Unitary Space-Time Modulation via Cayley Transform

YINDI JING AND BABAK HASSIBI
Department of Electrical Engineering
California Institute of Technology, Pasadena, CA 91125
e-mail: yindi,hassibi@systems.caltech.edu

Abstract

A recently proposed method for communicating with multiple antennas over block fading channels is *unitary space-time modulation* (USTM). In this method, the signals transmitted from the antennas, viewed as a matrix with spatial and temporal dimensions, form a unitary matrix, i.e., one with orthonormal columns. Since channel knowledge is not required at the receiver, USTM schemes are suitable for use on wireless links where channel tracking is undesirable or infeasible, either because of rapid changes in the channel characteristics or because of limited system resources. Recent results have shown that, if suitably designed, USTM schemes can achieve full channel capacity at high SNR, and, moreover, that all this can be done over a single coherence interval, provided the coherence interval and number of transmit antennas are sufficiently large—a phenomenon referred to as *autocoding*.

While all this is well recognized, what is not clear is how to generate good performing constellations of (non-square) unitary matrices, that lend themselves to efficient encoding/decoding. The schemes proposed so far either exhibit poor performance, especially at high rates, or have no efficient decoding algorithms. In this paper, we propose to use the *Cayley transform* to design USTM constellations. This work can be viewed as a generalization, to the non-square case, of the Cayley codes that have been proposed for *differential* USTM. The codes are designed based on an information-theoretic criterion, and lend themselves to polynomial-time (often cubic) near-maximum-likelihood decoding using a sphere decoding algorithm. Simulations suggest that the resulting codes allow for effective high-rate data transmission in multi-antenna communication systems without knowing the channel. However, our preliminary results do not show a substantial advantage over training-based schemes.

Index Terms – Wireless communications, Unitary space-time modulation, unitary space-time codes, diversity product, fading channels, Cayley transform, isotropic distribution, Cauchy random matrices.

1 Introduction and Model

It is well known that multiple transmit and/or receive antennas promise high data rates on scattering-rich wireless channels [1, 2]. Most of the proposed schemes that achieve these high rates require the propagation environment or channel to be known to the receiver (see, e.g., [1, 3, 4, 5] and the references therein). In practice, knowledge of the channel is often obtained via training: known signals are periodically transmitted for the receiver to learn the channel, and the channel parameters are tracked in between the transmission of the training signals. However, it is not always feasible or advantageous to use training-based schemes, especially when many antennas are used or either end of the link is moving so fast that the channel is changing very rapidly [6, 7].

Hence, there is much interest in space-time transmission schemes that do not require either the transmitter or receiver to know the channel. Information-theoretic calculations with a multi-antenna channel that changes in a block-fading manner first appeared in [8]. Based on these calculations, a new transmission scheme, referred to as *unitary space-time modulation* (USTM), in which the transmitted signals, viewed as matrices with spatial and temporal dimensions, form a unitary matrix, was proposed in [9]. Further information-theoretic calculations in [10] and [11] show that, at high SNR, USTM schemes are capable of achieving full channel capacity. Furthermore, in [12], it is shown that all this can be done over a single coherence interval, provided the coherence interval and number of transmit antennas are sufficiently large—a phenomenon referred to as *autocoding*.

While all this is well recognized, it is not clear how to design a constellation of non-square USTM matrices, that deliver on the above information-theoretic results and lend themselves to efficient encoding/decoding. The first technique to design USTM constellations was proposed in [13], which while it allows for efficient decoding, was later shown in [14] to have poor performance, especially at high rates. The constellation proposed in [14], on the other hand, while it theoretically has good performance, has to date no tractable decoding algorithm. Recently, a USTM design method based on the exponential map has been proposed in [15].

In this paper we propose to use the Cayley transform to design USTM constellations. This can be regarded as an extension, to the non-square case, of earlier work on Cayley codes for *differential* USTM [16]. As will be shown in this paper, this extension is non-trivial. Nonetheless, the codes designed here inherit many of the properties of Cayley differential codes. In particular, they:

1. Are very simple to encode: the data is broken into substreams used to parameterize the unitary matrix

2. Can be used for any number of transmit and receive antennas

3. Can be decoded in a variety of ways including simple polynomial-time linear-algebraic techniques such as successive nulling and cancelling (V-BLAST [17, 18]) or sphere decoding [19, 20]

4. Satisfy a probabilistic criterion: they maximize an expected distance between matrix pairs

The paper is organized as follows. Unitary space-time modulation and training-based schemes are introduced briefly in the following two subsections. In section 2, we first tersely present the Cayley transform and its advantages in parameterizing the space of unitary matrices and then illuminate in detail the encoding, decoding and design of our Cayley space-time codes. Simulation results, including the comparison of our Cayley codes with training-based schemes, are shown in section 3. The main result of our investigation is that the Cayley codes do not offer a substantial advantage over training-based schemes. Section 4 provides the conclusion and appendices A and B gives the mathematical calculations for optimizing our Cayley codes basis set.

1.1 Unitary space-time modulation

Consider a wireless communication system with M transmit antennas and N receive antennas. We use a block-fading channel with coherence interval T (for more on this model, see [8, 9]):

$$X = \sqrt{\frac{\rho T}{M}} SH + V \tag{1}$$

Here, $S : T \times M$ denotes the transmitted signal with s_{tm} the signal sent by the mth transmit antenna at time t. The tth row of S indicates the row vector of the transmitted values from all the transmitters at time t and the mth column indicates the transmitted values of the mth transmit antenna across the coherence interval. $H : M \times N$ is the complex-valued propagation matrix which remains constant during the coherent period T and h_{mn} is the propagation coefficient between the mth transmit antenna and the nth receive antenna. The h_{mn}s have a zero-mean unit-variance circularly-symmetric complex Gaussian distribution $\mathcal{CN}(0, 1)$ and are independent of each other. We assume that the channel information is unknown to both the transmitter and the receiver. $V : T \times N$ is the noise with v_{tn} the noise at the nth receive antenna at time t. The v_{tn}s are iid with $\mathcal{CN}(0, 1)$ distribution. $X : T \times N$ is the received signal matrix with x_{tn} the received value by the nth receive antenna at time t. The tth row of X indicates the row vector of the received values at all the receivers at time t and the nth column indicates the received values of the nth transmit antenna across the coherence interval. We impose an extra power constraint on the transmitted signal

$$\frac{1}{M} \sum_{m=1}^{M} \mathrm{E}\, |s_{tm}|^2 = \frac{1}{T}, \quad t = 1, 2, ..., T \tag{2}$$

which means that the average expected power over the M transmitted antennas is kept constant for each channel use. Therefore ρ represents the expected SNR at each receive antenna.

Conditioned on S, from (1), we can see that the received signal X has independent and identically distributed columns (across the N antennas). At a particular antenna, the T received symbols are zero-mean complex Gaussian, with the following $T \times T$ covariance matrix

$$\Lambda = I_T + (\rho T/M) SS^*,$$

where S^* means the conjugate transpose of matrix S and I_T is the $T \times T$ identity matrix. (Without causing confusion we omit the subscript sometime later.) The received signal thus has the following conditional probability density

$$p(X|S) = \frac{\exp(-tr\{\Lambda^{-1} X X^*\})}{\pi^{TN} \det^N \Lambda}, \tag{3}$$

where ”tr” denotes the trace function.

The conditional density (3) has considerable symmetry arising from the statistical equivalence of each time-sample and of each transmit antenna. Its special properties, combined with the concavity of the mutual information function, lead to the following theorem summarized in [8] [9] and [10].

Theorem 1 (Structure of Capacity-achieving Signal). *[8] A capacity-achieving random signal matrix for (1) may be constructed as a product $S = VD$, where V is a $T \times T$ isotropically distributed unitary matrix, and D is an independent $T \times M$ real, nonnegative, diagonal matrix. Furthermore, for either $T \gg M$, or high SNR with $T > M$, $d_{11} = d_{22} = \cdots = d_{MM} = 1$ achieves capacity where D_{ii} is the ith diagonal entry of D.*

An isotropically distributed $T \times T$ unitary matrix has a probability density that is unchanged when the matrix is multiplied by any deterministic unitary matrix. In a natural way, an isotropically distributed unitary matrix is the $T \times T$ counterpart of a complex scalar having unit magnitude and uniformly distributed phase. For more on the isotropic distribution, refer to [8].

Motivated by this theorem, [9] proposed to use the transmitted signal matrix S as $S = \Phi \left[I_M \quad 0_{T-M,M} \right]^t$ with Φ a $T \times T$ unitary matrix. The superscript 't' indicates the transpose and $0_{T-M,M}$ is the $(T - M) \times M$ matrix of all zeros. (Without causing of confusion we omit the subscript later.) This is called unitary space-time modulation (USTM), and such an S is called a $T \times M$ unitary matrix since its M columns are orthonormal. In the USTM scheme, the transmitted signals are chosen from a constellation $\mathcal{V} = \{S_1, \ldots, S_L\}$ of $L = 2^{RT}$ (where R is the transmission rate) $T \times M$ unitary matrices. The ML decoder is given by

$$\hat{\ell} = \arg \max_{\ell=1,\ldots,L} \|X^* S_\ell\|_F^2 = \arg \min_{\ell=1,\ldots,L} \|X^* S_\ell^\perp\|_F^2, \tag{4}$$

where $S^\perp$ is the $T \times (T - M)$ unitary complement matrix of the $T \times M$ unitary matrix S, that is, $[S, S^\perp]$ is a $T \times T$ unitary matrix. $\|\cdot\|_F$ indicates the Frobenius norm.

In [9] it is also shown that the pairwise block probability of error (of transmitting S_ℓ and erroneously decoding $S_{\ell'}$) has the Chernoff upper bound

$$Pe \leq \frac{1}{2} \prod_{m=1}^{M} \left[\frac{1}{1 + \frac{(\rho T/M)^2 (1 - d_m^2)}{4(1 + \rho T/M)}} \right]^N,$$

where $1 \geq d_1 \geq ... \geq d_M \geq 0$ are the singular values of the $M \times M$ matrix $S_{\ell'}^* S_\ell$. The formula shows that the pairwise probability of error behaves as $|\det(S_{\ell'}^* S_\ell)|^{-2N}$. Therefore, most design schemes have focused on finding a constellation that maximizes $\min_{\ell \neq \ell'} |\det(S_{\ell'}^* S_\ell)|$. Since L can be quite large, this calls into question the feasibility of computing and using this performance criterion. The large number of signals also rules out the possibility of decoding via an exhaustive search. To design constellations that are huge, effective, and yet still simple, so that they can be decoded in real-time, we need to introduce some structure. We will show how the Cayley transform can be used later.

1.2 Training-based schemes

When the channel information of a multiple-antenna communication system is unknown, training-based schemes are generally used, by which known signals are periodically transmitted for the receiver to learn the channel. It is meaningful to compare the performance of our Cayley unitary space-time codes with that of the training-based schemes. We first introduce the training-based schemes here.

Training-based schemes dedicate part of the transmitted matrix S to be a known training signal from which H can be learned. In particular, training-based schemes are composed of two phases: The training phase and the data transmission phase.

The system equations for the training phase can be written as

$$X_\tau = \sqrt{\frac{\rho_\tau}{M}} S_\tau H + V_\tau, \ \ \mathrm{tr}\, S_\tau S_\tau^* = MT_\tau$$

where S_τ is the $T_\tau \times M$ complex matrix of training symbols sent over T_τ time samples and known to the receiver, ρ_τ is the SNR during the training phase, X_τ is the $T_\tau \times N$ complex received matrix, and V_τ is the noise matrix.

Similarly, the system equations for the data transmission phase can be written as

$$X_d = \sqrt{\frac{\rho_\tau}{M}} S_d H + V_d, \ \ \mathrm{E}\, \mathrm{tr}\, S_d S_d^* = MT_d$$

where S_d is the $T_d \times M$ complex matrix of data symbols sent over $T_d = T - T_\tau$ time samples, ρ_d is the SNR during the data transmission phase, X_d is the $T_d \times N$ complex received matrix, and V_d is the noise matrix. The normalization formula above has an expectation because S_d is random and unknown. Note that $\rho T = \rho_d T_d + \rho_\tau T_\tau$.

There are two general methods to estimate the channel information: the ML (maximum-likelihood) and the LMMSE (linear minimum-mean-square-error) estimation whose channel estimations are given as,

$$\hat{H} = \sqrt{\frac{M}{\rho_\tau}} (S_\tau^* S_\tau)^{-1} S_\tau^* X_\tau, \quad \hat{H} = \sqrt{\frac{M}{\rho_\tau}} (\frac{M}{\rho_\tau} T_M + S_\tau^* S_\tau)^{-1} S_\tau^* X_\tau,$$

respectively. In our simulations, the LMMSE estimation is used.

In [7], the optimal training to maximize the lower bound of the capacity for MMSE estimation is given. There are three parameters that are to be optimized. The first one is the training data S_τ. It is proved that the optimal solution is to choose the training signal as a multiple of a matrix with orthonormal columns. The second one is the length of the training interval. Setting $T_\tau = M$ is optimal for any ρ and T. And third, the optimal power distribution satisfies the following,

$$\begin{cases} \rho_d < \rho < \rho_\tau & \text{if } T > 2M \\ \rho_d = \rho = \rho_\tau & \text{if } T = 2M \\ \rho_d > \rho > \rho_\tau & \text{if } T < 2M \end{cases}$$

In simulations, we do the training in this optimal way by letting $T_\tau = M$ and $S_\tau = \sqrt{M} I_M$. For simplicity, equal-training-and-data-power $\rho_d = \rho = \rho_\tau$ is used, which is optimal when $T = 2M$. By combining the training phase equations and the data transmission phase equations, the system equations can be written as

$$\begin{bmatrix} X_\tau \\ X_d \end{bmatrix} = \sqrt{\rho} \begin{bmatrix} I_M \\ S_d \end{bmatrix} H + \begin{bmatrix} V_\tau \\ V_d \end{bmatrix} \tag{5}$$

Further assume that the $(T-M) \times M$ information matrix S_d is unitary. Then we have

$$S = \frac{1}{\sqrt{2}} \begin{bmatrix} I_M \\ S_d \end{bmatrix} \text{ and } S^\perp = \frac{1}{\sqrt{2}} \begin{bmatrix} -S_d^* \\ I_{T-M} \end{bmatrix} \tag{6}$$

where, $S^\perp$ is the $T \times (T-M)$ unitary complement matrix of the $T \times M$ unitary matrix S. If S_d is not unitary, then $S^\perp$ is only the orthogonal complement, $S^{\perp *} S = 0$, since the unitary complement may not exist.

2 Cayley unitary space-time codes

2.1 Parameterization of the unitary matrix space by the Cayley transform

In USTM, the first M columns of the $T \times T$ unitary matrices are chosen to be the transmitted signal. Therefore, let us first look at the space of the $T \times T$ unitary matrices which is referred as the *Stiefel* manifold. It is well-known that this manifold is highly non-linear and non-convex. Note that an arbitrary complex $T \times T$ matrix has $2T^2$ real parameters, but for a unitary one, there are T constraints to force each column to have unit norm and another $2 \times \frac{T(T-1)}{2}$ constraints to make the T columns pairwise orthogonal. Therefore, the Stiefel manifold has dimension $2T^2 - T - 2 \times \frac{T(T-1)}{2} = T^2$. Similarly, the space of $T \times M$ unitary matrices has dimension $2TM - M - 2 \times \frac{M(M-1)}{2} = 2TM - M^2$.

To design codes of unitary matrices, we need first a parameterization of the space. There are some parameterization methods in existence but all of them suffer from disadvantages for use in unitary space-time code design. We now briefly discuss these.

The first parameterization method is with Givens rotations. A unitary matrix Φ can be written as the product

$$\Phi = G_1 G_2 \cdots G_{T(T-1)/2} D G_{T(T+1)/2} \cdots G_{T(T-1)}$$

where D is a diagonal unitary matrix and the G_is are the Givens (or planar) rotations, one for each of the $\frac{T(T-1)}{2}$ two-dimensional hyper-planes [21]. It is conceivable that one can encode the data onto the angles of rotations and also the diagonal phases of D. But it is not a practical method since neither is the parameterization one-to-one (for example, one can reorder the Givens rotations) nor does systematic decoding appears to be possible.

Another method is to parameterize with Householder reflections. A unitary matrix Φ can be written as the product $\Phi = DH_1H_2 \cdots H_T$, where D is a diagonal matrix and the H_is are Householder matrices. This method is also not encouraging to us because we do not know how to encode and decode the data onto the Householder matrices in any efficient manner.

And also, unitary matrices can be parameterized with the matrix exponential $\Phi = e^{iA}$. When A is $T \times T$ Hermitian, Φ is unitary. The exponential map also has the difficulty of not being one-to-one. This can be overcome by imposing the constraints $0 \leqslant A < 2\pi I$, but the constraints are not linear although convex. We do not know how to sample the space of A to obtain a constellation of Φ. Moreover, the map is not easy to be converted at the receiver for $T > 1$. Nonetheless, a method based on the exponential map has been proposed in [15].

2.1.1 Cayley transform and its properties

The Cayley transform was proposed in [16] and used to design codes for differential unitary space-time modulation whereby both good performance and simple encoding and decoding are obtained.

The Cayley transform of a complex $T \times T$ matrix Y is defined to be

$$\Phi = (I + Y)^{-1}(I - Y)$$

where Y is assumed to have no eigenvalue at -1 so that the inverse exists. Let A be a $T \times T$ Hermitian matrix and consider the Cayley transform of the skew-Hermitian matrix $Y = iA$:

$$\Phi = (I + iA)^{-1}(I - iA). \tag{7}$$

First note that since iA is skew-Hermitian, it has no eigenvalue at -1 because all its eigenvalues are strictly imaginary. That means that $(I + iA)^{-1}$ always exists. The Cayley transform is the generalization of the scalar transform

$$v = \frac{1 - ia}{1 + ia}$$

that maps the real line to the unit circle. Notice that no finite point on the real line can be mapped to the -1 point on the unit circle.

Also,

$$\begin{aligned} \Phi\Phi^* &= (I + iA)^{-1}(I - iA)[(I + iA)^{-1}(I - iA)]^* \\ &= (I + iA)^{-1}(I - iA)(I + iA)(I - iA)^{-1} \\ &= I \end{aligned}$$

The second equation is true because $I - iA$, $I + iA$, $(I - iA)^{-1}$ and $(I + iA)^{-1}$ all commute. And similarly, $\Phi^*\Phi = I$ can also be proved. Therefore, similar to the matrix exponential, the Cayley transform maps Hermitian matrices to unitary matrices. Also, from (7) it can be proven easily that

$$iA = (I + \Phi)^{-1}(I - \Phi)$$

provided that $(I+\Phi)^{-1}$ exists. This shows that the Cayley transform and its inverse transform coincide. Thus, the Cayley transform is one-to-one. It is not an onto map because those unitary matrices with eigenvalues at -1 have no inverse images. Recall that the space of Hermitian or skew-Hermitian matrices has dimension T^2 which matches that of the Stiefel manifold.

We have shown that a matrix with no eigenvalues at -1 is unitary if and only if its Cayley transform is skew-Hermitian. Compared with other parameterizations of unitary matrices, the parameterization with Cayley transform is one-to-one and easily invertible. The Cayley transform maps the complicated Stiefel manifold of unitary matrices to the space of skew-Hermitian (Hermitian) matrices. And skew-Hermitian (Hermitian) matrices are easy to characterize since they form a linear vector space over the reals. Therefore, easy encoding and decoding can be obtained by this handy feature.

And also, it is proved in [16] that a set of unitary matrices is fully-diverse if and only if the set of their skew-Hermitian Cayley transforms is fully-diverse. This suggests that a promising performance set of unitary matrices can be obtained from a well-designed set of Hermitian matrices by Cayley transform.

2.2 Cayley unitary space-time codes

Because the Cayley transform maps the nonlinear Stiefel manifold to the linear space (over the reals) of Hermitian (or skew-Hermitian) matrices (and vice-versa), it is convenient and most straightforward to encode data linearly onto a skew-Hermitian matrix and then apply the Cayley transform to get a unitary matrix.

We call a Cayley unitary space-time code one for which each $T \times M$ unitary matrix is

$$S=(I_T+iA)^{-1}(I_T-iA)\begin{bmatrix} I_M \\ 0 \end{bmatrix} \tag{8}$$

with the Hermitian matrix A given by

$$A=\sum_{q=1}^{Q}\alpha_q A_q, \tag{9}$$

where $\alpha_1, \alpha_2, ..., \alpha_Q$ are real scalars (chosen from a set $\mathcal{A}$ with r possible values) and $A_1, A_2, ..., A_Q$ are fixed $T \times T$ complex Hermitian matrices.

The code is completely determined by the set of matrices $\{A_1, A_2, ..., A_Q\}$, which can be thought of as Hermitian basis matrices. Each individual codeword, on the other hand, is determined by our choice of the scalars $\alpha_1, \alpha_2, ..., \alpha_Q$ whose values are in the set $\mathcal{A}_r$ (The subscript 'r' represents the cardinality of the set). Since each of the Q real coefficients may take on r possible values and the code occupies T channel uses, the transmission rate is $R=(Q/T)\log_2 r$. We defer the discussion of how to design the A_q's and how to choose Q and the set $\mathcal{A}_r$ later in this section and concentrate on how to decode $\alpha_1, \alpha_2, ..., \alpha_Q$ at the receiver first.

2.3 Decoding of Cayley codes

Similar to the differential Cayley codes, our Cayley unitary space-time codes also have the good property of linear decoding, which means that the receiver can be made to form a system of linear equations in the real scalars $\alpha_1, \alpha_2, ..., \alpha_Q$. First, it is useful to see what our codes and their ML decoding look like.

We partition the $T \times T$ matrix A as $\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$, where A_{11} is an $M \times M$ matrix and A_{22} is a $(T-M) \times (T-M)$ matrix. For A being Hermitian, A_{11} and A_{22} must both be Hermitian and also $A_{21} = A_{12}^*$.

Observe that

$$\begin{aligned} \Phi &= (I+iA)^{-1}(I-iA) \\ &= (I+iA)^{-1}(2I-(I+iA)) \\ &= 2(I+iA)^{-1} - I \end{aligned}$$

Using the above, some algebra shows

$$\Phi = \begin{bmatrix} 2[I-(I+iA_{11})^{-1}A_{12}\Delta_2^{-1}A_{12}^*](I+iA_{11})^{-1} - I & -2i(I+iA_{11})^{-1}A_{12}\Delta_2^{-1} \\ -2i\Delta_2^{-1}A_{12}^*(I+iA_{11})^{-1} & 2\Delta_2^{-1} - I \end{bmatrix}$$

where $\Delta_2 = I + iA_{22} + A_{12}^*(I+iA_{11})^{-1}A_{12}$ is the Schur complement of $I+iA_{11}$ in $I+A$.

Therefore, our transmitted signal has the following structure,

$$S = \begin{bmatrix} 2[I-(I+iA_{11})^{-1}A_{12}\Delta_2^{-1}A_{12}^*](I+iA_{11})^{-1} - I \\ -2i\Delta_2^{-1}A_{12}^*(I+iA_{11})^{-1} \end{bmatrix} \quad \text{and} \quad S^{\perp} = \begin{bmatrix} -2i(I+iA_{11})^{-1}A_{12}\Delta_2^{-1} \\ 2\Delta_2^{-1} - I \end{bmatrix} \tag{10}$$

In fact, it can be algebraically verified that both S and $S^{\perp}$ are unitary.

By partitioning the received signal matrix X into an $M \times N$ block X_1 and a $(T-M) \times N$ block X_2 as $X = [X_1^t, X_2^t]^t$, the second form of the ML decoder in (4) reduces to

$$\arg\min_{\{\alpha_q\}} \left\| [-2iX_1^*(I+iA_{11})^{-1}A_{12} + X_2^*(2-\Delta_2)]\Delta_2^{-1} \right\|_F^2.$$

The reason for choosing the second form of the ML, as opposed to the first one, is that we prefer to minimize, rather than maximize the Frobenius norm. In fact, we shall presently see that a simple approximation leads us to a quadratic minimization problem, which can be solved conveniently via sphere decoding.

As mentioned, the decoder is not quadratic in the entries of A, which indicates that it is not quadratic in the α_qs since the matrix A is linear in the α_qs. Therefore, the system equation at the receiver is not linear. The formula looks intractable because there are matrix inverses as well as the Schur complement Δ_2. If we adopt the approach of [16] by ignoring the covariance of the additive noise term Δ_2^{-1}, we obtain

$$\arg\min_{\{\alpha_q\}} \left\| 2X_2^* - X_2^*\Delta_2 - 2iX_1^*(I+iA_{11})^{-1}A_{12} \right\|_F^2 \tag{11}$$

which, however, is still not quadratic in the entries of A. Therefore, to simplify the formula, more constraints should be imposed on the Hermitian matrix A. That means our A matrix should have a more handy structure. Fortunately, observe that the degrees of freedom in a $T \times T$ Hermitian matrix is T^2, but the degrees of freedom in a $T \times M$ unitary matrix S is only $2TM - M^2 = T^2 - (T-M)^2$. There are $(T-M)^2$ more degrees of freedom in A than we need. So let us exploit this. Indeed, if we let

$$(I + iA_{11})^{-1} A_{12} = B \tag{12}$$

for some *fixed* $M \times (T-M)$ matrix B by which $2M(T-M)$ degrees of freedom are lost [1]. Therefore,

$$A_{12} = (I + iA_{11})B \tag{13}$$

and

$$\Delta_2 = I + B^* B - iB^* A_{11} B + iA_{22}. \tag{14}$$

Some algebra shows that the above decoding formula (11) reduces to

$$\hat{\alpha}_{\text{lin}} = \arg\min_{\{\alpha_q\}} \| X_2^* - X_2^* B^* B - 2iX_1^* B + iX_2^* B^* A_{11} B - iX_2^* A_{22} \|_F^2 \tag{15}$$

which is now quadratic in the entries of A. Fast decoding methods such as sphere decoding and nulling and cancelling can be used in polynomial time as in BLAST [1].

We call (15) the "linearized" decoder because the system of equations obtained in solving for the unconstrained α_qs is linear. For a wide range of rates and SNR, (15) can be solved exactly in roughly $O(Q^3)$ computations using sphere decoding [19, 20]. Furthermore, simulation results show that the penalty for using (15) is small, especially when weighed against the complexity of exact ML. To facilitate the presentation of these decoding algorithms, we write down the equivalent channel model in matrices in the following section.

2.3.1 Equivalent model

From (12), $A_{12} = A_{21}^*$ is fully determined by A_{11}. Therefore, the degrees of freedoms in A are all from matrices A_{11} and A_{22}. The encoding formula (9) of A can thus be modified to the following encoding formulas of A_{11} and A_{22}.

$$A_{11} = \sum_{q=1}^{Q} \alpha_q A_{11,q} \quad \text{and} \quad A_{22} = \sum_{q=1}^{Q} \alpha_q A_{22,q} \tag{16}$$

where, Q is the number of possible $A_{11,q}$s and $A_{22,q}$s, $\alpha_1, \alpha_2, ...\alpha_Q$ are real scalars chosen from the set $\mathcal{A}_r$, $A_{11,1}, A_{11,2}, ..., A_{11,Q}$ and $A_{22,1}, A_{22,2}, ..., A_{22,Q}$ are fixed $M \times M$ and $(T-M) \times (T-M)$ complex

[1] With these conditions, the number of degrees of freedom in A is $T^2 - 2TM + 2M^2$, which is greater than $2TM - M^2$, the number of degrees of freedom in an arbitrary $T \times M$ unitary matrix, when $T \geq 3M$.

Hermitian matrices. The matrix A is constructed as the following,

$$\begin{aligned}
A &= \begin{bmatrix} A_{11} & (I+iA_{11})B \\ B^*(I-iA_{11}) & A_{22} \end{bmatrix} \\
&= \begin{bmatrix} \sum_{q=1}^{Q} \alpha_q A_{11,q} & (I+i\sum_{q=1}^{Q} \alpha_q A_{11,q})B \\ B^*(I-i\sum_{q=1}^{Q} \alpha_q A_{11,q}) & \sum_{q=1}^{Q} \alpha_q A_{22,q} \end{bmatrix} \\
&= \sum_{q=1}^{Q} \alpha_q \begin{bmatrix} A_{11,q} & iA_{11,q}B \\ -iB^*A_{11,q} & A_{22,q} \end{bmatrix} + \begin{bmatrix} 0 & B \\ B^* & 0 \end{bmatrix}
\end{aligned}$$

Therefore, the linearized ML decoder (15) can be written as,

$$\arg\min_{\{\alpha_q\}} \left\| X_2^* - X_2^* B^* B - 2iX_1^* B + i\sum_{q=1}^{Q} \alpha_q X_2^* B^* A_{11,q} B - i\sum_{q=1}^{Q} \alpha_q X_2^* A_{22,q} \right\|_F^2 \tag{17}$$

By defining

$$C = X_2^* - X_2^* B^* B - 2iX_1^* B, \;\; J_q = -iX_2^* B^* A_{11,q} B + iX_2^* A_{22,q} \tag{18}$$

for $q = 1, 2, ..., Q$, and decomposing the complex matrices C and J_q into their real and imaginary parts, the decoding formula (17) can be further rewritten as

$$\arg\min_{\{\alpha_q\}} \left\| \begin{bmatrix} C_R \\ C_I \end{bmatrix} - \begin{bmatrix} J_{1,R} & \cdots & J_{Q,R} \\ J_{1,I} & \cdots & J_{Q,I} \end{bmatrix} \begin{bmatrix} \alpha_1 I_{T-M} \\ \vdots \\ \alpha_Q I_{T-M} \end{bmatrix} \right\|_F^2$$

where C_R, C_I are the real and imaginary parts of the matrix C and $J_{i,R}, J_{i,I}$ are the real and imaginary parts of the matrices J_i. Denoting by $C_{R,j}, C_{I,j}, J_{i,R,j}, J_{i,I,j}$ the jth columns of $C_R, C_I, J_{i,R}, J_{i,I}$ for $j = 1, 2, ..., (T-M)$, and writing the matrices in the above formula column by column, the formula can be further simplified to

$$\arg\min_{\{\alpha_q\}} \|\mathcal{R} - \mathcal{H}\underline{\alpha}\|_F^2 \tag{19}$$

where $\mathcal{R}$ is the $2N(T-M)$-dimensional column vector $\begin{bmatrix} C_{R,1}^t & C_{I,1}^t & \cdots & C_{R,T-M}^t & C_{I,T-M}^t \end{bmatrix}^t$ and $\mathcal{H}$ is the $2N(T-M) \times Q)$ matrix

$$\begin{bmatrix} J_{1,R,1} & J_{2,R,1} & \cdots & J_{Q,R,1} \\ J_{1,I,1} & J_{2,I,1} & \cdots & J_{Q,I,1} \\ \vdots & \vdots & \ddots & \vdots \\ J_{1,R,T-M} & J_{2,R,T-M} & \cdots & J_{Q,R,T-M} \\ J_{1,I,T-M} & J_{2,I,T-M} & \cdots & J_{Q,I,T-M} \end{bmatrix} \tag{20}$$

$\underline{\alpha} = [\alpha_1, \cdots, \alpha_Q]^t$ is the vector of unknowns. We can get the equivalent channel model

$$\mathcal{R} = \mathcal{H}\underline{\alpha} + \mathcal{W} \tag{21}$$

where $\mathcal{W}$ is the noise matrix. $\underline{\alpha}$ appears to pass through an equivalent channel $\mathcal{H}$ that is known to the receiver because it is a function of $A_{11,1}, A_{11,2}, ..., A_{11,Q}, A_{22,1}, A_{22,2}, ..., A_{22,Q}$ and X_1, X_2 and is corrupted by additive noise [2]. The receiver can simply get the equivalent channel from (20).

Therefore, we have a simple linear system of equations that may be decoded using known techniques such as successive nulling and cancelling, its efficient square-root implementation, or sphere decoding. Efficient implementations of nulling and cancelling generally require $O(Q^3)$ computations. Sphere decoding can be regarded as a generalization of nulling and cancelling where at each step, rather than making a hard decision on the corresponding α_qs, one considers all the α_qs that lie within a sphere of certain radius. Sphere decoding has the important advantage over nulling and cancelling that it computes the *exact* solution. Its worst case behavior is exponential in Q, but its average behavior is comparable to nulling and cancelling. When the number of transmit antennas and the rate are small, ML decoding is possible. However, exact ML decoding generally requires a search over all possible $\alpha_1, ..., \alpha_Q$, which may be impractical for large T and R. Fortunately, the performance penalty for the linearized maximum likelihood (15) is small, especially weighed against the complexity of exact ML.

2.3.2 Number of independent equations

Nulling and cancelling explicitly requires that the number of equations be at least as large as the number of unknowns. Sphere decoding does not have this hard constraint, but it benefits from more equations because the computational complexity grows exponentially in the difference between the number of unknowns and the number of independent equations. To keep the complexity of the sphere decoding algorithm polynomial, it is important that the number of linear equations resulting from (15) be at least as large as the number of unknowns. (21) suggests that there are $2N(T-M)$ real equations and Q real unknowns. Hence we may impose the constraint

$$Q \leq 2N(T-M).$$

This argument assumes that the matrix H has full column rank. There is, at first glance, no reason to assume otherwise but it turns out to be false. Due to the Hermitian constraints, not all the $2M(T-M)$ equations are independent. A careful analysis yields the following result.

Theorem 2 (Rank of H). *The matrix given in (20) generally has rank*

$$rank(\mathcal{H}) = \begin{cases} \min\left(2N(T-M) - N^2, Q\right) & \text{if } T-M \geq N \\ \min\left((T-M)^2, Q\right) & \text{if } T-M < N \end{cases} \tag{22}$$

[2] In general, the covariance of the noise is dependent on the transmitted signal. However, in ignoring Δ_2^{-1} in (11), we have ignored this signal dependence.

Proof: First assume that $T - M \geq N$. The rank of $\mathcal{H}$ is the dimension of the range space of $\underline{c}$ in the equation $\underline{c} = \mathcal{H}\underline{a}$ as $\underline{a}$ varies. Equivalently, the rank of $\mathcal{H}$ is the dimension of the range space of the $N \times (T - M)$ complex matrix C in the equation $C = iX_2^*(A_{22} - B^*A_{11}B)$ when A_{11} and A_{22} vary. Because A_{11} and A_{22} are not arbitrary matrices, the range space of C cannot have all the $2(T - M)N$ dimensions as it appears. Now let's study the number of constraints added on the range space of C as A_{11} and A_{22} can only be Hermitian matrices. Since,

$$\begin{aligned} [C(iX_2)]^* &= -iX_2^*(A_{22} - B^*A_{11}B)(-i)X_2 \\ &= iX_2^*(A_{22} - B^*A_{11}B)(iX_2) \\ &= C(iX_2) \end{aligned}$$

which shows that the $N \times N$ matrix $C(iX_2)$ is Hermitian. This enforces N^2 linear constraints on the entries of C. Therefore, only at most $2(T - M)N - N^2$ entries of all the $2(T - M)N$ entries are free. Since, $\mathcal{H}$ is $2(T - M)N \times Q$, the rank of $\mathcal{H}$ is at most $\min\left(2(T - M)N - N^2, Q\right)$.

Now assume that $T - M < N$. We know that the $N \times N$ matrix $C(iX_2)$ is Hermitian but it has rank $T - M < N$ now instead of full rank. So, the entries of the lower right $(N - (T - M)) \times (N - (T - M))$ Hermitian sub-matrix of $C(iX_2)$ are uniquely determined by its other entries. Therefore the number of constraints yielded by the equations $C(iX_2) = (C(iX_2))^*$ is $N^2 - (N - (T - M))^2 = 2N(T - M) - (T - M)^2$. Thus, there are at most $2N(T - M) - (2N(T - M) - (T - M)^2) = (T - M)^2$ degrees of freedom in C. The rank of $\mathcal{H}$ is at most $\min((T - M)^2, Q)$.

We have essentially proved an upper bound on the rank. Our argument so far has not relied on any specific sets for A_{11} and A_{22}. When $A_{11} = 0$, we are reduced to studying $iX_2^*A_{22}$, which is the same setting as that of differential USTM [16]. In Theorem 1 of [16], it is argued that for a generic choice of the basis matrices $A_{22,1}, \cdots, A_{22,Q}$, the rank of $\mathcal{H}$ attains the upper bound. Therefore the same holds here and $\mathcal{H}$ attains the upper bound. $\square$

Theorem 2 shows that even though there are $2N(T - M)$ equations in (21), not all of them are independent. To have at least as many equations as unknowns, The following constraint is needed

$$Q \leq \begin{cases} 2N(T - M) - N^2 & \text{if } T - M \geq N \\ (T - M)^2 & \text{if } T - M < N \end{cases}$$

or equivalently,

$$Q \leq \min(T - M, N)\max(2(T - M) - N, T - M) \tag{23}$$

2.4 A geometric property of the Cayley space-time codes

With the choice (12) or equivalently (13), the first block of the transmitted matrix S in (10) can be simplified as the following.

$$\begin{aligned}
& 2[I-(I+iA_{11})^{-1}A_{12}\Delta_2^{-1}A_{12}^*](I+iA_{11})^{-1}-I \\
= \;& [2I-2B\Delta_2^{-1}B^*(I-iA_{11})-(I+iA_{11})](I+iA_{11})^{-1} \\
= \;& [(I-iA_{11})-2B\Delta_2^{-1}B^*(I-iA_{11})](I+iA_{11})^{-1} \\
= \;& [I-2B\Delta_2^{-1}B^*](I-iA_{11})(I+iA_{11})^{-1}
\end{aligned}$$

The second block of S equals $-2i\Delta_2^{-1}B^*(I-iA_{11})(I+iA_{11})^{-1}$. Since $(I-iA_{11})$ and $(I+iA_{11})^{-1}$ are commute,

$$S=\begin{bmatrix} I-2B\Delta_2^{-1}B^* \\ -2i\Delta_2^{-1}B^* \end{bmatrix}(I-iA_{11})(I+iA_{11})^{-1}.$$

Our Cayley unitary space-time code and its unitary complement can be written as

$$S=\begin{bmatrix} I & -iB \\ 0 & I \end{bmatrix}\begin{bmatrix} I_M \\ -2i\Delta_2^{-1}B^* \end{bmatrix}U_1 \quad \text{and} \quad S^\perp=\begin{bmatrix} -2iB\Delta_2^{-1} \\ 2\Delta_2^{-1}-I_{T-M} \end{bmatrix} \tag{24}$$

where

$$\Delta_2=I+B^*B-i\sum_{q=1}^{Q}\alpha_q B^*A_{11,q}B+i\sum_{q=1}^{Q}\alpha_q A_{22,q} \tag{25}$$

and $U_1=(I+iA_{11})^{-1}(I-iA_{11})$ is an $M\times M$ unitary matrix since it is the Cayley transform of the Hermitian matrix A_{11}.

The code is completely determined by the matrices $A_{11,1},A_{11,2},...,A_{11,Q}$ and $A_{22,1},A_{22,2},...,A_{22,Q}$, which can be thought of as Hermitian basis matrices. Each individual codeword, on the other hand, is determined by our choice of the scalars $\alpha_1,\alpha_2,...,\alpha_Q$ chosen from the set $\mathcal{A}_r$. Since there are Q basis matrices for A_{11} and A_{22}, and the code occupies T channel uses, the transmission rate is

$$R=\frac{Q}{T}\log_2 r \tag{26}$$

Since the channel matrix H is unknown and if left multiplied by an $M\times M$ unitary matrix, its distribution remains unchanged, we can combine U_1 with the channel matrix H to get $H'=U_1H$. If we left multiply X,S and V by $\begin{bmatrix} I_M & -iB \\ 0 & I_{T-M} \end{bmatrix}^{-1}=\begin{bmatrix} I_M & iB \\ 0 & I_{T-M} \end{bmatrix}$ to get X',S' and V', the system equation (1) can be rewritten as,

$$X'=\sqrt{\frac{\rho T}{M}}\begin{bmatrix} I_M \\ -2i\Delta_2^{-1}B^* \end{bmatrix}H'+V'$$

We can see that this is very similar to the equations of the training-based schemes (6). The only difference is in the noises. In (6), entries of the noise are independent white Gaussian noise with zero mean and unit variance. Here, the entries of V' are no longer independent with unit variance, although they still have zero mean. The dependence of the noises is beneficial to the performance since more information can be obtained.

The following theorem about the structure of $S^\perp$ is needed later in the optimization of the basis matrices.

Theorem 3 (Difference of unitary complements of the transmitted signal). *The difference of the unitary complements $S^\perp$ and $\hat{S}^\perp$ of the transmitted signals S and $\hat{S}$ can be written as*

$$S^\perp - \hat{S}^\perp = 2 \begin{bmatrix} -iB \\ I \end{bmatrix} \Delta_2^{-1}(\hat{\Delta}_2 - \Delta_2)\hat{\Delta}_2^{-1} \tag{27}$$

where Δ_2 and $\hat{\Delta}_2$ are the corresponding Schur complements.

Proof: See Appendix A.

Another way to look at Theorem 3 is to note that

$$S^\perp = \begin{bmatrix} 0 \\ -I \end{bmatrix} + 2 \begin{bmatrix} -iB \\ I \end{bmatrix} \Delta_2^{-1}. \tag{28}$$

Without the unitary constraint, this is an affine space since all the data is encoded in Δ_2^{-1}. So, in general, the space of $S^\perp$ is the intersection of the linear affine space in (28) and the Stiefel manifold $S^{\perp *}S^\perp = I$. We can see from (27) or (28) that the dimension of the range space of $S^\perp - S'^\perp$ (equivalently the dimension of the affine space) is $T - M$. It is interesting to contrast this with the training case, which from (6), gives

$$S^\perp - \hat{S}^\perp = \frac{1}{\sqrt{2}} \begin{bmatrix} -(S_d^* - \hat{S}_d^*) \\ 0 \end{bmatrix} \tag{29}$$

Note now that the dimension of the affine space is $\min(M, T - M)$ which is no more than $T - M$ when $T > 2M$. So, the affine space of $S^\perp$ for the Cayley codes has a higher dimension than that of the training-based schemes when $T > 2M$.

2.5 Design of unitary space-time codes

Although we have introduced the Cayley unitary space-time code structure in (24), we have not yet specified Q, nor have we explained how to design the Hermitian basis matrix sets $\{A_{11,1}, A_{11,2}, ..., A_{11,Q}\}$ and $\{A_{22,1}, A_{22,2}, ..., A_{22,Q}\}$, or choose the discrete set $\mathcal{A}_r$ from which the α_qs are drawn. We now discuss these issues.

2.5.1 Design of Q

To make the constellation as rich as possible, we should make the number of degrees of freedom Q as large as possible. Therefore, as a general practice, we find it useful to take Q as its upper limit in (23). That is,

$$Q = \min(T - M, N) \max(2(T - M) - N, T - M) \tag{30}$$

We are left with how to design the discrete set $\mathcal{A}_r$ and how to choose $\{A_{11,1}, A_{11,2}, ...A_{11,Q}\}$ and $\{A_{22,1}, A_{22,2}, ...A_{22,Q}\}$.

2.5.2 Design of $\mathcal{A}_r$

As mentioned in the introduction, at high SNR, to achieve capacity in the sense of maximizing mutual information between X and S, $\Phi = (I + iA)^{-1}(I - iA)$ should assemble samples from an isotropic random distribution. Since our data modulate the A matrix (A_{11} and A_{22}), equivalently, we need to find the distribution on A that yields an isotropically distributed Φ.

As proved in [16], the unitary matrix Φ is isotropically distributed if and only if the Hermitian matrix A has the matrix Cauchy distribution

$$p(A) = \frac{2^{T^2-T}(T-1)! \cdots 1!}{\pi^{T(T+1)/2}} \frac{1}{\det(I + A^2)^T},$$

which is the matrix generalization of the familiar scalar Cauchy distribution

$$p(a) = \frac{1}{\pi(1 + a^2)}.$$

For the one-dimensional case, an isotropic-distributed scalar v can be written as $v = e^{i\theta}$, where θ is uniform over $[0, 2\pi)$. So, $a = -i\frac{1-e^{i\theta}}{1+e^{i\theta}} = -\tan(\theta/2)$ is Cauchy. When there is only one transmit antenna ($M = 1$), and the coherence interval is one channel use only ($T = 1$), the transmitted signals are scalars. There is no need to partition the matrix A. Therefore (9) is used instead of (16). We want our code constellation $A = \sum_{q=1}^{Q} \alpha_q A_q$ to resemble samples from a Cauchy random matrix distribution. Since there is only one degree of freedom in a scalar, it is obvious that $Q = 1$. Without loss of generality, setting $A_1 = 1$, we get

$$v = \frac{1 - i\alpha_1}{1 + i\alpha_1}, \text{ and } \alpha_1 = -i\frac{1 - v}{1 + v}.$$

To have a code with rate $R = (Q/T)\log_2 r$ with $T = M = 1$, $\mathcal{A}$ should have $r = 2^R$ points. Standard DPSK puts these points uniformly around the unit circle at angular intervals of $2\pi/r$ with the first point at π/r. For a point of angle θ on the unit circle, the corresponding value for α_1 is

$$\alpha_1 = -i\frac{1 - v}{1 + v} = -\tan(\theta/2). \tag{31}$$

For example, for $r = 2$, we have the set of points on unit circle $\mathcal{V} = \{e^{i\pi/2}, e^{-i\pi/2}\}$. From (31), the set of values for α_1 is $\mathcal{A}_2 = \{-1, 1\}$. For $r = 4$, $\mathcal{A}_4 = \{-2.4142, -0.4142, 0.4142, 2.4142\}$. It can be seen that the points rapidly spread themselves out as r increases, which reflects the heavy tail of the Cauchy distribution.

We denote $\mathcal{A}_r$ to be the image of (31) applied to the set $\{\pi/r, 3\pi/r, 5\pi/r, ..., (2r-1)\pi/r\}$. When $r \to \infty$, the fraction of points in the set less than some value x is given by the cumulative Cauchy distribution. Therefore, the set $\mathcal{A}_r$ can be regarded as an r-point discretization of a scalar Cauchy random variable.

For the systems with multiple transmit antennas and higher coherence intervals, no direct method is shown about how to choose $\mathcal{A}$. In that case, we also choose our set $\mathcal{A}$ to be the set given above. Thus, the α_qs are chosen as discretized scalar Cauchy random variables for any T and M. But to get rate R, from (26), we need to have

$$r^Q = 2^{RT}. \tag{32}$$

To complete the code construction, it is crucial that $\{A_{11,1}, A_{11,2}, ...A_{11,Q}\}$ and $\{A_{22,1}, A_{22,2}, ...A_{22,Q}\}$ be chosen appropriately, and we present a criterion in the next section.

2.5.3 Design of $A_{11,1}, A_{11,2}, ...A_{11,Q}, A_{22,1}, A_{22,2}, ...A_{22,Q}$

If the rates being considered are reasonably small, the diversity product criterion $\max_{l \neq l'} |\det(\Phi_l - \Phi_{l'})|$ is tractable. At high rates, however, it is not practical to pursue the full diversity criterion. There are two reasons for this: first, the criterion becomes intractable because of the number of matrices involved and second, the performance of the constellation may not be governed so much by its worst-case pairwise $|\det(\Phi_l - \Phi_{l'})|$, but rather by how well the matrices are distributed throughout the space of unitary matrices.

Similar to the differential Cayley code design in [16], for given $\mathcal{A}_r$ and the sets of basis matrices $\{A_{11,1}, A_{11,2}, ...A_{11,Q}\}$ and $\{A_{22,1}, A_{22,2}, ...A_{22,Q}\}$, we define a distance criterion for the resulting constellation of matrices $\mathcal{V}$ to be

$$\xi(\mathcal{V}) = \frac{1}{T-M} E \log\det(S^\perp - S'^\perp)^*(S^\perp - S'^\perp) \tag{33}$$

where S is given by (24) and (25) and S' is given by the same formulas except that the α_qs in (25) are replace by α'_qs. The expectation is over all possible $\alpha_q s$ and α'_qs chosen uniformly from $\mathcal{A}_r$ such that $(\alpha_1, ..., \alpha_Q) \neq (\alpha'_1, ..., \alpha'_Q)$. Remember that $S^\perp$ denotes the $T \times (T-M)$ unitary complement matrix of the $T \times M$ matrix S.

Let us first look at the difference between this criterion with that in [16]. Here, we use $S^\perp$ and $S'^\perp$ instead of S and S' themselves because the unitary complement instead of the transmitted signal itself is used in the linearized ML decoding. This criterion cannot be directly related to the diversity product as in the case of [16], but still, from the structure, it is a measure of the expected "distance" between the matrices $S^\perp$ and $S'^\perp$. Thus, maximizing $\xi(\mathcal{V})$ should be connected with lowering average pairwise error probability. Hopefully, optimizing the expected "distance" between the unitary complements $S^\perp$ and $S'^\perp$ instead of that between the unitary signals S and S' themselves will obtain a better performance. And also, since our constraints (12) is imposed to simplify Δ_2, which turns out to simplify $S^\perp$ as well, the calculation of our criterion is much easier than the calculation of the one used in [16], which maximize the expected "distance" between the unitary matrices Φ and Φ'. We therefore propose the optimization problem to be

$$\arg \max_{\{A_{11,q}, A_{22,q}\}, B} \xi(\mathcal{V}) \tag{34}$$

By (27), we can rewrite the optimization as a function of A_{11}, A_{22} and get the simplified formula,

$$\max_{\{A_{11,q_1}\},\{A_{22,q_2}\},B} \mathrm{E}\,\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2 - \mathrm{E}\,\log\det\Delta_2^2 - \mathrm{E}\,\log\det\Delta_2'^2 \tag{35}$$

where

$$\begin{cases} \Delta_2 = I + B^*B - iB^*A_{11}B + iA_{22} \\ \Delta'_2 = I + B^*B - iB^*A'_{11}B + iA'_{22} \end{cases} \text{and} \begin{cases} A_{11} = \sum_{q=1}^{Q} \alpha_q A_{11,q}, A_{22} = \sum_{q=1}^{Q} \alpha_q A_{22,q} \\ A'_{11} = \sum_{q=1}^{Q} \alpha'_q A_{11,q}, A'_{22} = \sum_{q=1}^{Q} \alpha'_q A_{22,q} \end{cases}$$

When r is large, the discrete sets from which α_qs, α'_qs are chosen from ($\mathcal{A}_r$) can be replaced with independent scalar Cauchy distributions. And by noticing that the sum of two independent Cauchy random variables is scaled-Cauchy, our criterion can be simplified to

$$\max_{\{A_{11,q}, A_{22,q}\}, B} \mathrm{E}\,\log\det(B^*A_{11}B - A_{22})^2 - 2\mathrm{E}\,\log\det\Delta_2^2 \tag{36}$$

Choosing the Frobenius norm of the basis matrices

The entries of the $A_{11,q}$s and $A_{22,q}$s in (35) are unconstrained other than that they must be Hermitian matrices. However, we found that it is beneficial to constrain the Frobenius norm of all the matrices in $\{A_{11,q}\}$ to be the same, which we denote by γ_1. Similarly, for the matrices $\{A_{22,q}\}$, whose Frobenius norm we denote by γ_2. In fact, in our experience it is very important, for both the criterion function (35) and the ultimate constellation performance, that the correct Frobenius norms of the basis matrices be chosen. The gradients for the Frobenius norms γ_1 and γ_2 are given in Appendix C and gradient-ascent method is used. Since the optimization of B is too complicated to be done by gradient-ascent method and simulation shows that the Frobenius norm of B, and B itself, do not have significant effects on the performance as long as B is full rank, we choose B to be $\gamma_3[I_M, 0_{M\times(T-2M)}]$ with γ_3 close to 1. This has shown to perform well.

2.5.4 Design Summary

We now summarize the design method for a Cayley unitary space-time code with M transmit antennas and N receive antennas, and target rate R.

1. Choose $Q \le \min(T-M, N)\max(2(T-M)-N, T-M)$. Although this inequality is a soft limit for sphere decoding, we choose our Q that obeys the inequality to keep the decoding complexity polynomial.

2. Choose r that satisfies $r^Q = 2^{RT}$. We always choose r to be a power of 2 to simplify the bit allocation and use a standard Gray-code assignment of bits to the symbols of the sets $\mathcal{A}_r$.

3. Let $\mathcal{A}_r$ be the r-point discretization of the scalar Cauchy distribution obtained as the image of the function $\alpha = -\tan(\theta/2)$ applied to the set $\{\pi/r, 3\pi/r, 5\pi/r, ..., (2r-1)\pi/r\}$.

4. Choose $\{A_{11,q}\}$ and $\{A_{22,q}\}$ that solves the optimization problem (35). A gradient-ascent method can be used. The computation of the gradients of the criterion in (35) is presented in Appendix B. At the end of each iteration, gradient-ascent is used to optimize the Frobenius norms of the basis matrices $A_{11,1}, A_{11,2}, \cdots, A_{11,Q}$ and $A_{22,1}, A_{22,2}, \cdots, A_{22,Q}$. The computation of the gradients is given in Appendix C. Note first that the solution to (35) is highly non-unique. Another solution can be obtained by simply reordering the $A_{11,q}$s and $A_{22,q}$s. And also, since the criterion function is neither linear nor convex in the design variables $A_{11,q}$ and $A_{22,q}$, there is no guarantee of obtaining a global maximum. However, since the code design is performed off-line and only once, we can use more sophisticated optimization techniques to get a better solution. Simulation results show that the codes obtained by this method have good performance. The number of receive antennas N does not appear explicitly in the criterion (35), but it depends on N through the choice of Q. Hence, the optimal codes, for a given M, are different for different N.

3 Simulation results

In this section, we give examples of Cayley unitary space-time codes and the simulated performance of the codes for various number of antennas and rates. The fading coefficient from each transmit antenna to each receive antenna is modeled independently as a complex Gaussian variable with zero mean and unit variance and is kept constant for T channel uses. At each time, a zero-mean, unit-variance complex Gaussian noise is added to each receive antenna. Two error events of interest are demonstrated including block errors, which correspond to errors in decoding the $T \times M$ matrices $S_1, ..., S_L$, and bit errors, which correspond to errors in decoding $\alpha_1, ..., \alpha_Q$. The bits are allocated to each α_q by a Gray code and therefore, a block error may correspond to only a few bit errors. We first give an example to compare the performance of the linearized ML, which is given by (15) with that of the true ML, then performance comparisons of our codes with training-based methods are given.

3.1 Linearized ML vs. ML

In communications and code designs, the decoding complexity is an important issue. In our problem, when the transmission rate is high, for example, $R = 3$ and $T = 6, M = 3$, for one coherence interval, the true ML decoding involves a search over $2^{RT} = 2^{18} = 262,144$ 6×3 matrices, which is not practical. This is why we linearize the ML decoding to use the sphere decoding algorithm.

But we need to know what is the penalty of using (15) instead of the true ML. Here an example is given for the case of a two transmit, one receive antenna system with coherence interval of four channel uses operating at rate $R = 1.5$ with $Q = 3$ and $r = 2$. The number of signal matrices is $2^{RT} = 64$, for which

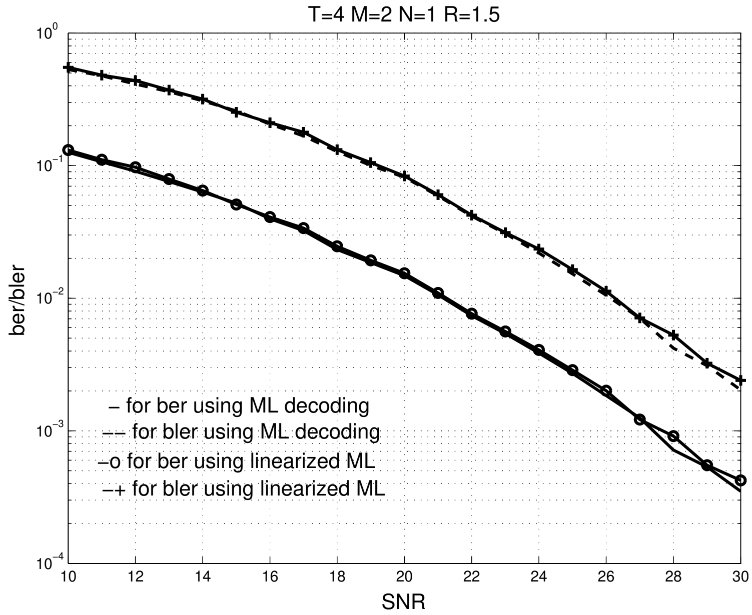

Figure 1: $T = 4, M = 2, N = 1, R = 1.5$: ber and bler of the linearized ML give by (15) compared with the true ML.

the true ML is feasible. The resulting bit error rate and block error rate curves for the linearized ML are the line with circles and line with stars in Fig. 1. The resulting bit error rate and block error rate curves for the the true ML are the solid line and the dashed line in the figure. We can see from Fig. 1 that the performance loss for the linearized ML decoding is almost neglectable but the computational complexity is saved greatly by using the linearized ML decoding which is implemented by sphere decoder.

3.2 Cayley unitary space-time codes vs. training-based codes

In this section a few examples of the Cayley codes for various multiple antenna communication systems are given and their performance compared with that of the training-based codes is also showed.

As discussed in the introduction, a commonly used scheme for unknown channel multiple antenna communication systems is to obtain the channel information via training. It is important and meaningful to compare our code with that of the training codes. Training-based schemes and the optimal way to do training are discussed in section 1.2. In most of the following simulations, different space-time codes are used in the data transmission phase for different system settings. Sphere decoding is used in decoding all the Cayley codes and the decoding of the training-based codes is always ML, but the algorithm varies according to the codes used. The details of the codes used (the basis matrices, etc.) can be obtained by contacting the authors.

Example of $T = 4, M = 2, N = 2$

The first example is for the case of two transmit and two receive antennas with coherence interval $T = 4$. For the training-based schemes, half of the coherence interval is used for training. For the data transmission phase, we consider two different space-time codes. The first one is the well-known orthogonal design in which the transmitted data matrix has the following structure.

$$S_d = \begin{bmatrix} a & b \\ -\bar{b} & \bar{a} \end{bmatrix}$$

By choosing a and b from the signal set of 16-QAM equally likely, the rate of the training-based code is 2 bits per channel use. The same as the Cayley codes, bits are allocated to each entry by the Gray code. The second one is the LD code proposed in [5]:

$$S_d = \sum_{q=1}^{4} (\alpha_q A_q + i\beta_q B_q), \quad \alpha_q, \beta_q \in \{-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\}$$

where

$$A_1 = B_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, A_2 = B_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, A_3 = B_3 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, A_4 = B_4 = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

Clearly, the rate of the training-based LD code is also 2. For the Cayley code, from (30), we choose $Q = 4$. To attain rate 2, $r = 4$ from (32). The Cayley code was obtained by finding a local maximum to (36).

The performance curves are shown in Fig. 2. The solid line with circles/stars indicates the ber/bler of the Cayley code at rate 2. The solid/dashed line indicates the ber/bler of the training-based orthogonal design at rate 2 and the solid line with plus signs/dash-dotted line shows the ber/bler of the training-based LD code at rate 2. We can see from the figure that the Cayley code underperforms the optimal training-based codes by $3-4$dB. However, our results are preliminary and it is conceivable that better performance may be obtained by further optimization of (35) or (36).

Example of $T = 5, M = 2, N = 1$

For the training-based scheme of this setting, 2 channel uses of each coherence interval are allocated to training. Therefore, in the data transmission phase, bits are encoded into a 3×2 data matrix S_d. Since we are not aware of any 3×2 space-time code, we employ an uncoded transmission scheme, where each element of S_d is chosen independently from a BPSK constellation, resulting in rate $6/5$. This allows us to compare the Cayley codes with the the uncoded training-based scheme. Two Cayley codes are analyzed here: the Cayley code at rate 1 with $Q = 5, r = 2$ and the Cayley code at rate 2 with $Q = 5, r = 4$.

The performance curves are shown in Fig. 3. The solid line with circles/stars indicates the ber/bler of the Cayley code at rate 1, solid line with plus signs/the dash-dotted line shows the ber/bler of the Cayley code at rate 2 and the solid/dashed line shows the ber and bler of the training-based scheme, which has a rate

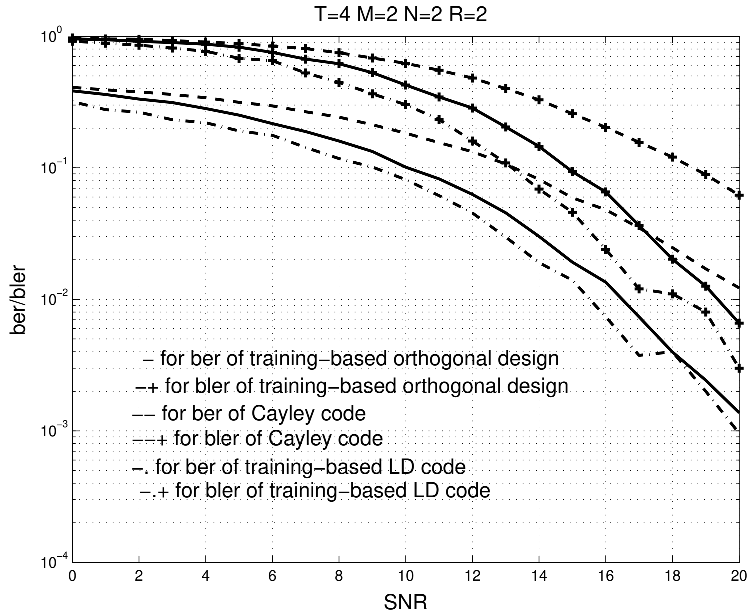

Figure 2: $T = 4, M = 2, N = 2, R = 2$: ber and bler of the Cayley code compared with the training-based orthogonal design and the training-based LD code.

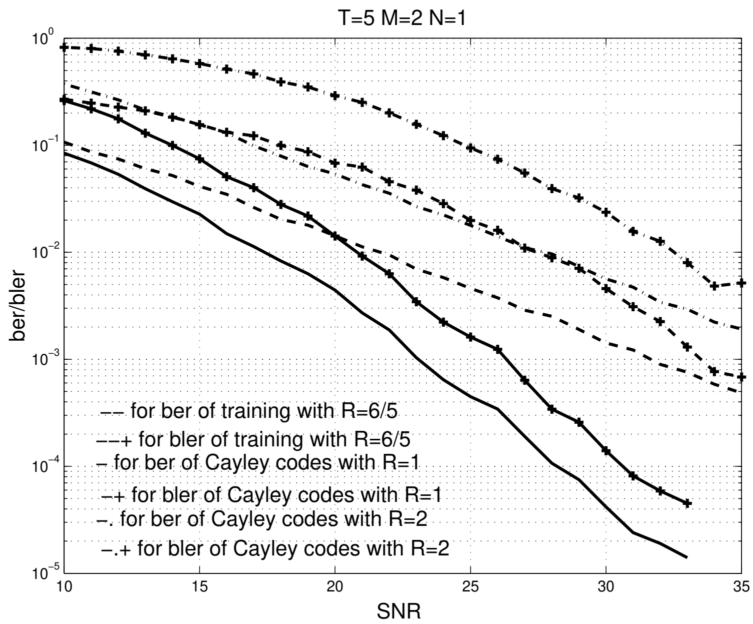

Figure 3: $T = 5, M = 2, N = 1$: ber and bler of the Cayley codes compared with the uncoded training-based scheme.

of $6/5$. Exhaustive search is used in decoding the training-based scheme and sphere decoding is applied to decode the Cayley codes.

We can see that our Cayley code at rate 1 has lower ber and bler than the training-based scheme at rate $6/5$ at any SNR. And, even at a rate which is $4/5$ higher, 2 compared with $6/5$, the performance of the Cayley code is comparable to that of the training-based scheme when the SNR is as high as 35.

Example of $T = 7, M = 3, N = 1$

For this system setting, 3 channel uses of each coherence interval are allocated to training. In the data transmission phase of the training-based scheme, we use the optimized LD code given in [5]:

$$S_d = \begin{bmatrix} \alpha_1+\alpha_3+i\left[\frac{\beta_2+\beta_3}{\sqrt{2}}+\beta_4\right] & \frac{\alpha_2-\alpha_4}{\sqrt{2}}-i\left[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}\right] & 0 \\ \frac{-\alpha_2+\alpha_4}{\sqrt{2}}-i\left[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}\right] & \alpha_1-i\frac{\beta_2+\beta_3}{\sqrt{2}} & -\frac{\alpha_2+\alpha_4}{\sqrt{2}}+i\left[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}\right] \\ 0 & \frac{\alpha_2+\alpha_4}{\sqrt{2}}+i\left[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}\right] & \alpha_1-\alpha_3+i\left[\frac{\beta_2+\beta_3}{\sqrt{2}}-\beta_4\right] \\ \frac{\alpha_2-\alpha_4}{\sqrt{2}}+i\left[\frac{\beta_1}{\sqrt{2}}+\frac{\beta_2-\beta_3}{2}\right] & -\alpha_3+i\beta_4 & -\frac{\alpha_2+\alpha_4}{\sqrt{2}}+i\left[\frac{\beta_1}{\sqrt{2}}-\frac{\beta_2-\beta_3}{2}\right] \end{bmatrix}.$$

By setting α_i, β_i in BPSK, we obtain a LD code at rate $8/7$. For the Cayley code, we choose $Q = 7$ and $r = 2$ and the rate of the code is 1.

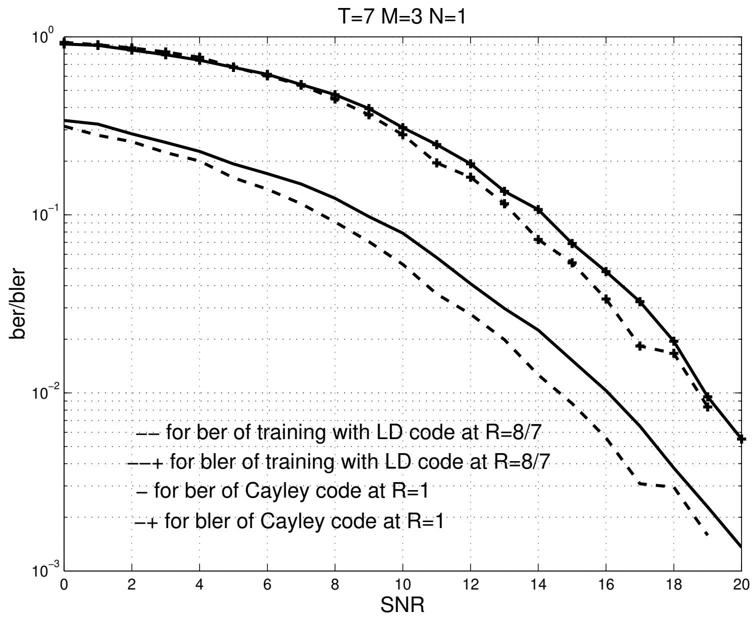

Figure 4: $T = 7, M = 3, N = 1$: ber and bler of the Cayley code compared with the training-based LD codes.

The performance curves are shown in Fig. 4. The solid line with circles/stars indicates the ber/bler of the Cayley code at rate 1, and the solid/dashed line shows the ber/bler of the training-based LD code, which has a rate of $8/7$. Sphere decoding is applied in the decoding of both codes. From the figure we can see that the performance of the Cayley code is close to the performance of the training-based LD code. Therefore, at a rate $1/7$ lower, the Cayley code is comparable with the training-based LD code. Again, our results are preliminary and further optimization of (35) or (36) may yield improved performance.

4 Conclusion

Cayley unitary space-time codes are developed in this paper. The codes do not require channel knowledge at either the transmitter or the receiver, are simple to encode and decode, apply to any combination of transmit and receive antennas. They are designed with a probabilistic criterion: they maximize the expected log-determinant of the difference between matrix pairs. The Cayley transform is used to construct the codes because it maps the nonlinear Stiefel manifold of unitary matrices to the linear space of skew-Hermitian matrices. The transmitted data is broken into substreams $\alpha_1, ..., \alpha_Q$, then linearly encoded in the Cayley transform domain. We showed that by constraining $A_{12} = (I + iA_{11})B$ and ignoring the data dependence of the additive noise, $\alpha_1, ..., \alpha_Q$ appear linearly at the receiver. Therefore, linear decoding algorithms such as sphere decoding and nulling and cancelling can be used in polynomial time. Our code has a similar structure as training-based schemes after transformations.

The recipe for designing Cayley unitary space-time codes for any combination of transmit/receive antennas and coherence intervals is given and also, simulation examples are shown to compare our Cayley codes with optimized training-based space-time codes and uncoded training-based schemes for different system settings. Our simulation results are preliminary, but indicates that the Cayley codes generated with this recipe slightly underperform optimized training-based schemes using orthogonal designs and/or LD codes. However, they are clearly superior to uncoded training-based space-time schemes. Further optimization of the Cayley code basis matrices (in (35 or (35)) is necessary for a complete comparison of the performance with training-based schemes.

A Proof of Theorem 3

Theorem 3 (Difference of unitary complements of the transmitted signal). *The difference of the unitary complements $S^\perp$ and $\hat{S}^\perp$ of the transmitted signals S and $\hat{S}$ can be written as*

$$S^\perp - \hat{S}^\perp = 2 \begin{bmatrix} -iB \\ I \end{bmatrix} \Delta_2^{-1}(\hat{\Delta}_2 - \Delta_2)\hat{\Delta}_2^{-1}$$

where Δ_2 and $\hat{\Delta}_2$ are the corresponding Schur complements.

Proof: First by simple algebra, $\Delta_1^{-1}A_{12}(I + iA_{22})^{-1} = (I + iA_{11})^{-1}A_{12}\Delta_2^{-1}$ can be proved, which is equivalent to $B\Delta_2^{-1} = \Delta_1^{-1}A_{12}(I + iA_{22})^{-1}$. From (24),

$$S^\perp = \begin{bmatrix} -2iB\Delta_2^{-1} \\ 2\Delta_2^{-1} - I \end{bmatrix} = \begin{bmatrix} -2iB \\ 2I - \Delta_2 \end{bmatrix} \Delta_2^{-1}$$

and

$$S^{\perp} = \begin{bmatrix} -2i\Delta_1^{-1}A_{12}(I+iA_{22})^{-1} \\ 2\Delta_2^{-1} - I \end{bmatrix} = \begin{bmatrix} \Delta_1^{-1} & 0 \\ 0 & \Delta_2^{-1} \end{bmatrix} \begin{bmatrix} -2iA_{12}(I+iA_{22})^{-1} \\ 2I - \Delta_2 \end{bmatrix}$$

Therefore,

$$\begin{aligned} S^{\perp} - \hat{S}^{\perp} &= \begin{bmatrix} \Delta_1^{-1} & 0 \\ 0 & \Delta_2^{-1} \end{bmatrix} \left(\begin{bmatrix} -2iA_{12}(I+iA_{22})^{-1} \\ 2I - \Delta_2 \end{bmatrix} \hat{\Delta}_2 - \begin{bmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{bmatrix} \begin{bmatrix} -2iB \\ 2I - \hat{\Delta}_2 \end{bmatrix} \right) \hat{\Delta}_2^{-1} \\ &= \begin{bmatrix} \Delta_1^{-1} & 0 \\ 0 & \Delta_2^{-1} \end{bmatrix} \begin{bmatrix} -2iA_{12}(I+iA_{22})^{-1}\hat{\Delta}_2 + 2i\Delta_1 B \\ 2\hat{\Delta}_2 - \Delta_2\hat{\Delta}_2 - 2\Delta_2 + \Delta_2\hat{\Delta}_2 \end{bmatrix} \hat{\Delta}_2^{-1} \\ &= \begin{bmatrix} \Delta_1^{-1} & 0 \\ 0 & \Delta_2^{-1} \end{bmatrix} \begin{bmatrix} 2iA_{12}(I+iA_{22})^{-1}\Delta_2 - 2iA_{12}(I+iA_{22})^{-1}\hat{\Delta}_2 \\ 2(\hat{\Delta}_2 - \Delta_2) \end{bmatrix} \hat{\Delta}_2^{-1} \\ &= \begin{bmatrix} -2i\Delta_1^{-1}A_{12}(I+iA_{22})^{-1} & 0 \\ 0 & 2\Delta_2^{-1} \end{bmatrix} \begin{bmatrix} \hat{\Delta}_2 - \Delta_2 \\ \hat{\Delta}_2 - \Delta_2 \end{bmatrix} \hat{\Delta}_2^{-1} \\ &= \begin{bmatrix} -2iB\Delta_2^{-1} \\ 2\Delta_2^{-1} \end{bmatrix} (\hat{\Delta}_2 - \Delta_2)\hat{\Delta}_2^{-1} \\ &= 2 \begin{bmatrix} -iB \\ I \end{bmatrix} \Delta_2^{-1}(\hat{\Delta}_2 - \Delta_2)\hat{\Delta}_2^{-1} \end{aligned}$$

□

B Gradient of criterion (35)

In the simulation presented in this paper the maximization of the design criterion function (35) is performed using a simple gradient-ascent method. In this section, we compute the gradient of (35) that this method requires.

We are interested in the gradient with respect to the matrices $A_{11,1}, ..., A_{11,Q}$ and $A_{22,1}, ..., A_{22,Q}$ of the design function (35), which is equivalent to

$$\max_{\{A_{11,q}, A_{22,q}\}, B} \mathrm{E} \log\det[B^*(A_{11} - A'_{11})B - (A_{22} - A'_{22})]^2 - 2\mathrm{E} \log\det \Delta_2^2 \tag{B.1}$$

To compute the gradient of a real function $f(A_q)$ with respect to the entries of the Hermitian matrix A_q, we use the formulas

$$\left[\frac{\partial f(A_q)}{\partial Re A_q}\right]_{j,k} = \min_{\delta\to 0} \frac{1}{\delta}[f(A_q + \delta(e_j e_k^t + e_k e_j^t)) - f(A_q)], j \neq k \tag{B.2}$$

$$\left[\frac{\partial f(A_q)}{\partial Im A_q}\right]_{j,k} = \min_{\delta\to 0} \frac{1}{\delta}[f(A_q + i\delta(e_j e_k^t - e_k e_j^t)) - f(A_q)], j \neq k \tag{B.3}$$

$$\left[\frac{\partial f(A_q)}{\partial A_q}\right]_{j,j} = \min_{\delta\to 0} \frac{1}{\delta}[f(A_q + \delta e_j e_j^t) - f(A_q)] \tag{B.4}$$

where e_j is the unit column vector of the same dimension of columns of A_q who has a one in the jth entry and zeros elsewhere. That is, when we calculating the gradient with respect to $A_{11,q}$, e_j should has dimension M and for the gradient with respect to $A_{22,q}$, it is $T-M$ instead. e_j^t means the transpose of e_j.

First, note that $A_{11}-A'_{11}=\sum_{q=1}^{Q}A_{11,q}a_q$ where $a_q=\alpha_q-\alpha'_q$ and similarly, $A_{22}-A'_{22}=\sum_{q=1}^{Q}A_{22,q}a_q$. Therefore, to apply (B.2) to the first term of (B.1) with respect to $A_{11,q}$, let $H=B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})$ and we compute

$$\begin{aligned}
& \log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})+B^*(e_je_k^t+e_ke_j^t)B\delta a_q]^2 \\
= & \log\det\{H^2+[HB^*(e_je_k^t+e_ke_j^t)B+B^*(e_je_k^t+e_ke_j^t)BH]\delta a_q+o(\delta^2)I\} \\
= & \log\det H^2+\log\det\{I+H^{-2}[HB^*(e_je_k^t+e_ke_j^t)B+B^*(e_je_k^t+e_ke_j^t)BH]\delta a_q+o(\delta^2)I\} \\
= & \log\det H^2+\mathrm{tr}\,\{H^{-2}[HB^*(e_je_k^t+e_ke_j^t)B+B^*(e_je_k^t+e_ke_j^t)BH]\delta a_q\}+o(\delta^2) \\
= & \log\det H^2+\mathrm{tr}\,\{H^{-1}B^*(e_je_k^t+e_ke_j^t)B+H*-2B^*(e_je_k^t+e_ke_j^t)BH\}\delta a_q\}+o(\delta^2) \\
= & \log\det H^2+\mathrm{tr}\,\{BH^{-1}B^*(e_je_k^t+e_ke_j^t)+BH^{-1}B^*(e_je_k^t+e_ke_j^t)\}\delta a_q\}+o(\delta^2) \\
= & \log\det H^2+(2\{BH^{-1}B\}_{k,j}+2\{BH^{-1}B\}_{j,k})a_q+o(\delta^2) \\
= & \log\det H^2+4Re\{BH^{-1}B^*\}_{j,k}a_q+o(\delta^2)
\end{aligned}$$

We use $\mathrm{tr}\,AB=\mathrm{tr}\,BA$ and the last equality follows because $BH^{-1}B^*$ is Hermitian. We may now apply (B.2) to obtain

$$\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial ReA_{11,q}}\right]_{j,k}=4\mathrm{E}\,Re\{BH^{-1}B^*\}_{j,k}a_q,\ \ j\neq k$$

The gradient with respect to the imaginary components of $A_{11,q}$ can be obtained in a similar way as the following

$$\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial ImA_{11,q}}\right]_{j,k}=4\mathrm{E}\,Im\{BH^{-1}B^*\}_{j,k}a_q,\ \ j\neq k$$

And the gradient with respect to the diagonal elements is

$$\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial A_{11,q}}\right]_{j,j}=2\mathrm{E}\,\{BH^{-1}B^*\}_{j,j}a_q$$

Similarly, we get the gradient with respect to $A_{22,q}$,

$$\begin{aligned}
\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial ReA_{22,q}}\right]_{j,k} &= -4\mathrm{E}\,ReH^{-1}_{j,k}a_q,\ \ j\neq k \\
\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial ImA_{22,q}}\right]_{j,k} &= -4\mathrm{E}\,ImH^{-1}_{j,k}a_q,\ \ j\neq k \\
\left[\frac{\partial\log\det[B^*(A_{11}-A'_{11})B-(A_{22}-A'_{22})]^2}{\partial A_{22,q}}\right]_{j,j} &= -2\mathrm{E}\,H^{-1}_{j,j}a_q
\end{aligned}$$

For the second term, by using the same method, the following results are obtained

$$\begin{aligned}
\left[\frac{\partial \log\det \Delta_2^2}{\partial Re A_{11,q}}\right]_{j,k} &= 2\mathrm{E}\, Re(D + D^* + E + E^*)_{j,k}\alpha_q, \ \ j \neq k \\
\left[\frac{\partial \log\det \Delta_2^2}{\partial Im A_{11,q}}\right]_{j,k} &= 2\mathrm{E}\, Im(D + D^* + E + E^*)_{j,k}\alpha_q, \ \ j \neq k \\
\left[\frac{\partial \log\det \Delta_2^2}{\partial A_{11,q}}\right]_{j,j} &= 2\mathrm{E}\,(D + E)_{j,j}\alpha_q \\
\left[\frac{\partial \log\det \Delta_2^2}{\partial Re A_{22,q}}\right]_{j,k} &= 2\mathrm{E}\, Re(F + F^* + G + G^*)_{j,k}\alpha_q, \ \ j \neq k \\
\left[\frac{\partial \log\det \Delta_2^2}{\partial Im A_{22,q}}\right]_{j,k} &= 2\mathrm{E}\, Im(F + F^* + G + G^*)_{j,k}\alpha_q, \ \ j \neq k \\
\left[\frac{\partial \log\det \Delta_2^2}{\partial A_{22,q}}\right]_{j,j} &= 2\mathrm{E}\,(F + G)_{j,j}\alpha_a
\end{aligned}$$

where

$$\begin{aligned}
D &= iB\Delta_2^{-2}(I + iA_{22})B^* \\
E &= iB\Delta_2^{-2}B^*(I - iA_{11})BB^* \\
F &= \Delta_2^{-2}A_{22} \\
G &= iB^*(I + iA_{11})B\Delta_2^{-2}
\end{aligned}$$

and all the expectations are over all possible $\alpha_1, ..., \alpha_Q$.

C Gradient of Frobenius norms of the basis sets

Let γ_1 be a multiplicative factor that we use to multiple every $A_{11,q}$ and γ_2 a multiplicative factor that we use to multiple every $A_{22,q}$. Thus, γ_1^2 and γ_2^2 are the Frobenius norms of matrices in $\{A_{11,q}\}$ and $\{A_{22,q}\}$. We solve for the optimal $\gamma_1, \gamma_2 > 0$ by maximizing the criterion function in (35)

$$\xi(\gamma_1, \gamma_2) = \mathrm{E}\ \log\det[\gamma_1 B^*(A_{11} - A'_{11})B - \gamma_2(A_{22} - A'_{22})]^2 - 2\mathrm{E}\ \log\det \Delta_2^2$$

where

$$\Delta_2 = I + B^*B - i\gamma_1 B^* \sum_{q=1}^{Q} \alpha_q A_{11,q} B + i\gamma_2 \sum_{q=1}^{Q_1} \alpha_q A_{22,q}$$

As in the optimization of $A_{11,q}, A_{22,q}$, gradient-ascent method is used. To compute the gradient of a real function $f(x1, x2)$ with respect to x_1 and x_2, we use the formulas

$$\begin{aligned}
\frac{\partial f(x_1, x_2)}{\partial x_1} &= \lim_{\delta \to 0} \frac{1}{\delta}[f(x_1 + \delta, x_2) - f(x_1, x_2)] \\
\frac{\partial f(x_1, x_2)}{\partial x_2} &= \lim_{\delta \to 0} \frac{1}{\delta}[f(x_1, x_2 + \delta) - f(x_1, x_2)]
\end{aligned}$$

And the results are

$$\begin{aligned}
\frac{\partial\xi(\gamma_1,\gamma_2)}{\partial\gamma_1} &= -2\mathrm{E}\,\mathrm{tr}\,\{f^{-1}[2\gamma_1 B^*A_{11}BB^*A_{11}B + iB^*(BB^*A_{11} - A_{11}BB^*)B \\
&\quad -\gamma_2(A_{22}B^*A_{11}B + A_{11}BA_{22}B^*)]\} \\
&\quad +\mathrm{E}\,\mathrm{tr}\,[g^{-1}(2\gamma_1 B^*(A_{11} - A'_{11})BB^*(A_{11} - A'_{11})B \\
&\quad -\gamma_2((A_{22} - A'_{22})B^*(A_{11} - A'_{11})B + (A_{11} - A'_{11})B(A_{22} - A'_{22})B^*))] \\
\frac{\partial\xi(\gamma_1,\gamma_2)}{\partial\gamma_2} &= -2\mathrm{E}\,\mathrm{tr}\,[f^{-1}(2\gamma_1 A_{22}^2 - i(B^*BA_{22} + A_{22}BB^*) - \gamma_1(A_{22}B^*A_{11}B + A_{11}BA_{22}B^*))] \\
&\quad \mathrm{E}\,\mathrm{tr}\,[g^{-1}(2\gamma_2 A_{22}^2 - \gamma_1((A_{22} - A'_{22})B^*(A_{11} - A'_{11})B \\
&\quad +(A_{11} - A'_{11})B(A_{22} - A'_{22})B^*))]
\end{aligned}$$

Simulation shows that good performance is obtained when γ_1 and γ_2 are not too far away from unity.

Acknowledgment

This work was supported by an AFOSR for MURI “Mathematical Infrastructure for Robust Virtual Engineering,” URI “Protecting Infrastructures from Themselves” and Caltech’s Lee Center for Advanced Networking.

References

[1] G. J. Foschini, “Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas,” *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[2] I. E. Telatar, “Capacity of multi-antenna gaussian channels,” *Eur. Trans. Telecom.*, vol. 10, pp. 585–595, Nov. 1999.

[3] S. M. Alamouti, “A simple transmitter diversity scheme for wireless communications,” *IEEE J. Sel. Area Comm.*, pp. 1451–1458, Oct. 1998.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, “Space-time codes for high data rate wireless communication: Performance criterion and code construction,” *IEEE Trans. Info. Theory*, vol. 44, pp. 744–765, 1998.

[5] B. Hassibi and B. Hochwald, “High-rate codes that are linear in space and time,” *IEEE Trans. Info. Theory*, vol. 48, pp. 1804–1824, July 2002. Download available at `http://mars.bell-labs.com`.

[6] T. L. Marzetta, “Blast training: Estimating channel characteristics for high-capacity space-time wireless,” in *Proc. 37th Annual Allerton Conference on Communications, Control, and Computing*, Sept. 22–24 1999.

[7] B. Hassibi and B. Hochwald, “How much training is needed in multiple-antenna wireless links?,” *submitted to IEEE Trans. Info. Theory*, 2000. Download available at `http://mars.bell-labs.com`.

[8] T. L. Marzetta and B. M. Hochwald, “Capacity of a mobile multiple-antenna communication link in Rayleigh flat fading,” *IEEE Trans. Info. Theory*, vol. 45, pp. 139–157, 1999.

[9] B. M. Hochwald and T. L. Marzetta, "Unitary space-time modulation for multiple-antenna communication in Rayleigh flat-fading," *IEEE Trans. Info. Theory*, vol. 46, pp. 543–564, Mar. 2000.

[10] L. Zheng and D. Tse, "Communication on the Grassman manifold: a geometric approach to the noncoherent multiple-antenna channel," *IEEE Trans. Info. Theory*, vol. 48, pp. 359–383, 2002.

[11] B. Hassibi and T. L. Marzetta, "Multiple-antennas and isotropically-random unitary inputs: The received signal density in closed-form," *IEEE Trans. on Info. Thy.*, vol. 48, pp. 1473–1484, 2002.

[12] B. Hochwald, T. Marzetta, and B. Hassibi, "Space-time autocoding," *IEEE Trans. Info. Theory*, pp. 2761–2781, Nov 2001.

[13] B. Hochwald, T. Marzetta, T. Richardson, W. Sweldens, and R. Urbanke, "Systematic design of unitary space-time constellations," *IEEE Trans. Info. Theory*, pp. 1962–1973, 2000.

[14] T. Marzetta, B. Hassibi, and B. Hochwald, "Structured unitary space-time constellations," *IEEE Trans. Info. Theory*, pp. 942–950, 2002.

[15] S. Gaulliou, I. Kammoun, and J. Belfiore, "Space-time codes for the glrt noncoherent detector," *in ISIT 2002*.

[16] B. Hassibi and B. Hochwald, "Cayley differential unitary spce-time codes," *IEEE Trans. Info. Theory*, pp. 1458–1503, 2002.

[17] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electronic Letters*, vol. 35, pp. 14–16, Jan. 1999.

[18] B. Hassibi, "An efficient square-root algorithm for BLAST," *submitted to IEEE Trans. Sig. Proc.*, 1999. Download available at `http://mars.bell-labs.com`.

[19] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, vol. 44, pp. 463–471, April 1985.

[20] M. O. Damen, A. Chkeif, and J.-C. Belfiore, "Lattice code decoder for space-time codes," *IEEE Comm. Let.*, pp. 161–163, May 2000.

[21] R. Horn and C. Johnson, *Topics in Matrix Analysis*. Cambridge: University Press, 1991.