

Matrix-Lifting Semi-Definite Programming for Decoding in Multiple Antenna Systems

Amin Mobasher and Amir K. Khandani

Abstract

This paper presents a computationally efficient decoder for multiple antenna systems. The proposed algorithm can be used for any constellation (QAM or PSK) and any labeling method. The decoder is based on matrix-lifting Semi-Definite Programming (SDP). The strength of the proposed method lies in a new relaxation algorithm applied to the method of [1]. This results in a reduction of the number of variables from $(NK + 1)^2$ to $(2N + K)^2$, where N is the number of antennas and K is the number of constellation points in each real dimension. Since the computational complexity of solving SDP is a polynomial function of the number of variables, we have a significant complexity reduction. Moreover, the proposed method offers a better performance as compared to the best quasi-maximum likelihood decoding methods reported in the literature.

I. INTRODUCTION

The problem of Maximum Likelihood (ML) decoding in Multi-Input Multi-Output (MIMO) wireless systems is known to be NP-hard. A variety of sub-optimum polynomial time algorithms based on Semi-Definite Programming (SDP) are suggested for MIMO decoding [1]–[9]. The first quasi ML decoding methods based on SDP were introduced for PSK signalling [2]–[5], offering a near ML performance and a polynomial time worst

The authors are with Coding & Signal Transmission Laboratory (www.cst.uwaterloo.ca), Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada, N2L 3G1, e-mail: {amin, khandani}@cst.uwaterloo.ca

This work is partly presented in the 10th Canadian Workshop on Information Theory (CWIT'07), Edmonton, AB, June 2007.

case complexity. Subsequently, SDP methods were used for decoding of MIMO systems based on QAM constellation [1], [6].

The method presented in [6] is for MIMO systems using 16-QAM, where the structure of constellation is captured by a polynomial constraint. Then, by introducing some slack variables, the constraints are expressed in terms of quadratic polynomials. This method can be generalized for larger constellations at the cost of defining more slack variables, increasing the complexity, and significantly decreasing the performance. The method proposed in [7] is a further relaxation of [6], only utilizing upper and lower bounds on the symbol energy in the relaxation step. There is a very slight degradation in performance compared to [6]; however, its computational complexity is independent of the constellation size for any uniform QAM (order of complexity is cubic). The method in [8] is a further tightening of [7] by appending some inequality conditions that are implicit in the alphabet constraint. Its computational complexity is still less than [6].

In [1], an efficient approximate ML decoder for MIMO systems is developed based on vector lifting SDP. The transmitted vector is expanded as a linear combination (with zero-one coefficients) of all the possible constellation points in each dimension. Using this formulation, the distance minimization in Euclidean space is expressed in terms of a binary quadratic minimization problem. The minimization of this problem is over the set of all binary rank-one matrices with column sums equal to one. Although the algorithm in [1] is a sub-optimal decoding method, it is shown that by adding several extra constraints, it can approach the ML performance. However, implementing the extra constraints increases the computational complexity.

In this paper, we introduce a new algorithm based on matrix-lifting SDP [10], [11] for any constellation (QAM or PSK) and any labeling method. This algorithm is inspired by the method in [1] with an efficient implementation resulting in a better performance and lower computational complexity. In SDP optimization problems, the computational complexity is a polynomial function of the number of variables. Using the proposed method, the number of variables in [1] is decreased from $(NK + 1)^2$ to $(2N + K)^2$, where N is the number of antennas and K is the number of constellation points in each real dimension. In addition to this large reduction in the complexity, simulation results

show that the proposed algorithm also outperforms all other known convex quasi-ML decoding methods, e.g. [6]–[8].

Following notations are used in the sequel. The space of $N \times K$ (resp. $N \times N$) real matrices is denoted by $\mathcal{M}_{N \times K}$ (resp. \mathcal{M}_N), and the space of $N \times N$ symmetric matrices is denoted by \mathcal{S}_N . For a $N \times K$ matrix $\mathbf{X} \in \mathcal{M}_{N \times K}$, the (i, j) th element is represented by x_{ij} , where $1 \leq i \leq N$, $1 \leq j \leq K$, i.e. $\mathbf{X} = [x_{ij}]$. We use $\text{trace}(\mathbf{A})$ to denote the trace of a square matrix \mathbf{A} . The space of symmetric matrices is considered with the trace inner product $\langle \mathbf{A}, \mathbf{B} \rangle = \text{trace}(\mathbf{A}\mathbf{B})$. For $\mathbf{A}, \mathbf{B} \in \mathcal{S}_N$, $\mathbf{A} \succeq 0$ (resp. $\mathbf{A} \succ 0$) denotes positive semi-definiteness (resp. positive definiteness), and $\mathbf{A} \succeq \mathbf{B}$ denotes $\mathbf{A} - \mathbf{B} \succeq 0$. For two matrices $\mathbf{A}, \mathbf{B} \in \mathcal{M}_N$, $\mathbf{A} \geq \mathbf{B}$, ($\mathbf{A} > \mathbf{B}$) means $a_{ij} \geq b_{ij}$, ($a_{ij} > b_{ij}$) for all i, j . The Kronecker product of two matrices \mathbf{A} and \mathbf{B} is denoted by $\mathbf{A} \otimes \mathbf{B}$. For $\mathbf{X} \in \mathcal{M}_{N \times K}$, $\text{vec}(\mathbf{X})$ denotes the vector in \mathbb{R}^{NK} (real NK -dimensional space) that is formed from the columns of the matrix \mathbf{X} . For $\mathbf{X} \in \mathcal{M}_N$, $\text{diag}(\mathbf{X})$ is a vector of the diagonal elements of \mathbf{X} . We use $\mathbf{e}_N \in \mathbb{R}^N$ (resp. $\mathbf{0}_N \in \mathbb{R}^N$) to denote the $N \times 1$ vector of all ones (resp. all zeros), $\mathbf{E}_{N \times K} \in \mathcal{M}_{N \times K}$ to denote the matrix of all ones, and \mathbf{I}_N to denote the $N \times N$ Identity matrix. For $\mathbf{X} \in \mathcal{M}_{N \times K}$, the notation $\mathbf{X}(1 : i, 1 : j)$, $i < K$ and $j < N$ denotes the sub-matrix of \mathbf{X} containing the first i rows and the first j columns.

The rest of the paper is organized as follows. The problem formulation is introduced in Section II. Section III is the review of the vector-lifting semi-definite programming presented in [1]. In Section IV, we propose our new algorithm based on matrix-lifting semi-definite programming. We use the geometry of the relaxation to find a projected relaxation which has a better performance. In Section V, we present an optimization method, based on matrix nearness to find the integer solution of the original decoding problem from the relaxed optimization problem. Finally, Section VI conclude the paper with some simulation results.

II. PROBLEM FORMULATION

A MIMO system with \tilde{M} transmit antennas and \tilde{N} receive antennas can be modeled by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $M = 2\tilde{M}$, $N = 2\tilde{N}$, \mathbf{y} is the $M \times 1$ received vector, \mathbf{H} is $M \times N$ real channel matrix, \mathbf{n} is $N \times 1$ additive white Gaussian noise vector, and \mathbf{x} is $N \times 1$ data vector whose components are selected from the set $\{s_1, \dots, s_K\}$, see [1]. Noting $x_i \in \{s_1, \dots, s_K\}$, for $i = 1, \dots, N$, we have

$$x_i = u_{i,1}s_1 + u_{i,2}s_2 + \dots + u_{i,K}s_K, \quad (2)$$

where

$$u_{i,j} \in \{0, 1\} \quad \text{and} \quad \sum_{j=1}^K u_{i,j} = 1, \quad \forall i = 1, \dots, N. \quad (3)$$

Let

$$\mathbf{U} = \begin{bmatrix} u_{1,1} & \dots & u_{1,K} \\ u_{2,1} & \dots & u_{2,K} \\ \vdots & \ddots & \vdots \\ u_{N,1} & \dots & u_{N,K} \end{bmatrix} \quad \text{and} \quad \mathbf{s} = \begin{bmatrix} s_1 \\ \vdots \\ s_K \end{bmatrix}.$$

Therefore, the transmitted vector is $\mathbf{x} = \mathbf{U}\mathbf{s}$ where $\mathbf{U}\mathbf{e}_K = \mathbf{e}_N$.

At the receiver, the ML decoding is given by

$$\hat{\mathbf{x}} = \arg \min_{x_i \in \{s_1, \dots, s_K\}} \|\hat{\mathbf{y}} - \mathbf{H}\mathbf{x}\|^2, \quad (4)$$

where $\hat{\mathbf{x}}$ is the most likely input vector and $\hat{\mathbf{y}}$ is the received vector. Noting $\mathbf{x} = \mathbf{U}\mathbf{s}$, this problem is equivalent to

$$\begin{aligned} & \min_{\mathbf{U}\mathbf{e}_K = \mathbf{e}_N} \|\hat{\mathbf{y}} - \mathbf{H}\mathbf{U}\mathbf{s}\|^2 \equiv \\ & \min_{\mathbf{U}\mathbf{e}_K = \mathbf{e}_N} \mathbf{s}^T \mathbf{U}^T \mathbf{H}^T \mathbf{H} \mathbf{U} \mathbf{s} - 2\hat{\mathbf{y}}^T \mathbf{H} \mathbf{U} \mathbf{s}. \end{aligned} \quad (5)$$

Therefore, the decoding problem can be formulated as

$$\begin{aligned} & \min \quad \mathbf{s}^T \mathbf{U}^T \mathbf{H}^T \mathbf{H} \mathbf{U} \mathbf{s} - 2\hat{\mathbf{y}}^T \mathbf{H} \mathbf{U} \mathbf{s} \\ & s.t. \quad \mathbf{U}\mathbf{e}_K = \mathbf{e}_N \\ & \quad \quad u_{i,j} \in \{0, 1\}. \end{aligned} \quad (6)$$

Let $\mathbf{Q} = \mathbf{H}^T \mathbf{H}$, $\mathbf{S} = \mathbf{s}\mathbf{s}^T$, $\mathbf{C} = -\mathbf{s}\hat{\mathbf{y}}^T \mathbf{H}$, and let $\mathcal{E}_{N \times K}$ denote the set of all binary matrices in $\mathcal{M}_{N \times K}$ with row sums equal to one, i.e.

$$\mathcal{E}_{N \times K} = \{\mathbf{U} \in \mathcal{M}_{N \times K} : \mathbf{U}\mathbf{e}_K = \mathbf{e}_N, u_{ij} \in \{0, 1\}\}. \quad (7)$$

Therefore, the minimization problem (6) is

$$\begin{aligned} \min \quad & \text{trace}(\mathbf{S}\mathbf{U}^T\mathbf{Q}\mathbf{U} + 2\mathbf{C}\mathbf{U}) \\ \text{s.t.} \quad & \mathbf{U} \in \mathcal{E}_{N \times K} \end{aligned} \quad (8)$$

III. VECTOR-LIFTING SEMI-DEFINITE PROGRAMMING

In order to solve the optimization problem (8), the authors in [1] proposed a *quadratic vector optimization* solution by defining $\mathbf{u} = \text{vec}(\mathbf{U}^T)$, $\mathbf{U} \in \mathcal{E}_{N \times K}$. By using this notation, the objective function is replaced by $\mathbf{u}^T(\mathbf{Q} \otimes \mathbf{S})\mathbf{u} + 2\text{vec}(\mathbf{C})^T\mathbf{u}$. Then, the quadratic form is linearized using the vector $\begin{bmatrix} 1 \\ \mathbf{u} \end{bmatrix}$, i.e.

$$\begin{aligned} \mathbf{Z}_{\mathbf{u}} &= \begin{bmatrix} 1 \\ \mathbf{u} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{u}^T \end{bmatrix} \\ &= \begin{bmatrix} 1 & \mathbf{u}^T \\ \mathbf{u} & \mathbf{u}\mathbf{u}^T \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{u}^T \\ \mathbf{u} & \mathbf{X} \end{bmatrix}, \end{aligned} \quad (9)$$

where $\mathbf{X} = \mathbf{u}\mathbf{u}^T$ and it is relaxed to $\mathbf{X} \succeq \mathbf{u}\mathbf{u}^T$, or equivalently, by the Schur complement, to the lifted constraint $\begin{bmatrix} 1 & \mathbf{u}^T \\ \mathbf{u} & \mathbf{X} \end{bmatrix} \succeq 0$. Note that this matrix is selected from the set

$$\mathcal{F} := \text{conv} \{ \mathbf{Z}_{\mathbf{u}} : \mathbf{u} = \text{vec}(\mathbf{U}^T), \mathbf{U} \in \mathcal{E}_{N \times K} \}, \quad (10)$$

where $\text{conv}(\cdot)$ denotes the convex hull of a set. Therefore, the decoding problem using *vector lifting semi-definite programming* can be represented by

$$\begin{aligned} \text{trace} \quad & \begin{bmatrix} 0 & \text{vec}(\mathbf{C})^T \\ \text{vec}(\mathbf{C}) & \mathbf{Q} \otimes \mathbf{S} \end{bmatrix} \begin{bmatrix} 1 & \mathbf{u}^T \\ \mathbf{u} & \mathbf{X} \end{bmatrix} \\ \text{s.t.} \quad & \begin{bmatrix} 1 & \mathbf{u}^T \\ \mathbf{u} & \mathbf{X} \end{bmatrix} \in \mathcal{F}, \end{aligned} \quad (11)$$

which can be solved by SDP technique.

Note that in (11), the optimization parameter is a matrix in \mathcal{S}_{NK+1} , which has $(NK+1)^2$ variables. In the following, we reduce the number of optimization variables by exploiting the matrix structure of \mathbf{U} .

IV. MATRIX-LIFTING SEMI-DEFINITE PROGRAMMING

To keep the matrix \mathbf{U} in its original form in (8), the idea is to use the constraint $\mathbf{X} = \mathbf{U}^T \mathbf{U}$ instead of $\mathbf{X} = \mathbf{u}\mathbf{u}^T$. As a result, the relaxation $\mathbf{X} \succeq \mathbf{u}\mathbf{u}^T$ is transformed to $\mathbf{X} \succeq \mathbf{U}^T \mathbf{U}$, or equivalently, by the Schur complement, $\begin{bmatrix} \mathbf{I}_N & \mathbf{U} \\ \mathbf{U}^T & \mathbf{X} \end{bmatrix} \succeq 0$. This is known as matrix-lifting semi-definite programming. Define the new variable $\mathbf{V} = \mathbf{U}\mathbf{S}$. Since the matrix \mathbf{S} is symmetric, the objective function in (8) can be represented as the Quadratic Matrix Program [11]

$$\begin{aligned} & \text{trace} \left(\begin{bmatrix} \mathbf{U}^T & \mathbf{V}^T \end{bmatrix} \begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{Q} \\ \frac{1}{2}\mathbf{Q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix} + 2\mathbf{C}\mathbf{U} \right) \\ = & \text{trace} \left(\begin{bmatrix} \mathbf{0} & \frac{1}{2}\mathbf{Q} \\ \frac{1}{2}\mathbf{Q} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{U}^T & \mathbf{V}^T \end{bmatrix} + 2\mathbf{C}\mathbf{U} \right) \\ = & \text{trace} (\mathcal{L}_Q \mathbf{W}_U), \end{aligned} \quad (12)$$

where

$$\mathcal{L} = \begin{bmatrix} \mathbf{0} & \mathbf{C} & \mathbf{0} \\ \mathbf{C}^T & \mathbf{0} & \frac{1}{2}\mathbf{Q} \\ \mathbf{0} & \frac{1}{2}\mathbf{Q} & \mathbf{0} \end{bmatrix} \quad (13)$$

and

$$\mathbf{W}_U = \begin{bmatrix} \mathbf{I} & \mathbf{U}^T & \mathbf{V}^T \\ \mathbf{U} & \mathbf{U}\mathbf{U}^T & \mathbf{U}\mathbf{V}^T \\ \mathbf{V} & \mathbf{V}\mathbf{U}^T & \mathbf{V}\mathbf{V}^T \end{bmatrix}. \quad (14)$$

To linearize \mathbf{W}_U , we consider the matrix

$$\begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{U}^T & \mathbf{V}^T \end{bmatrix} = \begin{bmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{Y} & \mathbf{Z} \end{bmatrix}, \quad (15)$$

where $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathcal{S}_N$. This equality can be relaxed to

$$\begin{bmatrix} \mathbf{U}\mathbf{U}^T & \mathbf{U}\mathbf{V}^T \\ \mathbf{V}\mathbf{U}^T & \mathbf{V}\mathbf{V}^T \end{bmatrix} - \begin{bmatrix} \mathbf{X} & \mathbf{Y} \\ \mathbf{Y} & \mathbf{Z} \end{bmatrix} \preceq 0. \quad (16)$$

It can be shown that this relaxation is convex in the Löwner partial order and it is equivalent to the linear constraint [10]

$$\mathbf{W} \triangleq \begin{bmatrix} \mathbf{I} & \mathbf{U}^T & \mathbf{V}^T \\ \mathbf{U} & \mathbf{X} & \mathbf{Y} \\ \mathbf{V} & \mathbf{Y} & \mathbf{Z} \end{bmatrix} \succeq 0. \quad (17)$$

On the other hand, the feasible set in (8) is the set of binary matrices in $\mathcal{M}_{N \times K}$ with row sum equal to one, the set $\mathcal{E}_{N \times K}$ in (7). By relaxing the rank-one constraint for the matrix variable in (12), we have a tractable SDP problem. The feasible set for the objective function in (12) is approximated by

$$\begin{aligned} \mathcal{F}_{\mathcal{M}} = \text{conv} \{ & \mathbf{W}_{\mathbf{U}} \mid \mathbf{U} \in \mathcal{M}_{N \times K} : \mathbf{U}\mathbf{e}_K = \mathbf{e}_N, \\ & u_{ij} \in \{0, 1\}, \forall i, j; \mathbf{V} = \mathbf{U}\mathbf{S} \} \end{aligned} \quad (18)$$

Therefore, the decoding problem can be represented by

$$\begin{aligned} \min \quad & \text{trace}(\mathcal{L}\mathbf{W}) \\ \text{s.t.} \quad & \mathbf{W} \in \mathcal{F}_{\mathcal{M}}. \end{aligned} \quad (19)$$

Note that the size of matrix \mathbf{W} is $(2N+K) \times (2N+K)$, compared to $(NK+1) \times (NK+1)$ in [1]. In SDP optimization problems, the computational complexity is a polynomial function of the number of variables (elements of \mathbf{W}). By the new implementation of (19), the number of variables in [1] is decreased from $(NK+1)^2$ to $(2N+K)^2$, resulting in a large reduction in the complexity.

Although the rank constraint in (15) is relaxed, we can still consider some additional linear constraints to further improve the quality of the solution. These constraints are valid for the non-convex rank-constrained decoding problem. However, we force the SDP problem to satisfy these constraints. Consider the auxiliary matrix \mathbf{V} and the symmetric matrices \mathbf{X} , \mathbf{Y} and \mathbf{Z} in matrix \mathbf{W} . Since $\mathbf{U} \in \mathcal{E}_{N \times K}$ and $\sum_{j=1}^N u_{ij}^2 = 1$, it is clear that $\text{diag}(\mathbf{X}) = \mathbf{e}_N$. Also, \mathbf{Y} represents $\mathbf{U}\mathbf{S}\mathbf{U}^T$ and \mathbf{Z} represents $\mathbf{U}\mathbf{S}^2\mathbf{U}^T$. It is easy to show that

$$\text{diag}(\mathbf{Y}) = \mathbf{U}\text{diag}(\mathbf{S}) \quad \text{and} \quad \text{diag}(\mathbf{Z}) = \mathbf{U}\text{diag}(\mathbf{S}^2). \quad (20)$$

Moreover, $\mathbf{S} = \mathbf{s}\mathbf{s}^T$ (rank-one matrix) and $\mathbf{S}^2 = (\sum_{i=1}^K s_i^2)\mathbf{S}$. Therefore, instead of $\text{diag}(\mathbf{Z}) = \mathbf{U}\text{diag}(\mathbf{S}^2)$, we have a stronger result for \mathbf{Z} , i.e. $\mathbf{Z} = (\sum_{i=1}^K s_i^2)\mathbf{Y}$. Therefore, we have

$$\begin{aligned}
\min \quad & \text{trace} \left(\mathcal{L} \begin{bmatrix} \mathbf{I} & \mathbf{U}^T & \mathbf{V}^T \\ \mathbf{U} & \mathbf{X} & \mathbf{Y} \\ \mathbf{V} & \mathbf{Y} & \mathbf{Z} \end{bmatrix} \right) \\
s.t. \quad & \mathbf{U}\mathbf{e}_K = \mathbf{e}_N ; \mathbf{U} \geq 0 \\
& \mathbf{V} = \mathbf{U}\mathbf{S} \\
& \text{diag}(\mathbf{X}) = \mathbf{e}_N \\
& \text{diag}(\mathbf{Y}) = \mathbf{U}\text{diag}(\mathbf{S}) \\
& \mathbf{Z} = \left(\sum_{i=1}^K s_i^2 \right) \mathbf{Y} \\
& \begin{bmatrix} \mathbf{I} & \mathbf{U}^T & \mathbf{V}^T \\ \mathbf{U} & \mathbf{X} & \mathbf{Y} \\ \mathbf{V} & \mathbf{Y} & \mathbf{Z} \end{bmatrix} \succeq 0 \\
& \mathbf{U}, \mathbf{V} \in \mathcal{M}_{N \times K}, \mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathcal{S}_N
\end{aligned} \tag{21}$$

The equation in (20) determines the diagonal elements of \mathbf{Y} . This property is hidden in the special structure of \mathbf{U} , i.e. $\mathbf{U} \in \mathcal{E}_{N \times K}$. By using this property, we can even add more constraints. The equation $\mathbf{Y} = \mathbf{U}\mathbf{S}\mathbf{U}^T$ implies that $Y_{ij} = S_{kl}$ for some k and l . Therefore, the value of Y_{ij} is between the minimum and the maximum elements of \mathbf{S} . In addition, it can be easily shown that in communication applications, \mathbf{S} , \mathbf{Y} , and \mathbf{Z} are diagonal dominant matrices (since $\mathbf{s}^T \mathbf{e}_K = 0$). This property can be also used to add more constraints to improve the quality of the solution. Our studies show that the improvements due to including the above constraints are marginal. Therefore, in the sequel, we focus on the form given in (21) with the following consideration. The objective function in (8) is $\text{trace}(\mathbf{S}\mathbf{U}^T\mathbf{Q}\mathbf{U} + 2\mathbf{C}\mathbf{U})$ which is equivalent to $\text{trace}(\mathbf{Q}\mathbf{U}\mathbf{S}\mathbf{U}^T + 2\mathbf{U}\mathbf{C})$. Exchanging the role of \mathbf{Q} and \mathbf{S} results in two different formulations. Here, the auxiliary variable \mathbf{V} is defined as $\mathbf{Q}\mathbf{U}$. Similarly, the auxiliary variables \mathbf{X} , \mathbf{Y} , and \mathbf{Z} represents $\mathbf{U}^T\mathbf{U}$, $\mathbf{U}^T\mathbf{Q}\mathbf{U}$, and $\mathbf{U}^T\mathbf{Q}^2\mathbf{U}$, respectively. Therefore, it is easy to show that the equivalent minimization

problem is

$$\begin{aligned}
\min \quad & \text{trace} \left(\begin{bmatrix} \mathbf{0} & \mathbf{C} & \mathbf{0} \\ \mathbf{C}^T & \mathbf{0} & \frac{1}{2}\mathbf{S} \\ \mathbf{0} & \frac{1}{2}\mathbf{S} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{U} & \mathbf{V} \\ \mathbf{U}^T & \mathbf{X} & \mathbf{Y} \\ \mathbf{V}^T & \mathbf{Y} & \mathbf{Z} \end{bmatrix} \right) \\
s.t. \quad & \mathbf{U}\mathbf{e}_K = \mathbf{e}_N ; \mathbf{U} \geq 0 \\
& \mathbf{V} = \mathbf{Q}\mathbf{U} \\
& \text{diag}(\mathbf{X}) = \mathbf{U}^T \mathbf{e}_N ; X_{ij} = 0 \ i \neq j \\
& \mathbf{Y}\mathbf{e}_K = \mathbf{U}^T \mathbf{Q}\mathbf{e}_N ; \text{trace}(\mathbf{Y}\mathbf{E}_K) = \text{trace}(\mathbf{Q}\mathbf{E}_N) \\
& \mathbf{Z}\mathbf{e}_K = \mathbf{U}^T \mathbf{Q}^2 \mathbf{e}_N ; \text{trace}(\mathbf{Z}\mathbf{E}_K) = \text{trace}(\mathbf{Q}^2 \mathbf{E}_N) \\
& \begin{bmatrix} \mathbf{I} & \mathbf{U} & \mathbf{V} \\ \mathbf{U}^T & \mathbf{X} & \mathbf{Y} \\ \mathbf{V}^T & \mathbf{Y} & \mathbf{Z} \end{bmatrix} \succeq 0 \\
& \mathbf{U}, \mathbf{V} \in \mathcal{M}_{N \times K}, \mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathcal{S}^K, \tag{22}
\end{aligned}$$

where the size of the variable matrix is $(2K + N)$. Note that both (21) and (22) are equivalent, however, depending on the structure of the system (values of N and K), we can use the one which offers a smaller number of variables. In the following, we focus on (21), which is a better choice for $N \leq K$.

A. Geometry of the Relaxation

In this section, we eliminate the constraints defining $\mathbf{U}\mathbf{e}_K = \mathbf{e}_N$ by providing a tractable representation of the linear manifold spanned by this constraint. This method is called *gradient projection* or *reduced gradient method* [12]. The following lemma is on the representation of matrices having sum of the elements in each row equal to one. This lemma is used in our reduced gradient method.

Lemma 1: [1] Let $\mathbf{G} = \left[\mathbf{I}_{K-1} \mid -\mathbf{e}_{K-1} \right] \in \mathcal{M}_{(K-1) \times K}$ and $\mathbf{F} = \frac{1}{K} (\mathbf{E}_{N \times K} - \mathbf{E}_{N \times (K-1)} \mathbf{G}) \in \mathcal{M}_{N \times K}$. A matrix $\mathbf{U} \in \mathcal{M}_{N \times K}$ with the property that the summation of its elements in each row is equal to one, i.e. $\mathbf{U}\mathbf{e}_K = \mathbf{e}_N$, can be written as

$$\mathbf{U} = \mathbf{F} + \hat{\mathbf{U}}\mathbf{G}, \tag{23}$$

where $\hat{\mathbf{U}} = \mathbf{U}(\mathbf{1} : \mathbf{N}, \mathbf{1} : (\mathbf{K} - 1))$.

Corollary 1: $\forall \mathbf{U} \in \mathcal{E}_{N \times K}, \exists \hat{\mathbf{U}} \in \mathcal{M}_{N \times (K-1)}, \hat{u}_{ij} \in \{0, 1\}$ s.t. $\mathbf{U} = \mathbf{F} + \hat{\mathbf{U}}\mathbf{G}$, where $\hat{\mathbf{U}} = \mathbf{U}(\mathbf{1} : \mathbf{N}, \mathbf{1} : (\mathbf{K} - 1))$. Note that the summation of each row of $\hat{\mathbf{U}}$ is 0 or 1.

Consider the minimization problem (8). By substituting (23), the objective function is

$$\begin{aligned}
& \text{trace}(\mathbf{S}\mathbf{U}^T\mathbf{Q}\mathbf{U} + 2\mathbf{C}\mathbf{U}) \\
&= \text{trace}\left(\mathbf{S}(\mathbf{F} + \hat{\mathbf{U}}\mathbf{G})^T\mathbf{Q}(\mathbf{F} + \hat{\mathbf{U}}\mathbf{G}) + 2\mathbf{C}(\mathbf{F} + \hat{\mathbf{U}}\mathbf{G})\right) \\
&= \text{trace}\left(\mathbf{G}\mathbf{S}\mathbf{G}^T\hat{\mathbf{U}}^T\mathbf{Q}\hat{\mathbf{U}} + \mathbf{G}\mathbf{S}\mathbf{F}^T\mathbf{Q}\hat{\mathbf{U}} + \mathbf{Q}\mathbf{F}\mathbf{S}\mathbf{G}^T\hat{\mathbf{U}}^T\right. \\
&\quad \left.+ \mathbf{G}\mathbf{C}\hat{\mathbf{U}} + \mathbf{C}^T\mathbf{G}^T\hat{\mathbf{U}}^T + 2\mathbf{C}\mathbf{F} + \mathbf{S}\mathbf{F}^T\mathbf{Q}\mathbf{F}\right) \\
&= \text{trace}\left(\hat{\mathcal{L}}\mathbf{W}_{\hat{\mathbf{U}}} + 2\mathbf{C}\mathbf{F} + \mathbf{S}\mathbf{F}^T\mathbf{Q}\mathbf{F}\right), \tag{24}
\end{aligned}$$

where

$$\begin{aligned}
\hat{\mathcal{L}} &= \begin{bmatrix} \mathbf{0} & \mathbf{G}\mathbf{S}\mathbf{F}^T\mathbf{Q} + \mathbf{G}\mathbf{C} & \mathbf{0} \\ \mathbf{Q}\mathbf{F}\mathbf{S}\mathbf{G}^T + \mathbf{C}^T\mathbf{G}^T & \mathbf{0} & \frac{1}{2}\mathbf{Q} \\ \mathbf{0} & \frac{1}{2}\mathbf{Q} & \mathbf{0} \end{bmatrix}, \\
\mathbf{W}_{\hat{\mathbf{U}}} &= \begin{bmatrix} \mathbf{I} & \hat{\mathbf{U}}^T & \hat{\mathbf{V}}^T \\ \hat{\mathbf{U}} & \hat{\mathbf{U}}\hat{\mathbf{U}}^T & \hat{\mathbf{U}}\hat{\mathbf{V}}^T \\ \hat{\mathbf{V}} & \hat{\mathbf{V}}\hat{\mathbf{U}}^T & \hat{\mathbf{V}}\hat{\mathbf{V}}^T \end{bmatrix}, \\
\hat{\mathbf{V}} &= \hat{\mathbf{U}}\mathbf{G}\mathbf{S}\mathbf{G}^T. \tag{25}
\end{aligned}$$

Therefore, (6) can be written as

$$\begin{aligned}
\min \quad & \text{trace}\left(\hat{\mathcal{L}}\mathbf{W}_{\hat{\mathbf{U}}}\right) \\
s.t. \quad & \hat{\mathbf{U}} = \mathbf{U}(\mathbf{1} : \mathbf{N}, \mathbf{1} : (\mathbf{K} - 1)); \quad \mathbf{U} \in \mathcal{E}_{N \times K} \\
& \hat{\mathbf{V}} = \hat{\mathbf{U}}(\mathbf{G}\mathbf{S}\mathbf{G}^T) \tag{26}
\end{aligned}$$

Using a similar procedure, we can show that (26) is equivalent to the following reduced matrix-lifting semi-definite programming problem:

$$\begin{aligned}
\min \quad & \text{trace} \left(\hat{\mathcal{L}} \begin{bmatrix} \mathbf{I} & \hat{\mathbf{U}}^T & \hat{\mathbf{V}}^T \\ \hat{\mathbf{U}} & \hat{\mathbf{X}} & \hat{\mathbf{Y}} \\ \hat{\mathbf{V}} & \hat{\mathbf{Y}} & \hat{\mathbf{Z}} \end{bmatrix} \right) \\
s.t. \quad & \hat{\mathbf{U}} \mathbf{e}_{\mathbf{K}-1} \leq \mathbf{e}_{\mathbf{N}} ; \hat{\mathbf{U}} \succeq \mathbf{0} \\
& \hat{\mathbf{V}} = \hat{\mathbf{U}} (\mathbf{G} \mathbf{S} \mathbf{G}^T) \\
& \text{diag}(\hat{\mathbf{X}}) = \hat{\mathbf{U}} \mathbf{e}_{\mathbf{K}-1} \\
& \text{diag}(\hat{\mathbf{Y}}) = \hat{\mathbf{U}} \text{diag}(\mathbf{G} \mathbf{S} \mathbf{G}^T) \\
& \hat{\mathbf{Z}} = \left(\sum_{i=1}^{\mathbf{K}-1} (\mathbf{s}_i - \mathbf{s}_{\mathbf{K}})^2 \right) \hat{\mathbf{Y}} \\
& \begin{bmatrix} \mathbf{I} & \hat{\mathbf{U}}^T & \hat{\mathbf{V}}^T \\ \hat{\mathbf{U}} & \hat{\mathbf{X}} & \hat{\mathbf{Y}} \\ \hat{\mathbf{V}} & \hat{\mathbf{Y}} & \hat{\mathbf{Z}} \end{bmatrix} \succeq \mathbf{0} \\
& \hat{\mathbf{U}}, \hat{\mathbf{V}} \in \mathcal{M}_{\mathbf{N} \times (\mathbf{K}-1)}, \hat{\mathbf{X}}, \hat{\mathbf{Y}}, \hat{\mathbf{Z}} \in \mathcal{S}_{\mathbf{N}}
\end{aligned} \tag{27}$$

Note that this method can also be applied to the equivalent formulation in (22).

B. Solving the SDP Problem

The relaxed decoding problems can be solved using Interior-Point Methods (IPMs), which are the most common methods for solving SDP problems of moderate sizes with polynomial computational complexities [13]. There are a large number of IPM-based solvers to handle SDP problems, e.g., DSDP [14], SeDuMi [15], SDPA [16], etc. In our numerical experiments, we use SDPA solver.

In the matrix-lifting SDP optimization problem (21), the rank-constrained matrix $\mathbf{W}_{\mathbf{u}}$ is relaxed to the positive semi-definite matrix \mathbf{W} . Utilizing the rank-constrained property of the variable parameter, the relaxed problem (21) can be solved using a non-linear method, known as the *augmented Lagrangian algorithm*. This approach can be used for large problem sizes and the complexity can be significantly reduced, while the performance degradation is negligible [17].

V. INTEGER SOLUTION - MATRIX NEARNESS PROBLEM

Solving the relaxed decoding problems results in the solution $\tilde{\mathbf{U}}$. In general, this matrix is not in $\mathcal{E}_{N \times K}$. The condition $\mathbf{U}\mathbf{e}_K = \mathbf{e}_N$ is satisfied. However, the elements are between 0 and 1. This matrix has to be converted to a 0-1 matrix by finding a matrix in $\mathcal{E}_{N \times K}$ which is nearest to this matrix. Matrix approximation problems typically measure the distance between matrices with a norm. The Frobenius and spectral norms are common choices as they are analytically tractable.

To find the nearest solution in $\mathcal{E}_{N \times K}$ to $\tilde{\mathbf{U}}$, the solution of the relaxed problem, we solve

$$\min_{\mathbf{U} \in \mathcal{E}_{N \times K}} \|\mathbf{U} - \tilde{\mathbf{U}}\|_{\mathbb{F}}^2, \quad (28)$$

where $\|\mathbf{A}\|_{\mathbb{F}}^2$ is the Frobenius norm of the matrix \mathbf{A} which is defined as $\|\mathbf{A}\|_{\mathbb{F}}^2 = \text{trace}(\mathbf{A}\mathbf{A}^T)$, and

$$\begin{aligned} \|\mathbf{U} - \tilde{\mathbf{U}}\|_{\mathbb{F}}^2 &= \text{trace}\left((\mathbf{U} - \tilde{\mathbf{U}})(\mathbf{U} - \tilde{\mathbf{U}})^T\right) \\ &= N - 2\text{trace}(\tilde{\mathbf{U}}\mathbf{U}^T) + \text{trace}(\tilde{\mathbf{U}}\tilde{\mathbf{U}}^T). \end{aligned} \quad (29)$$

The last equality is due to the fact that for any $\mathbf{U} \in \mathcal{E}_{N \times K}$, we have $\text{diag}(\mathbf{U}\mathbf{U}^T) = \mathbf{e}_N$, see (21). Therefore, after removing the constants, finding the integer solution is the solution of the following problem:

$$\max_{\mathbf{U} \in \mathcal{E}_{N \times K}} \text{trace}(\tilde{\mathbf{U}}\mathbf{U}^T) \quad (30)$$

Consider the maximization problem

$$\begin{aligned} \max \quad & \text{trace}(\tilde{\mathbf{U}}\mathbf{U}^T) \\ \text{s.t.} \quad & \mathbf{U}\mathbf{e}_K = \mathbf{e}_N \\ & 0 \leq \mathbf{U} \leq 1, \end{aligned} \quad (31)$$

where \leq in the last constraint is element-wise. This problem is a linear programming problem with linear constraints and the optimum solution is a corner point meaning that the constraints are satisfied with equality at the optimum point. In other words, at the optimum point, $\mathbf{U} \in \mathcal{E}_{N \times K}$. Therefore, to find the solution for (30), we can simply solve

the linear problem (31), which is strongly polynomial time. To improve this result, the randomization algorithms, introduced in [1], can be further applied.

VI. SIMULATION RESULTS

We simulate the proposed matrix lifting method (27) for system with 4 transmit and 4 receive antennas employing 16-QAM. Fig. 1 shows the performance of the proposed method vs. the performance of the vector lifting method in [1] and the previous known methods in [6]–[8]. As it can be seen, the proposed method outperforms all other convex sub-optimal methods.

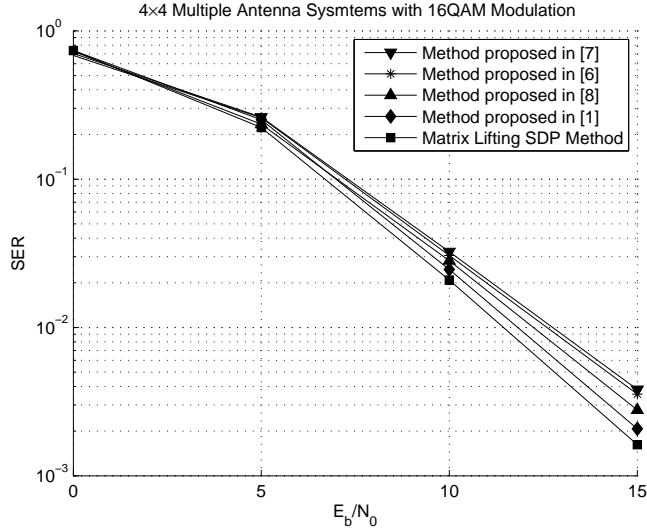


Fig. 1. Performance of the proposed matrix lifting SDP method in a MIMO system with 4 transmit and 4 receive antennas employing 16-QAM

The worst case complexity of the proposed method solved by IPMs is a polynomial function of the number of antennas (similar to the analysis in [1]). In the optimization problem of (21), where $N \leq K$, the dimension of the matrix variable \mathbf{W} is $m = O(K)$ and the number of constraints is $p = O(K^2)$. Similar to [1], it can be easily seen that a solution to (27) can be found in at most $O(K^{5.5})$ arithmetic operations (utilizing the sparsity of the rank-one constraint matrices), where the computational complexity of [1], [6], [8], [7] are $O(N^{5.5}K^{5.5})$, $O(N^{6.5}K^{6.5})$, $O(K^2N^{4.5} + K^3N^{3.5})$, and $O(N^{3.5})$

respectively¹. Note that for the equivalent optimization problem (22), where $K \leq N$, the computational complexity is at most $\mathcal{O}(N^{5.5})$. It must be emphasized that depending on values of N and K , we can implement the optimization problem (21) or (22) which results in less computational complexity.

Note that many of the constraints have very simple structures. This property can be used to develop an interior-point optimization algorithm fully exploiting the constraint structures of the problem, thereby getting complexity order better than that of using a general purpose solver such as SeDuMi or SDPA. Moreover, we can further reduce the complexity of the proposed method by implementing the augmented Lagrangian method [17].

ACKNOWLEDGEMENT

The authors would like to thank R. Sotirov, H. Wolkowicz and Y. Ding for many helpful discussions and comments. The authors would also like to thank T. Davidson, M. Kisiailiou, W. Ma and N. Sidiropoulos for their comments on an earlier draft of this work.

REFERENCES

- [1] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "A near maximum likelihood decoding algorithm for mimo systems based on semi-definite programming," *to appear in IEEE Trans. on Info. Theory*, Nov. 2007.
- [2] B. Steingrimsson, T. Luo, and K. M. Wong, "Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels," *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2710– 2719, Nov. 2003.
- [3] B. Steingrimsson, Z. Q. Luo, and K. M. Wong, "Quasi-ML detectors with soft output and low complexity for PSK modulated MIMO channels," in *4th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2003)*, 15-18 June 2003, pp. 427 – 431.
- [4] Z. Q. Luo, X. Luo, and M. Kisiailiou, "An Efficient Quasi-Maximum Likelihood Decoder for PSK Signals," in *ICASSP '03*, 2003.
- [5] W.-K. Ma, P. C. Ching, and Z. Ding, "Semidefinite relaxation based multiuser detection for m-ary psk multiuser systems," *IEEE Trans. on Signal Processing*, 2004.
- [6] A. Wiesel, Y. C. Eldar, and S. Shamai, "Semidefinite relaxation for detection of 16-QAM signaling in mimo channels," *IEEE Signal Processing Letters*, vol. 12, no. 9, Sept. 2005.

¹Due to space limit, we refer the reader to [1] for a comparison on the execution time of different methods.

- [7] N. D. Sidiropoulos and Z.-Q. Luo, "A semidefinite relaxation approach to mimo detection for high-order qam constellations," *IEEE SIGNAL PROCESSING LETTERS*, vol. 13, no. 9, Sept. 2006.
- [8] Y. Yang, C. Zhao, P. Zhou, and W. Xu, "Mimo detection of 16-qam signaling based on semidefinite relaxation," *to appear in IEEE SIGNAL PROCESSING LETTERS*, 2007.
- [9] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "A Near Maximum Likelihood Decoding Algorithm for MIMO Systems Based on Graph Partitioning," in *IEEE International Symposium on Information Theory*, Sept. 4-9 2005.
- [10] Y. Ding and H. Wolkowicz, "A Matrix-lifting Semidefinite Relaxation for the Quadratic Assignment Problem," Department of Combinatorics & Optimization, University of Waterloo, Tech. Rep. CORR 06-22, 2006.
- [11] A. Beck, "Quadratic matrix programming," *SIAM Journal on Optimization*, vol. 17, no. 4, pp. 1224–1238, 2007.
- [12] S. Hadley, F. Rendl, and H. Wolkowicz, "A new lower bound via projection for the quadratic assignment problem," *Math. Oper. Res.*, vol. 17, no. 3, pp. 727–739, 1992.
- [13] F. Alizadeh, J.-P. A. Haeberly, and M. Overton, "Primal-dual interior-point methods for semidefinite programming: Stability, convergence, and numerical results," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 746–768, 1998.
- [14] S. J. Benson and Y. Ye, "DSDP5 User Guide The Dual-Scaling Algorithm for Semidefinite Programming," Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, Tech. Rep. ANL/MCS-TM-255, 2004, available via the WWW site at <http://www.mcs.anl.gov/~benson>.
- [15] J. Sturm, "Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11-12, pp. 625–653, 1999.
- [16] M. Kojima, K. Fujisawa, K. Nakata, and M. Yamashita, "SDPA(SemiDefinite Programming Algorithm) Users Manual Version 6.00," Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo, 152-0033, Japan, Tech. Rep., 2002, available via the WWW site at <http://sdpa.is.titech.ac.jp/SDPA/>.
- [17] A. Mobasher and A. K. Khandani, "Matrix-Lifting SDP for Decoding in Multiple Antenna Systems," Department of E&CE, University of Waterloo, Tech. Rep. UW-E&CE 2007-06, 2007, available via the WWW site at <http://www.cst.uwaterloo.ca/~amin>.