A New Modulation Scheme for Space-Time Codes

Mehrdad Shamsi, Masoud Ardakani, Ian F. Blake
Edward S. Rogers Sr. Department of Electrical and Computer Engineering
University of Toronto

Abstract—A new modulation method for linear space-time codes is proposed based on using constellations of different sizes for different symbols. It is shown that the proposed method significantly reduces the complexity of the sphere decoding algorithm. Typically, a complexity reduction of one to two orders of magnitude can be achieved at the expense of about 3 dB coding gain. With a simple modification in our design, we reduce this loss to about 2 dB.

I. INTRODUCTION

Linear dispersion (LD) codes are among the most successful space-time code designs [1]. These code can be decoded by sphere decoding (SD) algorithm that has polynomial complexity in the number of symbols in SNR ranges of interest [2].

In this paper we show that by taking the symbols appearing in the early stages of the SD algorithm from smaller constellations and using larger constellations for the rest, one can considerably reduce the decoding complexity without losing rate or having significant performance degradation. We introduce a new modulation scheme for linear space-time codes, which we refer to as mixed-Q linear space-time codes. We also find the optimum order of symbol decoding in SD algorithm in the sense of minimizing the decoding complexity. The reduction in the search complexity can be as high as one to two orders of magnitude.

This paper is organized as follows. In Section II, we define some notations and the multiple antenna system model. In Section III, linear dispersion codes and sphere decoding algorithm are briefly reviewed. The main idea of this paper is presented in Section IV, where we define mixed-Q codes and analyze their performance and decoding complexity. Some simulation results are presented in Section V, and the paper is concluded in Section VI.

II. SYSTEM MODEL AND NOTATIONS

A. Notations

The following notations are used in this paper. The zero-mean, unit variance, circularly symmetric, complex Gaussian distribution is denoted by $CN(0,1)$. The transpose and the conjugate transpose of a matrix $A$ are denoted by $A^T$ and $A^*$ respectively. The vector form of a matrix $A$, denoted by $vec(A)$, is defined as a column vector constructed by stacking up the columns of $A$ one after another. We use $A \otimes B$ for the Kronecker product of $A$ and $B$. The $(i, j)^{th}$ element of a matrix $A$ is denoted by $a_{i,j}$. We find it more convenient to index components of a vector $\mathbf{v}$ in descending order, i.e., $\mathbf{v} = (v_n, \ldots, v_1)^T$. For a vector $\mathbf{v} \in \mathbb{C}^{n \times 1}$, its real-valued counterpart $\mathcal{R}\mathcal{E}(\mathbf{v}) \in \mathbb{R}^{2n \times 1}$ is formed by replacing the $k^{th}$ component $v_k = x_k + \sqrt{-1} y_k$ with $\begin{bmatrix} x_k \\ -y_k \end{bmatrix}$. For a matrix $A \in \mathbb{C}^{m \times n}$, we replace the $(i, j)^{th}$ element $a_{i,j} = x_{i,j} + \sqrt{-1} y_{i,j}$ with $\begin{bmatrix} x_{i,j} \\ -y_{i,j} \end{bmatrix}$ to get its real-valued counterpart $\mathcal{R}\mathcal{E}(A) \in \mathbb{R}^{2m \times 2n}$. We use a special notation for vectors with many equal components, for example $(2, 2, 2, 8, 8)$ is denoted by $(3\{2\}, 2\{8\})$.

B. System model

We consider a multiple antenna channel with $m$ transmitter and $n$ receiver antennas. At each block index, the transmitter chooses a matrix $C$ from a set of $m \times m$ complex matrices $\mathcal{C}$ and sends it. The received signal is an $m \times n$ matrix $X$. We can write the relation between the input and output of the system as $X = CG + W$, where $G$ and $W$ are in $\mathbb{C}^{m \times n}$ with $CN(0,1)$-distributed independent entries.

III. SPHERE DECODING

Suppose a stream of data bits is to be sent over $m$ transmit antennae and across $m$ channel uses. The LD encoder breaks each block of data bits to $p$ equal-length sub-block and maps each of them to a symbol taken from a constellation $\mathcal{S} \subseteq \mathbb{C}$. The resulting vector of symbols $s = (s_p, s_{p-1}, \ldots, s_1)$ is dispersed across time and antennae by sending a linear combination of the symbols, $A_p, A_{p-1}, \ldots, A_1$, with these symbols as the coefficients. Therefore the code matrices have the following form $C(s) = \sum_{i=1}^{p} \mathcal{S} A_i$. Obviously, the code performance depends on the choice of dispersion matrices $A_1, \ldots, A_p$. In [1], the mutual information between input and output of the channel is considered as a criterion for choosing $A_i$’s.

The linear structure of LD codes can be exploited to design polynomial complexity decoding algorithms in a wide range of SNR’s. Here, we reformulate the decoding problem into finding the closest point in a lattice. We rewrite the received signal equation in vector form as $vec(X) = (G^T \otimes I)Bs + vec(W)$, where $B = [vec(A_1), \ldots, vec(A_1)]$ and $s = [s_p, \ldots, s_1]^T$.

We can easily convert the complex-value equation (??) to a real-valued one to get $r = Hx + w$, where $r = \mathcal{R}\mathcal{E}(vec(X))$, $x = \mathcal{R}\mathcal{E}(s)$, $w = \mathcal{R}\mathcal{E}(vec(W))$ and $H = \mathcal{R}\mathcal{E}((G^T \otimes I)B)$. Since the noise vector $w$ is assumed to have independent $CN(0, \sigma^2)$-distributed entries, the ML decoding rule is

$$\hat{x} = \arg \min_{x} ||r - Hx||.$$  \hspace{1cm} (1)
Now, we describe the sphere decoding algorithm. We assume \(mn = p\), because for \(mn < p\) the generalized sphere decoding algorithm should be used which is exponential in \((p - mn)\), and for \(mn > p\) one can easily reduce to the case of \(mn = p\) [2]. Set \(q := 2mn = 2p\), therefore \(r \cdot w \cdot x \in \mathbb{R}^{q \times 1}\) and \(H \in \mathbb{R}^{q \times q}\). Our goal is to find a solution to (1) within the hypersphere of radius \(c\), for some constant \(c\), centered at \(r\). A lattice point is in this hypersphere if and only if \(||r - Hx||^2 \leq c^2\). First, we compute QR decomposition of \(H\).\ref{2} i.e., \(r\) is an upper triangular matrix \(R \in \mathbb{R}^{q \times q}\) and an orthogonal matrix \(Q \in \mathbb{R}^{q \times q}\) such that \(H = QR\). We have

\[
Q'r = Q'Rr + Q'w.
\]  

Let \(y := Q'r\) and \(v := Q'w\). Note that the entries of \(v\) have a Gaussian distribution with the same variance as the entries of \(w\) because \(Q'Q = I\). We can rewrite (2) as \(y = Rx + v\). Thus, \(x_1, ..., x_q\) have to satisfy the following necessary conditions in order to be in the hypersphere:

\[
d_1 := |y_1 - r_{q.q}x_1|^2 \leq c^2
\]

\[
d_2 := |y_2 - r_{q-1.q-1}x_2 - r_{q-1,q}x_1|^2 \leq c^2
\]

\vdots

\[
d_{q-1} := |y_q - r_{1,1}x_q - \cdots - r_{1,q}x_1|^2 \leq c^2.
\]

The first inequality above can be solved to find a set of possible values for \(x_1\), then by assuming a value for \(x_1\) from this set, a range of possible values for \(x_2\) will emerge. This procedure can be applied successively till one reaches the last inequality. Conceptually, this procedure forms a \((q + 1)\)-level tree with a starting node at level zero and the nodes at level \(i = 1, ..., q\) labelled by the choice of \(x_i\).

### IV. Mixed-Q space-time codes

A closer look at the necessary conditions of (3) shows that the first several inequalities impose a very loose condition of \(x_i\)'s. Notice that for the last inequality to be satisfied, \(c^2\) has to be greater than or equal to \(d_i\), the cumulative sum of distances, which can be much greater than \(d_i\)'s for small \(i\)'s. Hence, for small \(i\)'s almost all constellation symbols satisfy the loose inequalities, resulting in a search tree that expands exponentially at the first several levels. The algorithm starts pruning the tree only after the accumulated distance becomes comparable to \(c^2\).

Based on this observation, we propose to use smaller constellations (constellations with fewer number of points) for the symbols that appear at the higher levels of the tree, and larger constellations for those at the lower levels. In this way, at the higher levels where the tree expands exponentially, the rate of expansion is smaller. This results in dramatic reduction of the size of the search tree. On the other hand, the rate loss will be compensated through the symbols taken from larger constellations at the lower levels. We call this transmission scheme mixed-Q linear space-time codes, or mixed-Q codes for short, to contrast their difference with ordinary LD codes that we refer to as fixed-Q linear space-time codes.

### A. Mixed-Q codes definition

A mixed-Q encoder takes each block of data bits and breaks it into \(p\), not necessarily equal-length, sub-blocks and maps each resulting sub-block to a symbol \(s_i\) from a complex constellation \(S_i\). The encoder then computes the corresponding code matrix using

\[
C(s) = \sum_{i=1}^{p} A_i s_i,\quad s = (s_p, s_{p-1}, ..., s_1) \in S_p \times \cdots \times S_1,
\]

where \(A_i\)'s are dispersion matrices. As it can be seen from \(C(s)\), the following parameters are to be specified:

- Number of symbols per code matrix, \(p\). To allow use of the SD algorithm, \(p\) should satisfy \(p \leq mn\). To exploit the maximum degrees of freedom of the channel, we choose \(p = mn\).
- Dispersion matrices \(A_1, ..., A_p \in \mathbb{C}^{m \times m}\). We chose dispersion matrices to ensure full-diversity [3]. We also assume \(\text{tr}(AA^*) = m^2\).
- Constellations. In our design we only use rectangular or square QAM constellations.

Each code matrix carries \(p\) complex symbols and each complex symbol \(s_i \in S_i\) consists of two real symbols combined via \(s_i = x_{2i} + \sqrt{-1}x_{2i-1}\). Therefore, each code matrix carries \(q := 2p\) real symbols. In other words, each QAM constellation is the cartesian product of two PAM component constellations, which are not necessarily of the same size. The real symbol \(x_k\) comes from an \(l_k\)-PAM constellation, by which we mean the set \(Q_k = \{-l_k + 1, -l_k + 3, \ldots, l_k - 3, l_k - 1\}\).

Therefore, the \(q\)-tuple \(l = (l_1, ..., l_q)\) specifies the code. Each real symbol \(x_k \in Q_k\) carries \(b_k := \log l_k\) bits. The transmission rate is given by \(b = \frac{1}{m} \sum_{k=1}^{q} b_k\) and the average power of the code matrices, before normalization, is given by \(E_{av} = \frac{m^2}{3} \sum_{k=1}^{q} (l_k^2 - 1)\). Thus, we normalize each code matrix by \(E_{av}/m^2\) to keep their average power equal to \(m^2\).

We use \(C_m(l, q)\) to denote a mixed-Q code that encodes \(q/2\) complex symbols or equivalently \(q\) real symbols from \(l_k\)-PAM component constellations into a code matrix. For example \(C_m(8, 4, 4, 2, 4)\) means \(x_4, x_3, x_2, x_1\) are chosen from 8, 4, 4, 2-PAM constellations respectively. Similarly, we denote a fixed-Q code with \(C_f(l, q)\), e.g. \(C_f(2, 4)\) means each code matrix carries \(4\) real symbols taken from a 2-PAM constellation.

### B. Analysis of the search tree

In this section, we show that to have a \(C_m(l, q)\) with minimum decoding complexity, \(l = (l_1, ..., l_q)\) must satisfy \(l_1 \leq l_2 \leq \cdots \leq l_q\). In this case, we say the mixed-Q code is optimally ordered.

Recall that each path in the search tree of the SD algorithm represents a lattice point. Similarly, we can associate a tree to the entire lattice generated by matrix \(H\) over the component constellations \(Q_k\)'s of size \(l_k\). We call this tree a full-tree to contrast it with the pruned-tree constructed during the SD algorithm. We claim and prove a series of propositions on the size of the full-tree, and discuss the implications of

---

0-7803-8939-5/05/$20.00 (C) 2005 IEEE
these propositions on the decoding complexity of the mixed-Q codes.

First, we introduce some notations. Let \( I = (l_q, l_{q-1}, \ldots, l_1) \) be a \( q \)-tuple in \( \mathbb{N}^q \), \( T(I) \) be a tree of \( q \) levels with nodes at level \( k \) having \( l_k \) children, and \( \sigma \) be a permutation of \((1, 2, \ldots, q)\). We denote the \( q \)-tuple \((l_q, \ldots, l_1)\) by \( \sigma(I) \). The total number of nodes in tree \( T \) is denoted by \( N(T) \). Finally, \( N_k(T) \) denotes the number of tree nodes at level \( k \).

**Lemma 1:** Let \( T(I) \) be a tree with \( l_b > l_a \) for some \( a < b \in \{1, \ldots, q\} \). Swap \( l_a \) and \( l_b \) in \( I \) to get \( I' \). Then \( N(T(I')) < N(T(I)) \).

**Proof:** The number of nodes in \( T(I) \) is given by \( N(T(I)) = \sum_{k=1}^{q} \prod_{i=1}^{k} l_i \). It is easy to see that

\[
N(T(I)) - N(T(I')) = (1 - \frac{l_b}{l_a}) \sum_{i=a}^{b-1} \prod_{k=1}^{i} l_k > 0
\]

**Proposition 1:** Among all trees of the form \( T(\sigma(I)) \), where \( \sigma \) is a permutation of \((1, 2, \ldots, q)\), the one with \( l_1 \leq l_2 \leq \cdots \leq l_q \) has the minimum number of nodes.

**Proof:** Let \( T_{\text{min}} = T(l_q, \ldots, l_1) \) be the tree with minimum number of nodes. If for some \( i < j \), \( l_i > l_j \) then by Lemma 1, we can construct another tree by swapping \( l_i \) and \( l_j \) that has a smaller number of nodes, which is a contradiction.

The following proposition states that the number of nodes of the full-tree of an optimally ordered mixed-Q code is no larger than that of a fixed-Q code of the same rate.

**Proposition 2:** Let \( T_m = T(l_q, \ldots, l_1) \) and \( T_f = T(l'_q, \ldots, l'_1) \), where \( l_1 \leq l_2 \leq \cdots \leq l_q \) and \( l'_1 = \cdots = l'_{q-1} \). Then \( N(T_m) \leq N(T_f) \).

**Proof:** We have

\[
N(T_m) = \sum_{k=1}^{q} \prod_{i=1}^{k} l_i \quad \text{and} \quad N(T_f) = \sum_{k=1}^{q} \left( \prod_{i=1}^{k} l_i \right)^{k/q}
\]

Since \( l_1 \leq \cdots \leq l_q \), it is easy to see that

\[
\left( \prod_{i=1}^{k} l_i \right)^{k/q} \leq \left( \prod_{i=1}^{k} l_i \right)^{k/q}
\]

Thus \( N(T_m) \leq N(T_f) \).

**Corollary 1:** \( N_k(T_m) \leq N_k(T_f) \), for \( k = 1, \ldots, q \).

**Corollary 2:** Any weighted sum of \( N_k(T_m) \)'s with positive weights is less than the weighted sum of \( N_k(T_f) \) with the same set of weights, i.e.,

\[
\sum_{k=1}^{q} w_k N_k(T_m) \leq \sum_{k=1}^{q} w_k N_k(T_f), \quad w_k > 0 \quad \text{for} \quad k = 1, \ldots, q.
\]

Now consider a fixed-Q code \( C_f(l, q) \) and its mixed-Q counterpart \( C_m(l, q) \) of the same rate, and their associated full-trees \( T_f \) and \( T_m \), respectively. Although the SD algorithm searches over the pruned-tree, which represents the intersection of the hypersphere and the lattice, the full-tree size is a good indicator of the complexity of the search. As Corollary 1 suggests, the number of nodes at each level of the mixed-Q full-tree is less than that of the fixed-Q one. Also notice that pruning a tree can be thought as multiplying the number of nodes in each level with a weight less than one. Therefore from Corollary 2, using the same set of weights for both trees would result in a smaller pruned-tree for the mixed-Q code.

In reality, these weights depend on the transmitted codeword, constellation size, received signal, channel condition and initial radius of the hypersphere. Nevertheless, averaged over all input vectors and channel realizations one can intuitively argue that the pruned-tree of the mixed-Q code is smaller than that of the fixed-Q one, as verified in our simulation results.

Another important observation is that the dominant term in the total number of nodes is the number of nodes in the middle levels of the pruned-tree as clearly shown in Fig. 3. Notice that the number of such nodes for a mixed-Q code is much less than that of a fixed-Q one, because the rate of exponential expansion, Vardy of the mixed-Q tree at upper levels is smaller than that of fixed-Q tree.

### C. Choosing \( Q_i \)

In this section, we describe how the \( Q_i \)'s are chosen. Our argument is based on analyzing the minimum distance of constellation points. Basically, we will show that we cannot choose very large constellations, otherwise the performance degradation will be significant. For example, if in a fixed-Q code we use 4-PAM component constellation, the mixed-Q counterpart needs only 2-PAM, 4-QAM, and 8-QAM component constellations.

Recall that the coding gain of a space-time code \( C \) is \( \gamma_C = \min_{C \neq C'} \left( \prod_{i=1}^{r} \lambda_i \right)^{\frac{1}{2}} \), where \( \lambda_i \)'s are non-zero eigenvalues of \((C - C')(C - C')^* \) and \( \nu \) is its rank. For a fully-diverse LD code we can find an upper bound on the coding gain, as stated in the following proposition.

**Proposition 3:** The coding gain of a fully-diverse LD code constructed over a constellation of minimum squared distance \( d_{\text{min}}^2 \) is upper bounded by \( \gamma_C \leq m d_{\text{min}}^2 \).

**Proof:**

\[
\gamma_C = \min_{C \neq C'} \det((C - C')(C - C')^*)^{\frac{1}{2}}
\]

\[
\leq \min_{\nu, s_i} \det(|s_i - s_i'|^2 A_i A_i^*)^{\frac{1}{2}}
\]

\[
\leq \frac{d_{\text{min}}^2 \text{tr}(A_i A_i^*)}{m} = m d_{\text{min}}^2
\]

The last inequality above follows from the geometric-arithmetic means inequality, and to get the final result we assumed \( \text{tr}(A_i A_i^*) = m^2 \) as indicated in LD codes definition.

Thus, to have a good coding gain we must keep \( d_{\text{min}}^2 \) as large as possible. For a mixed-Q code \( C_m(l, q) \) and its same rate fixed-Q counterpart \( C_f(l, q) \), the minimum squared distances are given by \( d_{\text{min}}^2(C_f) = 12 q^{-1}(l^2 - 1)^{-1} \) and \( d_{\text{min}}^2(C_m) = 12 \left( \sum_{i=1}^{q} (l_i^2 - 1) \right)^{-1} \).

In the following proposition, by proving an inequality on the minimum distances, we show that the performance of a mixed-Q code is worse than its same rate fixed-Q counterpart.
Proposition 4: Let $C_f(l, q^b)$ and $C_m(l, q^b)$ be a fixed-Q and a mixed-Q code of the same rate respectively. Then

$$d_{\text{min}}^2(C_m) \leq d_{\text{min}}^2(C_f).$$

Proof: Let $b_i = \log_2 l_i$ and $b = \log_2 l$. $C_f$ and $C_m$ are of the same rate, therefore $qb = \sum_{i=1}^q b_i$. Since $f(x) = 4^x$ is a convex function, we have

$$4^b \leq \frac{1}{q} \sum_{i=1}^q 4^{b_i} \Rightarrow q(2^{2b} - 1) \leq \sum_{i=1}^q (2^{2b_i} - 1)$$

$$\Rightarrow d_{\text{min}}^2(C_m) \leq d_{\text{min}}^2(C_f)$$

The equality in (5) holds only if all $b_i$’s are equal. In other words, regardless of how we choose $Q_i$’s for a mixed-Q code, we expect performance degradation compared to a fixed-Q code of the same rate. Thus, it is reasonable to choose $b_i$’s as equal as possible. On the other hand, as we saw in the previous section mixed-Q codes have the benefit of less decoding complexity. The following two propositions shed light on the tradeoff between performance and complexity.

Proposition 5: Let $C_m(l, q^b)$ be a mixed-Q code, with $l_i = 2^{b_i}$. Let $b_{\text{min}} = \min(b_i)$ and $b_{\text{max}} = \max(b_i) > b_{\text{min}} + 1$. Construct $C'_m$ by replacing $b_{\text{min}}$ and $b_{\text{max}}$ with $b_{\text{min}} + 1$ and $b_{\text{max}} - 1$ respectively. Then $d_{\text{min}}(C'_m) > d_{\text{min}}(C_m)$ and $N(T(C'_m)) < N(T(C))$.

Therefore, among all mixed-Q codes supporting $gb$ bits per code matrix, the one with the best minimum distance is constructed over component constellations of size $2^b$, $2^{b-1}$ and $2^{b+1}$.

Notice that when only component constellations of size $2^b$, $2^{b-1}$ and $2^{b+1}$ are used, there is still a tradeoff between performance and complexity due to the number of symbols chosen from each component constellation, as the following proposition states.

Proposition 6: Let $l = (q_1 \{2^{b+1}\}, q_2 \{2^b\}, q_3 \{2^{b-1}\})$ and $I = ((q_3 - 1)\{2^{b+1}\}, (q_2 + 2)\{2^b\}, (q_1 - 1)\{2^{b-1}\})$ and construct mixed-Q codes $C = C_m(l, q^b)$ and $C' = C_m(l, q^b)$ of the same rate. Then $d_{\text{min}}(C'_m) > d_{\text{min}}(C_m)$ and $N(T(C'_m)) < N(T(C))$.

D. Performance improvement

As our analysis showed, the performance loss of mixed-Q codes is due to their smaller $d_{\text{min}}$. To address this problem, we modify the definition of mixed-Q codes to

$$C(s) = \sum_{i=1}^P a_i A_i s_i, \quad s = (s_p, s_{p-1}, \ldots, s_1) \in S_p \times \cdots \times S_1,$$

where $a_i$’s are real scaling factors.

By using $a_i$’s, we re-scale the constellations to avoid a very large average power before normalization. This way, the $d_{\text{min}}$ is less affected by the use of large constellations in a mixed-Q code. Notice that this modification does not change the number of nodes in the full-tree.

The above discussion is made to clarify our purpose of using these scaling factors. A more fundamental measure, which can be optimized using this degree of freedom, is the mutual information between transmitted and received signal as suggested in [1]. Especially for LD codes such as those in [3], where the dispersion matrices are chosen systematically to guarantee full-diversity, the optimization of the mutual information through the choice of $\alpha_i$’s can still improve the performance.

V. Simulation results

In this section, we compare the performance and the decoding complexity of mixed-Q and fixed-Q codes through some simulations. A system of $m$ transmit and $n$ receive antennas is considered. Each code matrix carries $q$ real (or equivalently $q/2$ complex) symbols. We designed mixed-Q and fixed-Q fully-diverse codes based on the method introduced in [3]. The channel matrix $G$ is generated randomly and remains fixed for a block length of $m$, and then changes independently to a new value. We used the Schnorr-Euchner variant of the SD algorithms in our simulations. The total number of nodes in the search tree $N(T)$ is computed and its average $E_{G_i}[N(T)]$ over a sufficient number of channel realizations is considered as the complexity measure. We find it more illuminating to compare the complexity exponent (CE) of different codes defined as $\log_q E_{G_i}[N(T)]$. We do not consider the complexity of the computing QR decomposition in the calculation of CE, as it is well known to be $O(q^3)$. In Fig. 1 and Fig. 2, we compare bit error rate (BER) and the complexity exponent of a fixed-Q and some mixed-Q codes with $m = 4$, $n = 3$ and $q = 24$ for different combinations of component constellations. We used 2-PAM, 4-PAM and 8-PAM component constellations (equivalent to 4-QAM, 16-QAM and 64-QAM complex constellations). As predicted by Proposition 5, the fixed-Q code $C_f(4, 24)$ has the best BER and the worst CE compared to its same rate mixed-Q counterparts $C_m(1, 24)$ for different choices of $I$. The tradeoff between complexity and performance is evident here; as we introduce more constellation disparity into the code, the decoding complexity decreases at the price of performance. Note the dramatic decrease in complexity by moving from $C_f(4, 24)$ to a $C_m(1, 24)$ where $I = \{8\{8\}, 8\{4\}, 8\{2\}\}$. For example, in this case, the complexity is reduced by a factor
of $24^1 \approx 24$ at SNR=17dB; to achieve the same complexity exponent by a fixed-Q code, the SNR should be increased to more than 30 dB.

In our simulations, we counted the number of nodes at each level of the search tree, which is denoted by $N_k$. In Fig. 3, we plot $\log(N_k)$ vs. $k$ at SNR=20dB for the same codes of Fig 2. As it can be seen for small $k$'s, these curves grow almost linearly for both mixed-Q and fixed-Q code, indicating almost exponential expansion of the tree in first several levels. However, the rate of expansion for the mixed-Q codes is smaller. As $k$ increases and we move to the lower levels of the tree, the algorithm starts pruning the tree.

Fig. 4 illustrates the effect of optimizing scaling factor $\alpha_k$'s on the performance and the complexity of the code. This optimization improves the performance of a mixed-Q code of $(8\{8\}, 8\{4\}, 8\{2\})$ in moderate to high SNR's.

Fig. 5 show the similar results for codes of parameters $m = 4$, $n = 4$, $q = 32$ and rate = 16 bits per channel use. As it can be seen the complexity reduction is more dramatic here, e.g. the mixed-Q complexity is about 100 times less than the fixed-Q at SNR=20 dB.

VI. CONCLUSION

We introduced mixed-Q modulation scheme for linear space-time codes. Through the analysis of the the search tree formed in SD algorithm, we proved that significant complexity reduction —typically one to two orders of magnitude— can be achieved at the expense of moderate performance loss, typically 3 dB. With a simple modification, we reduced the performance loss to about 2 dB.

We showed that to have the minimum decoding complexity, the decoder must decode the symbols in the increasing order of the size of the constellations they are coming from. Mixed-Q modulation is a general concept which can be applied to any sequential decoding systems for complexity reduction purposes.

REFERENCES


Fig. 2. complexity exponent vs. SNR, $m = 4$, $n = 3$, $q = 24$, rate = 12 bits per channel use, fixed-Q, and mixed-Q

Fig. 3. Number of nodes vs. $k$ (level), $m = 4$, $n = 3$, $q = 24$, rate = 12 bits per channel use at SNR=20dB, fixed-Q, and mixed-Q of $(8\{8\}, 8\{4\}, 8\{2\})$

Fig. 4. BER vs. SNR, $m = 4$, $n = 3$, $q = 24$, rate = 12 bits per channel use, $(8\{8\}, 8\{4\}, 8\{2\})$ mixed-Q code with and without scaling factors.

Fig. 5. complexity exponent vs. SNR, $m = 4$, $n = 4$, $q = 32$, rate = 16 bits per channel use, $(10\{8\}, 12\{4\}, 10\{2\})$ mixed-Q code with and without scaling factors.