

Efficient Joint Maximum-Likelihood Channel Estimation and Signal Detection

H. Vikalo[‡], B. Hassibi[‡] and P. Stoica[§]

Abstract

In wireless communication systems, channel state information is often assumed to be available at the receiver. Traditionally, a training sequence is used to obtain the estimate of the channel. Alternatively, the channel can be identified using known properties of the transmitted signal. However, the computational effort required to find the joint ML solution to the symbol detection and channel estimation problem increases exponentially with the dimension of the problem. To significantly reduce this computational effort, we formulate the joint ML estimation and detection as an integer least-squares problem, and show that it can be solved via sphere decoding in polynomial expected time.

Index Terms—Integer least-squares problem, sphere decoding, wireless communications, multiple-antenna systems, lattice problems, NP hard, expected complexity, joint detection and estimation

[‡](hvikalo,hassibi)@systems.caltech.edu; California Institute of Technology, Pasadena, CA 91125

This work was supported in part by the NSF under grant no. CCR-0133818, by the Office of Naval Research under grant no. N00014-02-1-0578, and by Caltech's Lee Center for Advanced Networking.

[§]Uppsala University, Department of Systems and Control, Uppsala, Sweden

1 Introduction

The pursuit for high-speed data services has resulted in a tremendous amount of research activity in the wireless communications community. To obtain high reliability of the transmission, particular attention has been paid to the design of receivers (see, e.g., [1] and references therein).

In the system design, one often assumes knowledge of the channel coefficients at the receiver. These are typically obtained by sending a training sequence, thus sacrificing a fraction of the transmission rate. On the other hand, in practical systems, due to rapid changes of the channel and/or limited resources, training and channel tracking may be infeasible. One possible remedy is to differentially encode the transmitted data and thus eliminate the need for the channel knowledge. Another one is to exploit known properties of the transmitted data to learn the channel blindly – for instance, one can exploit the fact that the transmitted data belongs to a finite alphabet.

We consider a problem of joint maximum-likelihood (ML) channel estimation and signal detection in a communication system where the transmitters use only one antenna but the receiver employs multiple antennas. This can, for instance, be the case on an uplink, where all users (transmitters) have a single antenna, while the base station (receiver) has multiple antennas. Let N denotes the number of receive antennas and let T be the time interval during which the channel coefficients remains constant and change to an independent set of values afterwards (i.e., we assume the block-fading channel model). The received signal can be written as

$$X = HS^* + W = H \begin{bmatrix} S_\tau^* & S_d^* \end{bmatrix} + W, \quad (1)$$

where $H = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \dots & \mathbf{h}_M \end{bmatrix}$ is the $N \times M$ channel matrix comprised of N single-input multi-output (SIMO) channel gain vectors $\mathbf{h}_i, i = 1, 2, \dots, M$, whose entries are independent, identically distributed (iid) complex Gaussian random variables $\mathcal{CN}(0, 1)$. Matrix S_τ is the (deterministic and known) $T_\tau \times M$ matrix of training symbols, while the matrix S_d is $T_d \times M$ matrix of

data symbols, and where $T = T_\tau + T_d$. We assume that all elements of S_τ and S_d are unitary, i.e.,

$$\begin{aligned} |s_\tau^{ij}|^2 &= 1, \quad 1 \leq i \leq T_\tau, 1 \leq j \leq M, \\ |s_d^{ij}|^2 &= 1, \quad 1 \leq i \leq T_d, 1 \leq j \leq M, \end{aligned} \tag{2}$$

where s_τ^{ij} denotes an (i, j) entry in S_τ , and where s_d^{ij} denotes an (i, j) entry in S_d . Furthermore, $N \times T$ matrix W is an additive noise matrix whose elements are assumed to be iid complex Gaussian random variables $\mathcal{C}(0, \sigma^2)$.

The joint ML channel estimation and signal detection problem can be stated as follows:

$$\min_{H, S} \|X - HS^*\|^2, \tag{3}$$

where the entries of S satisfy (2). Problem (3) is a mixed optimization problem: it is a least-squares problem in H and an *integer least-squares* problem in S . Traditionally, the solution to the integer least-squares problems is found by an exhaustive search over the entire symbol space. The complexity of exhaustive search is exponential in MT_d and often infeasible in practice. Therefore, low-complexity heuristic techniques, usually iterating between the S and the estimate of H , are often employed (see, e.g., [2] and the references therein). On the other hand, in communication applications, the sphere decoding [3] is recognized as a technique for solving integer least-squares problems at polynomial expected complexity [4, 5].

In this paper, we show how the sphere decoding algorithm can be employed to find the solution of (3), and prove that, at high SNR, the expected complexity results of [4, 5] hold. In other words, we show that the joint ML channel estimation and symbol detection problem (3) can be solved in polynomial expected time over a wide range of system parameters. We should remark that the basic sphere decoding algorithm performs the closest point search in a rectangular lattice and the available expected complexity results assume the same. Therefore, when discussing complexity, we will assume that the entries of S belongs to a QPSK constellation (so that (2) hold). However,

sphere decoding can easily be modified and used for detection of symbols coming from a PSK-modulation schemes [6].

Note that the model (1) include both the blind and the training-based scheme – blind scheme model is obtained by simply setting $T_\tau = 0$. In this paper, we consider both schemes. We start by considering single user system ($M = 1$), for which we solve (3) for both $T_\tau = 0$ and $T_\tau > 0$. This is presented in Section 2, while Section 3 contains generalization of the $T_\tau > 0$ case to the multi-user scenario. We discuss only training-based multi-user case in order to avoid ambiguities that arise in blind scenarios. An interesting result of Section 3 implies that the complexity of the algorithm grows only linearly with the number of users. The complexity of the sphere decoding algorithm for solving (3) is discussed in Section 3, and the results therein imply polynomial expected complexity of the algorithm over a wide range of system parameters. In Section 5, we briefly discuss use of soft sphere decoding for joint detection and decoding in systems employing channel codes. Simulation results are presented in Section 6, while the summary and conclusion are in Section 7.

2 Single user case

We consider a single-user case, i.e., consider the model (1) for $M = 1$. Then the received signal can be written as

$$X = \mathbf{h}\mathbf{s}^* + W = \mathbf{h} \begin{bmatrix} \mathbf{s}_\tau^* & \mathbf{s}_d^* \end{bmatrix} + W, \quad (4)$$

and the optimization problem that needs to be solved is

$$\min_{\mathbf{h}, \mathbf{s}} \|\mathbf{X} - \mathbf{h}\mathbf{s}^*\|^2, \quad (5)$$

where the entries of \mathbf{s} have unit power. The blind and the training-based schemes for solving (5) are treated in the remainder of this section.

2.1 Blind joint ML estimation and detection

Assuming that there is no training sequence ($T_\tau = 0$), so that $\mathbf{s}_d = \mathbf{s}$ and the model (4) becomes

$$X = \mathbf{h}\mathbf{s}^* + W = \mathbf{h}\mathbf{s}_d^* + W. \quad (6)$$

For any given \mathbf{s} , the channel $\hat{\mathbf{h}}$ that minimizes (6) is given by

$$\hat{\mathbf{h}} = X\mathbf{s}/\|\mathbf{s}\|^2 = \frac{1}{T}X\mathbf{s}, \quad (7)$$

where we used the assumption that the entries of \mathbf{s} have unit power. Substituting (7) in (5) gives

$$\begin{aligned} \|X - \mathbf{h}\mathbf{s}^*\|^2 &= \|X(I - \frac{1}{T}\mathbf{s}\mathbf{s}^*)\|^2 \\ &= \text{tr}[X(I - \frac{1}{T}\mathbf{s}\mathbf{s}^*)X^*] \\ &= \text{tr}(XX^*) - \frac{1}{T}\mathbf{s}^*X^*X\mathbf{s} \end{aligned} \quad (8)$$

Hence the optimization (5) is equivalent to solving

$$\max_{\mathbf{s}} \mathbf{s}^*X^*X\mathbf{s}. \quad (9)$$

Optimization problem (9) is over vectors \mathbf{s} with integer values and a straightforward way to solve it is via exhaustive search [2]. On the other hand, the sphere decoding algorithm efficiently solves optimization problems over integers. However, the sphere decoding problem *minimizes* objective function over integer-valued vectors, and thus we need to express (9) accordingly. To this end, let $\hat{\lambda} = \lambda_{\max}(X^*X)$ denote the maximum eigenvalue of X^*X , and let $\rho > \hat{\lambda}$ (for instance, we can choose $\rho = \text{tr}(X^*X)$). The problem (9) is then equivalent to

$$\min_{\mathbf{s}} \mathbf{s}^*(\rho I - X^*X)\mathbf{s}. \quad (10)$$

The optimization problem (10) is an integer least-squares problem. Furthermore, note that due to the choice of ρ , the matrix $(\rho I - X^*X)$ is positive definite and therefore it allows for the Cholesky factorization of the form

$$\rho I - X^*X = R^*R, \quad (11)$$

where R is an upper-triangular matrix. Thus the sphere decoding algorithm of Fincke and Pohst [3] can be applied to solve (10). Rather than exhaustively searching over the entire symbol space, the sphere decoding algorithm performs a limited search inside a sphere of an appropriately chosen radius r , i.e., finds the point \mathbf{s}_d that minimizes (10) among all lattice points that satisfy

$$\mathbf{s}^* X^* X \mathbf{s} \leq r^2. \quad (12)$$

(The choice of the radius r in (12) will be discussed in Section 4.) The closest lattice point to the origin inside the sphere is the solution to (10).

The sphere decoding algorithm breaks down condition (12) into a set of T necessary conditions that the components of the vector \mathbf{s} need to satisfy. The algorithm can be related to nulling and canceling (see, e.g., [7]) where, after a component of the vector \mathbf{s} that satisfies (12) is found, its contribution to $\mathbf{s}^* X^* X \mathbf{s}$ is subtracted. However, unlike in nulling and canceling, in sphere decoding the components of \mathbf{s} are not fixed until an entire vector \mathbf{s} which satisfies (12) is found.

We omit any further details for brevity and refer the interested reader to [3], or to [4, 5] and the references therein. Furthermore, for an application of complex sphere decoding, relevant for detection of symbols coming from PSK constellations, we refer the reader to [6].

Finally, we should remark that the sphere decoding algorithm can be interpreted as a tree search procedure, as illustrated in Figure 1, where the nodes on the k^{th} level of the tree correspond to k -dimensional subvectors \mathbf{s}^k of the unknown vector \mathbf{s} . We will find this tree-search interpretation useful when later discussing training-based schemes.

2.2 Joint ML estimation and detection in training-based schemes

Assume that $T_\tau > 0$, i.e., training sequence \mathbf{s}_τ precedes the data block \mathbf{s}_d . The optimization problem (10) can then be written as

$$\min_{\mathbf{s}_d} \begin{bmatrix} \mathbf{s}_\tau^* & \mathbf{s}_d^* \end{bmatrix} (\rho I - X^* X) \begin{bmatrix} \mathbf{s}_\tau \\ \mathbf{s}_d \end{bmatrix}. \quad (13)$$

For simplicity, we will find it useful to define

$$\rho I - X^* X = \Gamma.$$

Recall that we choose ρ such that $\rho > \lambda_{\max}(X^* X)$. Therefore, matrix Γ is positive definite.

Partition Γ as

$$\Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{12}^* & \Gamma_{22} \end{bmatrix},$$

and note that, since Γ is positive definite, so is Γ_{22} . Therefore, we can write

$$\begin{aligned} \begin{bmatrix} \mathbf{s}_\tau^* & \mathbf{s}_d^* \end{bmatrix} \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{12}^* & \Gamma_{22} \end{bmatrix} \begin{bmatrix} \mathbf{s}_\tau \\ \mathbf{s}_d \end{bmatrix} &= \mathbf{s}_\tau^* \Gamma_{11} \mathbf{s}_\tau + \mathbf{s}_\tau^* \Gamma_{12} \mathbf{s}_d + \mathbf{s}_d^* \Gamma_{12}^* \mathbf{s}_\tau + \mathbf{s}_d^* \Gamma_{22} \mathbf{s}_d \\ &= (\mathbf{s}_d - \Gamma_{22}^{-1} \Gamma_{12}^* \mathbf{s}_\tau)^* \Gamma_{22} (\mathbf{s}_d - \Gamma_{22}^{-1} \Gamma_{12}^* \mathbf{s}_\tau) + \mathbf{s}_\tau^* \Gamma_{11} \mathbf{s}_\tau \end{aligned}$$

Since $\mathbf{s}_\tau^* \Gamma_{11} \mathbf{s}_\tau$ does not depend on \mathbf{s}_d , the optimization (13) becomes

$$\min_{\mathbf{s}_d} (\mathbf{s}_d - \Gamma_{22}^{-1} \Gamma_{12}^* \mathbf{s}_\tau)^* \Gamma_{22} (\mathbf{s}_d - \Gamma_{22}^{-1} \Gamma_{12}^* \mathbf{s}_\tau) = \min_{\mathbf{s}_d} \|\mathbf{s}_d - \Gamma_{22}^{-1} \Gamma_{12}^* \mathbf{s}_\tau\|_{\Gamma_{22}}^2,$$

which we can solve with the sphere decoding algorithm.

3 Multi-user case

We now focus on the general multi-user case. The relevant optimization problem is the one (3).

For any S , optimal channel estimate H is obtained as

$$\hat{H} = XS(S^*S)^{-1}S^*,$$

and thus we have

$$\begin{aligned} \arg \min_{H,S} \|X - HS^*\|^2 &= \arg \min_S \|X - XS(S^*S)^{-1}S^*\|^2 \\ &= \arg \min_S \text{tr} \{X(I - S(S^*S)^{-1}S^*)X^*\} \\ &= \arg \max_S \text{tr} \{(S^*S)^{-1}S^*X^*XS\} \end{aligned}$$

Recall that S is comprised of unitary symbols. If the users are uncorrelated, then $S^*S \approx TI$, and it follows that

$$\arg \max_S \text{tr} \{(S^*S)^{-1}S^*X^*XS\} = \arg \max_{\mathbf{s}^1, \dots, \mathbf{s}^M} \sum_{k=1}^M (\mathbf{s}^k)^*(X^*X)\mathbf{s}^k,$$

where \mathbf{s}^k is the symbol vector of the k^{th} user (i.e., the k^{th} row in matrix S). Furthermore,

$$\max_{\mathbf{s}^k} \sum_{k=1}^M (\mathbf{s}^k)^*(X^*X)\mathbf{s}^k = \min_{\mathbf{s}^1, \dots, \mathbf{s}^M} \sum_{k=1}^M (\mathbf{s}^k)^*(\rho I - X^*X)\mathbf{s}^k,$$

and thus, since the users choose their signals independently, optimization (3) is equivalent to the optimization problem

$$\min_{\mathbf{s}^k} (\mathbf{s}^k)^*(\rho I - X^*X)\mathbf{s}^k, \quad k = 1, \dots, M. \quad (14)$$

From (14), it is clear that without knowing anything about the users, we could not resolve their signal vectors. Therefore, we insist on $T_\tau > 0$, i.e., we assume training matrix \mathcal{S}_τ with distinct

rows. This resolves user ambiguity. More interestingly, optimization (14) implies that in order to detect signals received from M users, we only need to find M points inside the T -dimensional sphere – therefore, the dimension of the problem does not increase with the number of users, as one may expect. As a result, computational complexity grows only *linearly* in the number of users.¹

To explain how the sphere decoding is employed in this case, recall the tree-search interpretation of the algorithm discussed in Section 2.1. We search for M vectors \mathbf{s}^k , $1 \leq k \leq M$, in T -dimensional sphere of radius r . Each such vector allows for partition of form

$$\mathbf{s}^k = \begin{bmatrix} \mathbf{s}_\tau^k \\ \mathbf{s}_d^k \end{bmatrix},$$

where \mathbf{s}_τ^k is known to the receiver. In fact, matrix S_τ is comprised of all such vectors \mathbf{s}_τ^k . For each \mathbf{s}_τ^k , the sphere decoding algorithm then performs search for \mathbf{s}^k minimizing (14). One can interpret the search for \mathbf{s}^k as a tree-search procedure where the root of the tree is fixed to \mathbf{s}_τ^k . This is illustrated in Figure 2.

Note that by insisting on $T_\tau > 0$, we sacrifice some rate. However, this is typically required in order to resolve user ambiguity. Once this ambiguity is gone, sphere decoding can be employed with no additional requirements on the structure of transmitted data.

4 Complexity of the algorithm and choice of the search radius

Choice of the search radius is crucial for the computational complexity of the sphere decoding algorithm. If the radius is too small, we may not find anything and thus may have to repeat the search with larger radius. On the other hand, if the radius is too big, the algorithm has to go through many points which requires significant computation time.

¹Actually, increase in the computational complexity is even less than linear in the number of users. The reason for that is in the fact that preprocessing – i.e., finding ρ and performing the QR-factorization of $(\rho I - X^* X)$ – needs to be done only once for all users.

A simple choice of the radius could be based on the heuristic solutions to (3). For instance, an obvious heuristic for solving (3) consists of finding the eigenvector corresponding to the maximum eigenvalue of X^*X (or, equivalently, the dominant right singular vector of X) and then projecting it onto symbol space (i.e., rounding each entry). This heuristic can be exploited as a starting point of the sphere decoding search – the norm of the heuristic solution can be used as the search radius. However, we cannot say much about the complexity of the sphere decoding algorithm for this deterministic choice of the search radius.

Alternatively, when the distribution of the objective function of the minimization is known, one can choose radius according to some parameters of that distribution. For instance, in [4], objective function of the minimization problem that is being solved by sphere decoding has chi-square distribution, and the search radius is chosen according to the variance of that distribution. In particular, the variance is scaled in such a way that the probability of finding a point inside the sphere is very high. Furthermore, the expected complexity of the sphere decoding algorithm for such a choice of the radius is found and shown to be polynomial over a wide range of SNRs.

In this section, we show next that the objective function in (10) has a chi-square distribution at high SNRs. Parameters of the distribution suggest a probabilistic choice of the search radius, while the expected complexity results of [4, 5] imply practical feasibility of the sphere decoding algorithm employed for solving (10). We start by proving the following lemma.

Lemma 1. *At high SNR, $\mathbf{s}^*(\rho I - X^*X)\mathbf{s}$ is chi-square distributed.*

Proof: Assume large SNR, i.e., $SNR \gg 1$. Consider the eigenvalue decomposition of X^*X ,

$$X^*X = \begin{bmatrix} \hat{\mathbf{u}} & \hat{G} \end{bmatrix} \begin{bmatrix} \hat{\lambda} & 0 \\ 0 & \hat{\Lambda} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}^* \\ \hat{G}^* \end{bmatrix},$$

where $\hat{\lambda}$ denotes the largest eigenvalue, and where diagonal elements of $\hat{\Lambda}$ are the remaining $T - 1$ eigenvalues in decreasing order. Taking $\rho = \hat{\lambda}$, we can write the objective function of the

minimization (10) as

$$\begin{aligned}
\mathbf{s}^*(\rho I - X^*X)\mathbf{s} &= \mathbf{s}^*(\hat{\lambda}I - \hat{\mathbf{u}}\hat{\mathbf{u}}^*\hat{\lambda} - \hat{G}\hat{\Lambda}\hat{G}^*)\mathbf{s} \\
&= \mathbf{s}^*[\hat{\lambda}(I - \hat{\mathbf{u}}\hat{\mathbf{u}}^*) - \hat{G}\hat{\Lambda}\hat{G}^*]\mathbf{s} \\
&= \mathbf{s}^*\hat{G}(\hat{\lambda}I - \hat{\Lambda})\hat{G}^*\mathbf{s}
\end{aligned} \tag{15}$$

where in going from the second to the third line above, we used $I - \hat{\mathbf{u}}\hat{\mathbf{u}}^* = \hat{G}\hat{G}^*$. Furthermore,

$$\begin{aligned}
X^*X &= (X - \mathbf{h}\mathbf{s}^*)(X - \mathbf{h}\mathbf{s}^*) \\
&= \|\mathbf{h}\|^2\mathbf{s}\mathbf{s}^* + \mathbf{s}\mathbf{h}^*W + W^*\mathbf{h}\mathbf{s}^* + W^*W
\end{aligned} \tag{16}$$

$$= \hat{\lambda}\hat{\mathbf{u}}\hat{\mathbf{u}}^* + \hat{G}\hat{\Lambda}\hat{G}^* \tag{17}$$

On the other hand, for $\sigma^2 \ll 1$ (i.e., $SNR \gg 1$),

$$X^*X = \|\mathbf{h}\|^2 T \begin{pmatrix} \mathbf{s} \\ \sqrt{T} \end{pmatrix} \begin{pmatrix} \mathbf{s}^* \\ \sqrt{T} \end{pmatrix},$$

and hence, at high SNRs, the theoretical values of $\hat{\lambda}$ and $\hat{\Lambda}$ become

$$\lambda = T\|\mathbf{h}\|^2, \quad \Lambda = 0. \tag{18}$$

Note that, using (16), we obtain

$$\begin{aligned}
\hat{G}^*(X^*X)\mathbf{s} &= \hat{G}^*(\|\mathbf{h}\|^2\mathbf{s}\mathbf{s}^* + \mathbf{s}\mathbf{h}^*W + W^*\mathbf{h}\mathbf{s}^* + W^*W)\mathbf{s} \\
&= \lambda(\hat{G}^*\mathbf{s}) + (\hat{G}^*\mathbf{s})\mathbf{h}^*W\mathbf{s} + \hat{G}^*W^*\mathbf{h}T + \hat{G}^*W^*W\mathbf{s}
\end{aligned} \tag{19}$$

On the other hand, from (17),

$$\hat{G}^*(X^*X)\mathbf{s} = \lambda \underbrace{\hat{G}^*\hat{\mathbf{u}}}_{=0} \hat{\mathbf{u}}^* + \underbrace{\hat{G}^*\hat{G}}_{=I} \hat{\Lambda} \hat{G}^*\mathbf{s} = \hat{\Lambda}(\hat{G}^*\mathbf{s}) \quad (20)$$

Combining (19) and (20) leads to

$$\lambda(\hat{G}^*\mathbf{s}) + (\hat{G}^*\mathbf{s})\mathbf{h}^*W\mathbf{s} + \hat{G}^*W^*\mathbf{h}T + \hat{G}^*W^*W\mathbf{s} = \hat{\Lambda}(\hat{G}^*\mathbf{s}),$$

from which, letting $\sigma^2 \rightarrow 0$ and neglecting higher order terms (those whose power goes as σ^4), we obtain

$$\lambda(\hat{G}^*\mathbf{s}) + \hat{G}^*W^*\mathbf{h}T \approx 0 \Rightarrow \lambda(\hat{G}^*\mathbf{s}) \approx -\hat{G}^*W^*\mathbf{h}T,$$

which, since $\sigma^2 \ll 1$, we can further write as $\lambda(\hat{G}^*\mathbf{s}) \approx -G^*W^*\mathbf{h}T$. Therefore, for high SNRs, $(\hat{G}^*\mathbf{s})$ is circular Gaussian with zero mean. To find its variance, note that

$$TG^*W^*\mathbf{h} = TG^* \begin{bmatrix} W_1^*\mathbf{h} \\ \vdots \\ W_N^*\mathbf{h} \end{bmatrix},$$

where W_k denotes the k^{th} column of W . Also, note that

$$E[W_k^*\mathbf{h}\mathbf{h}^*W_l] = E[\mathbf{h}^*W_lW_k^*\mathbf{h}] = \sigma^2\|\mathbf{h}\|^2\delta_{k,l},$$

and thus $\hat{G}^*\mathbf{s}$ covariance matrix is given by

$$\begin{aligned} \text{cov}(\hat{G}^*\mathbf{s}) &= \frac{T^2}{\lambda^2} \text{cov}(G^*W^*\mathbf{h}) \\ &= \frac{T^2}{\lambda^2} \|\mathbf{h}\|^2 \sigma^2 I = \frac{T}{\lambda} \sigma^2 I \\ &= T\sigma^2(\lambda I - \Lambda)^{-1}, \end{aligned}$$

where $\Lambda = 0$ was inserted for convenience. Therefore,

$$(\hat{G}^* \mathbf{s}) \sim \mathcal{CN} [0, T\sigma^2(\lambda I - \Lambda)^{-1}].$$

At high SNR, $\lambda \approx \hat{\lambda}$, $\Lambda \approx \hat{\Lambda}$, and therefore approximately

$$(\hat{G}^* \mathbf{s}) \sim \mathcal{CN} [0, T\hat{\sigma}^2(\hat{\lambda}I - \hat{\Lambda})^{-1}],$$

where $\hat{\sigma}^2$ is an estimate of σ^2 . This estimate can be obtained (see, e.g., [8] and the references therein) as the mean of the $(N - 1)$ smallest eigenvalues of XX^* (or, alternatively, the smallest $(N - 1)$ non-zero eigenvalues of X^*X).

Therefore, we have just shown that the scaled term in (15), $\frac{1}{T\hat{\sigma}^2}(\mathbf{s}^* \hat{G})(\hat{\lambda}I - \hat{\Lambda})(\hat{G}^* \mathbf{s})$, is chi-square distributed with $(T - 1)$ degrees of freedom, and so is $\frac{1}{T\hat{\sigma}^2} \mathbf{s}^*(\rho I - X^*X)\mathbf{s}$.

□

We use the result of Lemma 1 to choose the size of the search radius for sphere decoding:

1. *Single-user case, blind scheme:*

Recall that the probability density function of chi-square distribution is given by an incomplete gamma function. Therefore, we can choose the search radius $r^2 = \alpha T \hat{\sigma}^2$ so that the cumulative density function

$$\int_0^{\alpha T} \gamma(\lambda, T - 1) d\lambda = 1 - \epsilon, \quad (21)$$

where $\gamma(\cdot, \cdot)$ denotes an incomplete gamma function and $1 - \epsilon$ is set close to 1, say, 0.99. This means that the optimal point is inside the sphere with probability $1 - \epsilon$. If the algorithm does not find any point inside the sphere, new radius needs to be calculated (say, by setting $1 - \epsilon = 0.999$).

2. *Single-user case, training-based scheme:*

Denoting the above computed radius for the blind scheme by r_b , the radius for the training-based scheme, r_t , may be chosen as

$$r_t^2 = r_b^2 - s_\tau^*(\Gamma_{11} - \Gamma_{12}\Gamma_{22}^{-1}\Gamma_{12}^*)s_\tau.$$

□

We should further remark that $|R| = 0$ for $\rho = \hat{\lambda}$. Therefore, to prevent any possible numerical problems, we suggest choice $\rho > \hat{\lambda}$, as stated in Section 2. Then the search radius should be modified and it may be chosen as

$$r' = (\rho - \hat{\lambda}) + r,$$

where r is chosen as implied by (21).

5 Soft decoding of channel codes and other remarks

So far, we have assumed that the users send uncoded data. On the other hand, if the users employ channel codes, we can use sphere decoding techniques for soft iterative decoding. To this end, one can use modified sphere decoding (the FP-MAP algorithm) proposed in [13], or the list sphere decoding proposed in [6]. Both modifications exploit the fact that sphere decoding finds *all* points that satisfy sphere constraint, e.g., in the single-user case, finds all vectors \mathbf{s} such that

$$\|X - \mathbf{h}\mathbf{s}\|^2 \leq r^2.$$

(For the FP-MAP algorithm, the condition is $\|X - \mathbf{h}\mathbf{s}\|^2 \leq r^2 + \sum \log p(s^i)$, where $p(s^i)$ is the *a priori* information about the i^{th} component of \mathbf{s} .) By an appropriate choice of the radius, one can guarantee that the algorithm will return a number of points, which are then used to for a *soft*

decision about the components of the vector \mathbf{s} . Assuming that transmitter, prior to modulation, employs channel codes (such as convolutional, turbo, LDPC, etc.), the soft decisions made by the modified sphere decoder are then iterated with the soft decisions made by the channel decoder. For brevity, we refrain ourselves from a more detailed discussion and refer the reader to [6], [13].

On another note, in this paper we have been primarily concerned with posing the problem of the joint ML channel estimation and signal detection in a form to which sphere decoding can be applied, and establishing relation with the existing analytical treatment of the complexity of sphere decoding. There are many practical realizations of sphere decoding that perform even better. For instance, the Schnorr-Euchner version of sphere decoding presented in [10], and further analyzed in [11], the radius scheduling suboptimal scheme introduced in [12], etc. Applying those techniques to further lower the complexity of sphere decoding in the context of joint ML estimation and detection are highly desirable.

6 Simulation results

In this section, we give several examples that illustrate the performance and the corresponding complexity of the sphere decoding algorithm in the previously described scenarios.

Example 1 [Single-user system]: Consider a single user which employs one antenna, while the receiver employs $N = 4$ antennas. The data is transmitted in blocks of length $T = 20$. We compare the BER performance of the sphere decoding algorithm and the iterative least squares with projections (ILSP) algorithm (see, e.g., [2], [9]). [Simulation results are obtained by performing Monte Carlo runs in which \mathbf{h} and W are varied.] The ILSP essentially finds the least squares estimate of the symbols and projects it onto the symbol space to obtain the estimate $\hat{\mathbf{s}}$. Then $\hat{\mathbf{s}}$ is used to update the estimate of the channel and the two aforementioned steps repeated until convergence.

As shown in Figure 2, the sphere decoding algorithm significantly outperforms ILSP over the

considered range of SNR. Figure 3 shows the complexity exponent, defined as

$$e = \log_T F,$$

where F denotes the total number of operations required to detect a vector \mathbf{s} . (For the sphere decoding, the total complexity is a sum of the operations for the QR factorization and the sphere decoding search). It is evident that the sphere decoding has roughly cubic complexity over the range of SNRs of interest.

Example 2 [Multi-user system]:

Example 3 [Single-user system, coded data]:

7 Summary and Conclusion

We considered the joint ML channel estimation and signal detection problem for single-input multiple-output wireless channels. To reduce the computational effort, we formulated the design problem so that it can be solved via the use of sphere decoding. It was shown that the algorithm, when applied to the problem herein, has polynomial expected complexity.

We treated both single-user and multi-user scenarios. For the latter, we have shown that the complexity of sphere decoding for joint ML estimation and detection in multi-user systems grows only linearly with the number of users. Simulations illustrated performance of the algorithm and its complexity. When compared with the best available heuristics, the sphere decoding provided much better performance at comparable complexity over a wide range of SNR.

There are many directions for the future work and possible extensions of the current work. Applying more powerful variations of sphere decoding (such as Schnorr-Euchner and radius scheduling techniques) to further lower the complexity of sphere decoding in the context of joint ML estimation and detection is of high interest. Furthermore, extension to the case where both trans-

mitter and receiver employ multi-antenna is highly desirable.

References

- [1] A. Paulraj and C. B. Papadias, "Space-time processing for wireless communications," *IEEE Signal Magazine*, pp. 49-83, November 1997.
- [2] P. Bohlin, "Iterative least square techniques with applications to adaptive antennas and CDMA systems," *Tech. report no. 409L*, Chalmers U. of Techn., 2001.
- [3] 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.
- [4] B. Hassibi and H. Vikalo, "On Sphere Decoding Algorithm. I. Expected Complexity," submitted to *IEEE Transactions on Signal Processing*.
- [5] H. Vikalo and B. Hassibi, "On Sphere Decoding Algorithm. II. Generalizations, Second-order Statistics, and Applications to Communications," submitted to *IEEE Transactions on Signal Processing*.
- [6] B. Hochwald and S. Ten Brink, "Achieving near-capacity on a multiple-antenna channel," submitted to *IEEE Transactions on Communications*, 2001.
- [7] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. Journal*, vol. 1, no. 2, pp. 41-59, 1996.
- [8] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Trans. on Signal Processing*, vol. 37, no. 5, pp. 720-741, May 1989.

- [9] S. Talwar, M. Viberg, and A. Paulraj, "Blind separation of synchronous co-channel digital signals using an antenna array - part I: algorithms." *IEEE Transactions on Signal Processing*, vol. 44, no. 5, pp. 1184-1897, May 1996.
- [10] C. P. Schnorr and M. Euchner, "Lattice basis reduction: improved practical algorithms and solving subset sum problems," *Mathematical Programming*, vol. 66, pp. 181-191, 1994.
- [11] E. Agrell and T. Eriksson and A. Vardy and K. Zeger, "Closest point search in lattices," *IEEE Transactions on Information Theory*, vol. 48, no. 8, pp. 2001–2214.
- [12] R. Gowaikar and B. Hassibi, "Efficient maximum-likelihood decoding via statistical pruning," submitted to *IEEE Transactions on Information Theory*, 2003.
- [13] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative Decoding for MIMO Channels via Modified Sphere Decoding," to appear in *IEEE Transactions on Wireless Communications*, 2004.

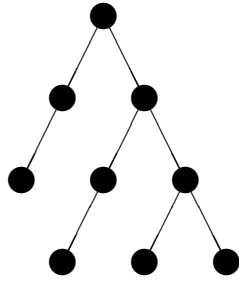


Figure 1: *Tree search of the sphere decoding algorithm.*

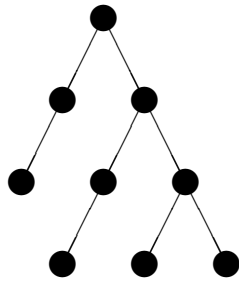


Figure 2: *Sphere decoding for multi-user case.*

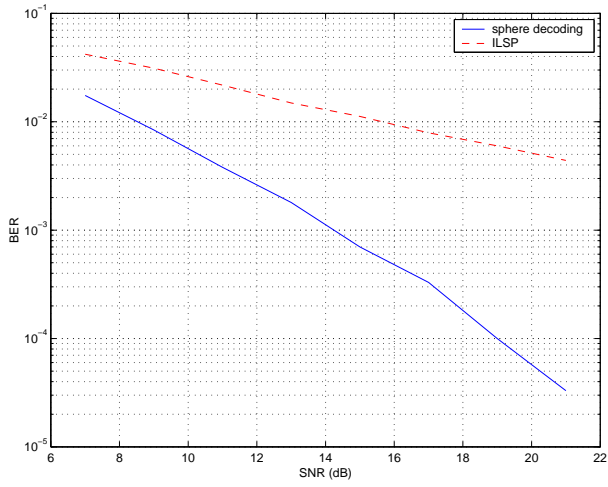


Figure 3: BER comparison of sphere decoding and ILSP algorithms, $n = 4, T = 20$.

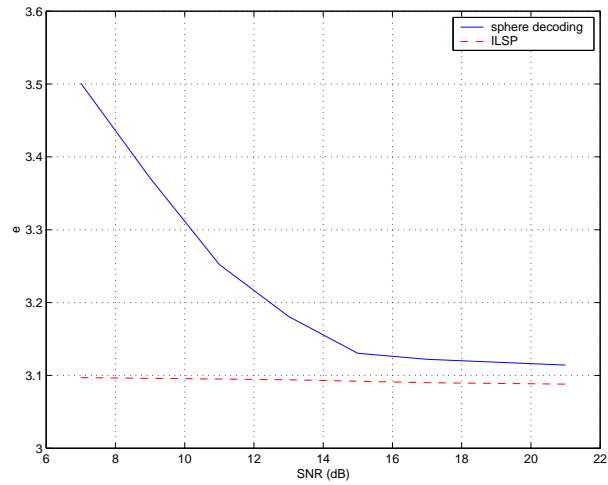


Figure 4: Expected complexity exponent, $n = 4, T = 20$.