relative to $\mathbb{Q}(i)$ is $1+i$. To construct the principal ideal of $\mathcal{A}$, we map the elements of $\mathbb{L}$ to $\mathcal{A}$ by replacing ζ_{16} with e. Thus, we get the principal ideal $\mathcal{I}_{\mathcal{A}}$ generated by $\mathbf{B} = 1 + e^3$ with reduced norm $N_{\mathrm{red}}(\mathbf{B}) = 1 + i$ and represented by the matrix

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ i & 1 & 0 & 0 \\ 0 & i & 1 & 0 \\ 0 & 0 & i & 1 \end{bmatrix} \quad \text{with} \quad e = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ i & 0 & 0 & 0 \end{bmatrix}. \quad (8)$$

B. Perfect Subcodes and Lattices Identification

The subcodes $\mathcal{P}_k$ are obtained as right principal ideal of $\mathcal{P}$ where k is the partition level

$$\mathcal{P}_k = \{\mathbf{X}\mathbf{B}^k, \mathbf{X} \in \mathcal{P}\} \quad \text{and} \quad \mathcal{P}_k \subseteq \mathcal{P}. \quad (9)$$

Since the Perfect Code $\mathcal{P}$ has been previously identified as a rotated lattice of $\mathbb{Z}^{32}$, then all the perfect subcodes $\mathcal{P}_k$ correspond to sublattices Λ_k of $\mathbb{Z}^{32}$ and their minimum determinant will be $\Delta_{\min} = 2^k \delta_{min}$ as $\det(\mathbf{B}) = 1 + i$.

In addition, $\mathbf{B}$ is of index 16 over $\mathbb{Z}^{32}$ $\left(\mathbb{Z}^{32} = \mathbb{Z}[i]^{16}\right)$, so we have 16 cosets *i.e.,* 16 ways of partition between any consecutive lattices. Thus, we need 4 bits to label these cosets. Considering the powers of the ideal, we notice that $\mathbf{B}^8 = 2I_4$ after lattice reduction, then $\mathcal{P}_8 = 2\mathcal{P}$ is the scaled Perfect Code that corresponds to $2\mathbb{Z}^{32}$. Therefore, we have $k = 8$ levels to complete the partition chain from $\mathbb{Z}^{32}$ to $2\mathbb{Z}^{32}$.

Using Construction A [5], any integer lattice can be related to a linear binary code so that any sublattice can be written as

$$\Lambda_k = 2\mathbb{Z}^{32} + C_k. \quad (10)$$

In order to identify these sublattices in the partition chain, we vectorize the subcodes matrices (vec) and we proceed to real and imaginary parts separation $(real)$ as for the Perfect Code. We obtain

$$\mathbf{R}_k \mathbf{s} = real\left(vec\left(\mathbf{X}\mathbf{B}^k\right)\right). \quad (11)$$

The sublattices generator matrices are obtained as $\mathbf{T}_k = \mathbf{R}^T \mathbf{R}_k$ with the matrices $\mathbf{T}_k$ obtained after lattice reduction. Then, the lattices are identified according to their Gram matrices [5] since

$$\det(\Lambda_k) = \det(\mathrm{Gram}(\mathbf{T}_k)) = \det\left(\mathbf{T}\mathbf{T}^T\right), \quad (12)$$

and by finding the corresponding binary codes $C_k(n, k', d)$ that need 4 additional redundant bits at each level of partition to label the 16 cosets so that $k' = n - 4k$. We present in Table I the partition chain of the 4×4 Perfect Code with the identified lattices and binary codes. The Perfect Code corresponds to the lattice $\mathbb{Z}^{32}$ that is related to the universe code $C_0(32, 32, 1)$.

TABLE I
PARTITION CHAIN OF THE 4×4 PERFECT CODE

k	PC subcode	Lattice	Binary code	$\Delta_{\min}$
0	$\mathcal{P}$	$\mathbb{Z}^{32}$	$C_0(32, 32, 1)$	δ_{min}
1	$\mathcal{P}_1$	D_8^4	$C_1(32, 28, 2)$	$2\delta_{min}$
2	$\mathcal{P}_2$	D_4^8	$C_2(32, 24, 2)$	$4\delta_{min}$
3	$\mathcal{P}_3$	$D_4^6 L_8$	$C_3(32, 20, 2)$	$8\delta_{min}$
4	$\mathcal{P}_4$	E_8^4	$C_4(32, 16, 4)$	$16\delta_{min}$
5	$\mathcal{P}_5$	$D_4^2 L_8^3$	$C_5(32, 12, 4)$	$32\delta_{min}$
6	$\mathcal{P}_6$	L_8^4	$C_6(32, 8, 4)$	$64\delta_{min}$
7	$\mathcal{P}_7$	L_{32}	$C_7(32, 4, 8)$	$128\delta_{min}$
8	$\mathcal{P}_8$	$2\mathbb{Z}^{32}$	$C_8(32, 0, \infty)$	$256\delta_{min}$

In the sequel, based on the useful binary lattices in [5], we give an exemple to explain the identification of the sublattice Λ_4 corresponding to the subcode $\mathcal{P}_4$ for the partition level $k = 4$. At this level, $\mathcal{P}_4 = \{\mathbf{X}\mathbf{B}^4\}$ and the sublattice Λ_4 has a determinant

$$\det(\Lambda_4) = \det(\mathrm{Gram}(\mathbf{T}_4)) = 2^{32} = 256^4$$

The Gosset lattice E_8, the densest eight-dimensional lattice, has determinant $\det(E_8) = 256$ and is related to the extended Hamming code or Reed-Muller code $RM(1,3) = C(8,4,4)$ as

$$E_8 = 2\mathbb{Z}^8 + C(8, 4, 4).$$

On the other hand, for $k = 4$, we need $4k = 16$ bits to label the cosets of Λ_4 and thus the binary code will be as $C_2(32, 16, d = ?)$. As a result, the lattice Λ_4 can be seen as the direct sum of four Gosset lattices E_8 with the binary code $C_4(32, 16, 4)$ the direct sum of four codes $C(8, 4, 4)$. This yields

$$\Lambda_4 = E_8^4 = 2\mathbb{Z}^{32} + C_4(32, 16, 4).$$

Since the binary code C_4 is obtained as direct sum of four codes, using the generator matrices of these component codes, we derive the 32-dimensonal generator matrix G_4 as block diagonal of them. Therefore, we have

$$G_{(8,4,4)} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$G_4 = \begin{bmatrix} G_{(8,4,4)} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & G_{(8,4,4)} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & G_{(8,4,4)} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & G_{(8,4,4)} \end{bmatrix}.$$

By proceeding similarly, we obtain the partition chain of the Perfect Code $\mathcal{P}/\mathcal{P}_1/\ldots/\mathcal{P}_8$ that corresponds to a partition chain of the lattices $\mathbb{Z}^{32}/D_8^4/\dots/2\mathbb{Z}^{32}$ related to a partition chain of convenient binary codes $C_0/C_1/\dots/C_8$. The concept of partition chain stems from the fact that these sequences of codes and lattices can be considered as sequences of additive groups where each group is a subgroup of the previous one. Thus, the sequences are said to be nested yielding 8-level partition chain where each code and each lattice is subcode and sublattice of the previous one, respectively.

$$\mathcal{P} \supseteq \mathcal{P}_1 \supseteq \mathcal{P}_2 \supseteq \mathcal{P}_3 \supseteq \mathcal{P}_4 \supseteq \mathcal{P}_5 \supseteq \mathcal{P}_6 \supseteq \mathcal{P}_7 \supseteq \mathcal{P}_8$$
$$\mathbb{Z}^{32} \supseteq D_8^4 \supseteq D_4^8 \supseteq D_4^6 L_3^2 \supseteq E_8^4 \supseteq D_4^2 L_8^3 \supseteq L_8^4 \supseteq L_{32} \supseteq 2\mathbb{Z}^{32}$$
$$C_0 \supseteq C_1 \supseteq C_2 \supseteq C_3 \supseteq C_4 \supseteq C_5 \supseteq C_6 \supseteq C_7 \supseteq C_8.$$

IV. Partition Encoder

According to the theory of coset(lattice) codes, considering the integer binary lattices in the Perfect Code partition $\mathbb{Z}^{32}/\Lambda_k/2\mathbb{Z}^{32}$, Construction A states that the lattice points of Λ_k lie within the sequences of cosets of $2\mathbb{Z}^{32}$ that are specified by the linear binary codewords of C_k

$$\Lambda_k = 2\mathbb{Z}^{32} + C_k \Rightarrow \mathbf{s} = 2\mathbf{u} + \mathbf{c}. \quad (13)$$

The lattices Λ_k are infinite; but in a transmission system, Construction A bit-labels the lattice points within finite constellations $\Lambda_k \cap \mathcal{B}$ carved from these infinite lattices with bounding region $\mathcal{B}$ of Q-QAM constellations. So that, any constellation point $\mathbf{s} \in \Lambda_k \cap \mathcal{B}$ is a 32-dimensional vector of information bits b_i, divided into two parts $\mathbf{b}_1$ used for $2\mathbf{u} \in 2\mathbb{Z}^{32} \cap \mathcal{B}$ and $\mathbf{b}_2$ used for the binary codeword of C_k as $\mathbf{c} = \mathbf{b}_2 G_k$. Then, the sublattice encoder of any perfect subcode $\mathcal{P}_k$ as presented in Figure 1 is decomposed into:

- Coded bits of $\mathbf{c}$ that select one of the codewords of C_k.
- Uncoded bits of $\mathbf{u}$ that label the $2\mathbb{Z}^{32} \cap \mathcal{B}$ points.

The 32-dimensional vector $\mathbf{u} = [u_0, \dots, u_{32}]^T$ has integer components whereas the binary codeword $\mathbf{c} = [c_0, \dots, c_{32}]^T$ has binary components $c_i \in GF(2)$ and should be lifted to integers. They are then added modulo 2 to form the lattice point $\mathbf{s}$ that is mapped to the perfect codewords.

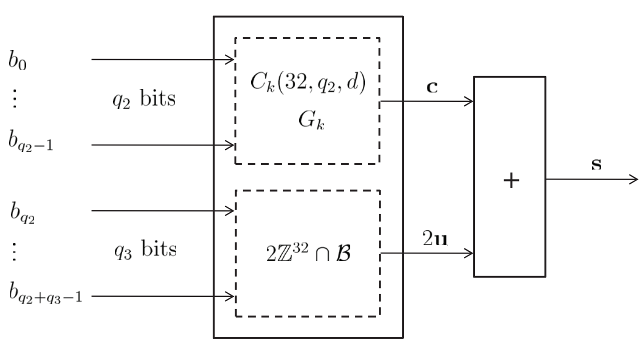

Fig. 1. Sublattice Encoder

Given that the constellation point $\mathbf{s}$ carries 16 information symbols in 4×4 transmission scheme, we assume that it is labeled by $16q$ bits corresponding to q bits per Q-QAM symbol, and hence $\mathcal{B} = \mathcal{B}_{\mathrm{QAM}}^{16}$. Therefore, we can divide this bit label into q_2 bits used to label the 2^{q_2} codewords of C_k and q_3 bits used to label the uncoded bits of $2\mathbf{u}$.

In the partition $\Lambda_k/2\mathbb{Z}^{32}$, Λ_k is spanned into $N_c = |\Lambda_k/2\mathbb{Z}^{32}|$ cosets of $2\mathbb{Z}^{32}$ that correspond to the $2^{k'}$ codewords of $C_k(n, k', d)$. We have mentioned earlier that to label 16 cosets between consecutive lattices we need 4 redundant bits. Consequently, the total redundancy of the partition is $r(\Lambda_k) = 4k$, then $k' = n - 4k$ and $q_2 = k' = 32 - 4k$.

In addition, to label the uncoded bits of $\mathbf{u}$, we note first that q_3 depends on the constellation size of $\mathcal{B}$. Since we have 2^q-QAM symbols, there are $Q/16 = 2^{q-2}$ distinct points of them that correspond to the pairs $(2u_{2i}, 2u_{2i+1}) \in \mathcal{B}_{\mathrm{QAM}}, i = 0, \dots 15$, as the codeword vector $\mathbf{c}$ consists in 2-bits pairs. Then, we need $q_3 = 16(q-2)$ bits to label the vector $2\mathbf{u} \in 2\mathbb{Z}^{32} \cap \mathcal{B}$.

Recalling the exemple of $k = 4$, the perfect subcode $\mathcal{P}_4$ corresponds to sublattice E_8^4 and binary code $C_4(32, 16, 4)$. The partition encoder of $\mathcal{P}_4$ consists in labeling the symbols $\mathbf{s} \in E_8^4 \cap \mathcal{B}_{\mathrm{QAM}}$. Assuming 16-QAM modulation $(q = 4)$, we use

- $q_2 = 32 - 4 \times 4 = 16$ bits to select the codeword $\mathbf{c}$ of C_4 as $\mathbf{c} = \mathbf{b}_2 G_4$
- $q_3 = 16(4-2) = 32$ bits to label $2\mathbf{u}$ in $2\mathbb{Z}_{32} \cap \mathcal{B}$

Therefore, we have $q_2 + q_3 = 48$ information bits encoded with E_8^4 to obtain the 32-dimensional information symbol vector $\mathbf{s}$.

Due to this bit labeling, we get the constellation points $\mathbf{s} = 2\mathbf{u} + \mathbf{c}, \mathbf{s} \in \Lambda_k \cap \mathcal{B}$ of a perfect subcode $\mathcal{P}_k$ and accordingly, the system model of the partitioned scheme is expressed

$$\mathbf{y} = \mathcal{H}\mathbf{R}\mathbf{s} + \mathbf{n} = \mathcal{H}\mathbf{R}\,(2\mathbf{u} + \mathbf{c}) + \mathbf{n}. \quad (14)$$

V. New Partition Stack Decoder

A decoding scheme which turns to be optimal in [3] is to use the Sphere Decoder (SD) to minimize the $N_c = |\Lambda_k/2\mathbb{Z}^{32}|$ squared euclidean distances in each coset *i.e.,* to find for each $\mathbf{c}$ of the $2^{k'}$ codewords of C_k the optimal $\mathbf{u}$, then, to make a decision on $\mathbf{u}$ with minimal distance

$$\mathbf{y}_j^{'} = \mathbf{y} - \mathcal{H}\mathbf{R}\mathbf{c}_j\,,\; j = 1, \dots, N_c, \quad (15)$$
$$\hat{\mathbf{u}} = \arg\min_j \Big(\min_{\mathbf{u}_j \in \mathbb{Z}^{32}} \parallel \mathbf{y}_j^{'} - 2\mathcal{H}\mathbf{R}\mathbf{u}_j \parallel^2 \Big). \quad (16)$$

However, the problem with SD is that it becomes excessively complex when large number of antennas and higher modulation size are involved. Besides, performing N_c SD decoders is too complex for Golden partition and becomes infeasible for 4×4 partition. To reduce this complexity, we propose a new optimal approach for decoding partitioned schemes with any dimension. It consists of jointly decoding the vectors $\mathbf{u}$ and $\mathbf{c}$ by applying the Stack sequential algorithm [6]. This algorithm lies in a tree search to find the best path in terms of distance measure while using an ordered list of paths referred to as stack.

A. Decoder Description

In order to expose the tree structure, QR decomposition is performed on the combined channel and lattice matrix $\mathcal{H}\mathbf{R}$ so that, the received signal can be written as

$$\mathbf{y}_1 = \mathbf{Q}_1^T \mathbf{y} = \mathbf{R}_1(2\mathbf{u}+\mathbf{c}) + \mathbf{Q}_1^T \mathbf{n} = \mathbf{R}_1(2\mathbf{u}+\mathbf{c}) + \mathbf{n}_1 \quad (17)$$

We obtain $\mathbf{R}_1$ an upper triangular matrix, thus a tree search of $n = 32$ levels can be used to solve the decoding problem. It begins at level $i = n$ and aims to find the leaf node at the last level that has the least squared euclidean distance. We define for this Stack decoder

- $\mathfrak{L}$ as the list length of the stack
- $\mathfrak{D}_i$ at level i as the euclidean distance between the received vector $\mathbf{y}_1^i = (y_1^i, \ldots y_{n-1}^i)$ and the lattice point corresponding to the node $(\mathbf{u}_i, \mathbf{c}_i)$ in the tree
- parent node at level i as the node $(\mathbf{u}_i, \mathbf{c}_i)$ whose path $\big((u_i, \ldots u_{n-1}), (c_i, \ldots c_{n-1})\big)$ in the tree gives the minimal distance and thus is positioned at the top of the stack since the stack is sorted in ascending order of the euclidean distance $\mathfrak{D}$
- child node j at level $i-1$ as the node $(\mathbf{u}_{i-1}^j, \mathbf{c}_{i-1}^j)$ generated by the parent node $(\mathbf{u}_i, \mathbf{c}_i)$ and that has the path $\big((u_{i-1}^j, u_i, \ldots u_{n-1}), (c_{i-1}^j, c_i, \ldots c_{n-1})\big)$

The search process starts with a Stack list containing a parent node whose path is empty and distance is set to zero $\mathfrak{D}_n = 0$. At level i of the tree, child nodes $(\mathbf{u}_{i-1}^j, \mathbf{c}_{i-1}^j)$ are generated from the parent node $(\mathbf{u}_i, \mathbf{c}_i)$ and the only survivors are those whose distance $\mathfrak{D}_{i-1}^j$ is lower than a predefined threshold $\mathfrak{D}_{\min}$. So that the extended parent node is eliminated from the stack and replaced by its surviving children. Then, the Stack list is sorted yielding a new parent node at the top. The process ends when a leaf node $(i = 0)$ reaches the top of the stack meaning that all intermediate nodes have higher distances than this leaf node.

We also consider systematic structure of binary codes $C_k(n, k', d)$ in the partitioned schemes, thus the stack computational complexity is reduced since couples (u, c) are generated until level $n - k'$ in the tree. For the remaining levels, child nodes are generated with only u_i varying and c_i calculated from the previous codeword components as

$$\begin{aligned} \mathbf{c} &= \Big[\big[\underbrace{c_0, \ldots, c_{n-k'-1}}_{k' \text{ coded bits}}, \underbrace{c_{n-k'} \ldots c_{n-1}}_{k' \text{ info bits}}\big]\Big]^T \\ &= \Big[\big[\underbrace{b_0 \ldots b_{k'-1}}_{k' \text{ info bits}}\big]\big[\mathbf{P}_{k' \times (n-k')} \big| \mathbf{I}_{k' \times k'}\big]\Big]^T. \end{aligned} \quad (18)$$

B. Accelerated Stack Decoding

The initial threshold $\mathfrak{D}_{\min}$ is determined before the tree search by applying a simple DFE detection on $\mathbf{y}_1$ and is updated during the Stack search every time a leaf node is reached. In addition, in order to approach the optimal solution and accelerate the tree search, we use the Simulated Annealing (SA) algorithm when the stack list reaches a given length $\mathfrak{L} = \mathfrak{L}_{\text{SA}}$. Simulated Annealing is a probabilistic algorithm that provides good solution for complex optimization problem in a fixed amount of time by trying random variations of the current solution while avoiding local minima. In our case, SA allows to optimize the initial solution obtained by the DFE detector in order to get a tighter bound on $\mathfrak{D}$. The SA is initiated when the stack is filled with $\mathfrak{L}_{\text{SA}}$ nodes and only if the search process has not yet attained a leaf node. Therefore, a new threshold $\mathfrak{D}_{\min}$ is updated in an acceptable duration by choosing conveniently the length $\mathfrak{L}_{\text{SA}}$ and the SA parameters. Then all the tree branches with higher distance $\mathfrak{D} > \mathfrak{D}_{min}$ are removed allowing to reduce the Stack list, hence the complexity of the subsequent search phase.

C. Suboptimal Stack Decoding

Until now, all the procedures to speed up the tree search do not affect the decoder optimality. However, suboptimal version can be derived when the Stack list length is fixed. In fact, limiting the stack intends to keep only the nodes with the lowest distances, and to neglect the other paths in order to free positions for the new inserted nodes. But, the problem is that the neglected paths may include the correct solution leading to an error in the decoding. Note that the limit on the stack storage can be done either on the overall list length $\mathfrak{L}_{\max}$ or by restricting the lists on each level of the tree $\mathfrak{L}_{\max_i}$. The parameters can be chosen appropriately in order to compromise between complexity and performance of the suboptimal decoding scheme.

VI. Simulation Results

In this section, we evaluate the performance of the 4×4 Perfect Code and Golden Code partitions compared to the uncoded systems using the proposed Stack decoder. In order to have a valid comparison between both schemes, we should maintain the same spectral efficiency. However, a rate loss is induced in the partitioned schemes due to the code redundancy. To compensate for this loss, a constellation expansion is required meaning that the same number of bits is transmitted with higher energy, thus higher modulation order is used in a subcode compared to uncoded code. For instance, the constellation points of the uncoded Perfect Code $\mathcal{P}$ are QAM symbols carved from the infinite lattice $\mathbb{Z}^{32}$ such that $\mathbf{s} \in \mathbb{Z}^{32} \cap \mathcal{B}'$ whereas $\mathbf{s} \in \Lambda_k \cap \mathcal{B}$ for perfect subcodes. So, we need $\mathcal{B} \supset \mathcal{B}'$.

The asymptotic coding gain between partitioned and uncoded schemes, with the same rate but different constellation energies $E_{s,1}$, $E_{s,2}$ and different minimum determinants $\Delta_{\min,1}$, $\Delta_{\min,2}$ is defined in [2] as

$$\gamma_{as} = \frac{\sqrt[N_t]{\Delta_{\min,1}}/E_{s,1}}{\sqrt[N_t]{\Delta_{\min,2}}/E_{s,2}}. \quad (19)$$

For simulations, we use 16-QAM modulation with the subcodes and we assume that the constellations are scaled to match $\mathbb{Z}[i] + (1+i)/2$. The average energy per symbol is equal to $E_s = 0.5, 1.5, 2.5$ for $Q = 4, 8, 16$, respectively. Considering that the MIMO channel remains static during a frame length of $L = 100$ codewords, we compare the performances in terms of Codeword Error Rate (CER).

We perform four levels partition for the 4×4 code:

$$E_8^4 = 2\mathbb{Z}^{32} + (32, 16, 4), q_2 = 16 \text{ and } q_3 = 32$$

and two levels partition for the Golden Code:

$$E_8 = 2\mathbb{Z}^8 + (8, 4, 4), q_2 = 4 \text{ and } q_3 = 8$$

Preserving the same spectral efficiency for comparison, for 4×4 case, 16-QAM with energy $(E_{s,1} = 2.5)$ is used to encode $q_2 + q_3 = 48$ bits with E_8^4 while an 8-QAM is required with the Perfect Code since $16q' = 48, q' = 3$ bits, corresponding to the spectral efficiency of $\eta = 12$ bpcu and the average energy of $E_{s,2} = 1.5$. Likewise, for 2×2 case, 16-QAM is required to encode 12 bits with E_8 (4 redundant bits used for the code), while the uncoded Golden Code requires only an 8-QAM. In both cases, $2\mathbf{u}$ contains the 4-QAM symbols of $\mathcal{B}$ with integer values $\{+1, -1\}$.

Figure 2 shows that the proposed Stack decoder achieves the SD performance given in [3]. The 4×4 Perfect Code partition allows a gain of 2.2 dB at BER of 10^{-3} compared to uncoded scheme. The simulation gain is higher than the asymptotical coding gain computed as

$$\gamma_{as} = \frac{\sqrt[4]{\Delta_{\min,p}}/E_{s,1}}{\sqrt[4]{\Delta_{\min,u}}/E_{s,2}} = \frac{\sqrt[4]{16\delta_{min}}/2.5}{\sqrt[4]{\delta_{min}}/1.5} = 1.2$$

$$(\gamma_{as})_{\mathrm{dB}} = 10\log_{10}(1.2) = 0.8\mathrm{dB}.$$

This gap is related to the approximation of the Pairwise Error Probability (PEP) leading to an approximate measure of the coding gain [7]. This latter depends on the minimum determinant over all pairs of distinct codewords according to the design criteria of space-time codes [7]. However, the minimum determinant is not necessarily the most frequent between constellation points. Therefore, one should take into account not only many minumum determinants but also their numbers or probability of occurrence (multiplicities). Thus, the dominant term in the PEP could correspond to higher $\Delta_{\min}$ leading to higher gain than expected.

Figure 3 depicts the complexity of proposed Stack decoder compared to conventional SD decoder with partitioned Golden Code. The average search time of both algorithms is evaluated versus signal to noise ratio E_b/N_0 for 1 million transmitted codewords. It can be observed clearly that the Stack decoder converges much faster than SD for 2×2 dimensions. In 4×4 schemes, the 32-dimensional SD repeated 2^{16} times is impractible.

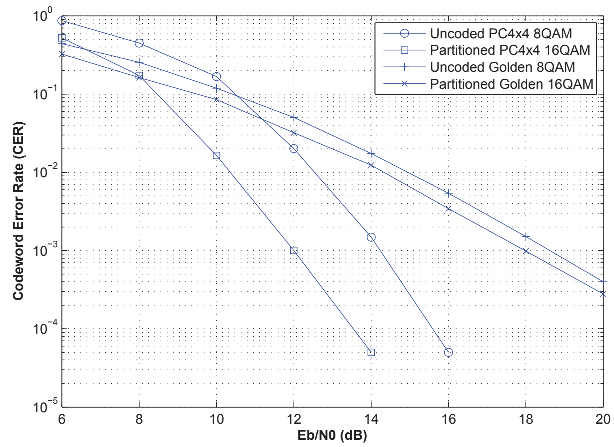

Fig. 2. Partitioned codes with 16-QAM vs uncoded codes with 8-QAM

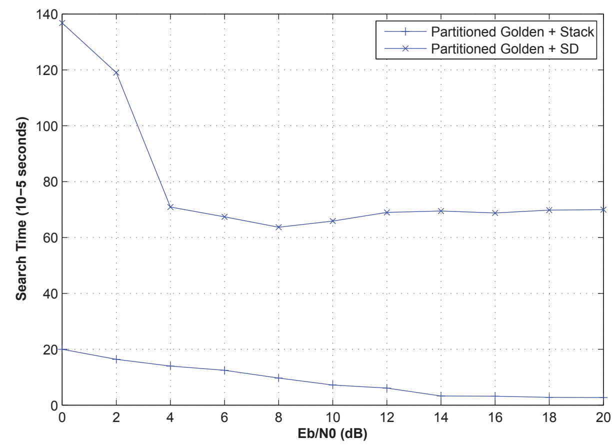

Fig. 3. Search time of Stack vs SD for Golden partition

VII. Conclusion

We investigate in this paper the 4×4 Perfect Code partition over slow fading channels. By set partitioning, we increase the minimum determinant between perfect codewords, and thus increase the coding gain and enhance the performance over the uncoded scheme. Based on coset codes theory, we use Construction A to identify the 8-level partition chain of the Perfect Code subcodes with the corresponding sublattices and binary codes. We propose a new optimal decoder for decoding partitioned schemes with any dimension based on the Stack sequential algorithm. We validate this decoder for the Golden Code partition and show that it gives the ML performance [3] while reducing considerably the complexity. For the 4×4 partition, we otain an interesting gain over the uncoded Perfect Code. Therefore, this motivates us to study in future works the Perfect Space-Time Trellis Coded Modulation design based on this partitioning scheme.

Acknowledgment

The first author would like to thank Mr. Elias Doumith for the fruitful discussions on Simulated Annealing algorithm.

References

[1] J.-C. Belfiore, G. Rekaya, and E. Viterbo, "The Golden code: a 2×2 full-rate space-time code with non-vanishing determinants", IEEE Trans. Inform. Theory, vol. 51, n. 4, 2005.
[2] F. Oggier, G. Rekaya, J-C. Belfiore, and E.Viterbo, "Perfect space-time block codes", IEEE Trans. Inform. Theory, vol. 52, n. 9, 2006.
[3] Y. Hong, E. Viterbo,and J.-C. Belfiore, "Golden space-time trellis coded modulation", IEEE Trans. Inform. Theory, vol. 53, n. 5, 2007.
[4] L. Luzzi, G. Rekaya, J.-C. Belfiore, and E. Viterbo, "Golden space-time block boded modulation", IEEE Proc. Inform. Theory Workshop, 2008.
[5] Jr. G. D. Forney, "Coset codes. I. Introduction and geometrical classification", IEEE Trans. Inform. Theory, vol. 34, no. 5, 1988.
[6] J. Felinek, "Fast sequential decoding algorithm using a stack", IBM J. Res. Develop., 1969.
[7] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction", IEEE Trans. Inform. Theory, vol. 44, no. 2, 1998.