A RANDOMIZATION METHOD FOR QUASI MAXIMUM LIKELIHOOD DECODING

Amin Mobasher*, Mahmoud Taherzadeh*, Renata Sotirov†, and Amir K. Khandani*

*Electrical and Computer Engineering Department
University of Waterloo, Waterloo, ON, Canada
Email:{amin, taherzad, khandani}@cst.uwaterloo.ca

†Department of Mathematics & Statistics
The University of Melbourne, Parkville, VIC, Australia
Email:rsotirov@ms.unimelb.edu.au

***Abstract*— In Multiple-Input Multiple-Output (MIMO) systems, Maximum-Likelihood (ML) decoding is equivalent to £nding the closest lattice point in an N dimensional complex space. In [1], we have proposed several quasi-maximum likelihood relaxation models for decoding in MIMO systems based on semi-de£nite programming. In this paper, we propose randomization algorithms that £nd a near-optimum solution of the decoding problem by exploring the solution of the corresponding semi-de£nite relaxations.**

1 INTRODUCTION

Recently, there has been a considerable interest in Multi-Input Multi-Output (MIMO) antenna systems due to achieving a very high capacity compared to single-antenna systems [2]. In MIMO systems, a vector is transmitted by transmit antennas. In the receiver, a corrupted version of this vector affected by the channel noise and fading is received. Decoding concerns the operation of receiving the transmitted vector from the received signal. This problem is usually expressed in terms of "lattice decoding" which is known to be NP-hard.

Quasi-maximum likelihood detection is a near optimum algorithm for lattice decoding based on a binary programming formulation and semi-de£nite relaxation [1], [3]. More precisely, the distance minimization in the Euclidean space is formulated in terms of a binary quadratic minimization problem. Then, the resulting problem is transformed into a relaxation problem using Semi-De£nite Programming (SDP). The solution for the distance minimization problem is a rank-one binary matrix. However, the rank-one constraint is removed in the relaxation problem. Therefore, the solution for the relaxation problems is not necessarily a binary rank-one matrix. This solution is changed to a 0–1 rank-one matrix through a randomization procedure. The extreme points of the feasible set for the relaxation problem are the binary rank-one matrices. The randomization procedure determinants some of the extreme points by using a solution of the SDP relaxation. Among these extreme points, the one which results in the smallest value for the distance-minimization objective function is chosen as the solution point.

The randomization procedure in [3] is based on $\{-1, 1\}$ elements. Usually, communication applications deal with 0–1 vectors, and the formulation of the problem with $\{-1, 1\}$ elements is not always a simple task. Here, we propose a method that depends on the bit values, $\{0, 1\}$. Also, with a smaller number of iterations it achieves a better performance compared to those which are relying on $\{-1, 1\}$ elements.

2 MIMO SYSTEM MODEL

A MIMO system with $\tilde{N}$ transmit antenna and $\tilde{M}$ receive antenna is modelled as

$$\tilde{\mathbf{y}} = \sqrt{\frac{SNR}{\tilde{M}\tilde{E}_{s_{av}}}}\tilde{\mathbf{H}}\tilde{\mathbf{x}} + \tilde{\mathbf{n}}, \tag{1}$$

where $\tilde{\mathbf{H}} = \left[\tilde{h}_{ij}\right]$ is the $\tilde{M} \times \tilde{N}$ channel matrix with independent, identically distributed complex Gaussian random variables with zero mean and unit variance, $\tilde{\mathbf{n}}$ is an $\tilde{M} \times 1$ complex additive white Gaussian noise vector with zero mean and unit variance elements, and $\tilde{\mathbf{x}}$ is an $\tilde{N} \times 1$ vector whose components are the signals sent from each transmit antenna and selected from a complex set $\tilde{\mathcal{S}} = \{\tilde{s}_1, \tilde{s}_2, \cdots, \tilde{s}_K\}$ with the average energy $\tilde{E}_{s_{av}}$. The parameter SNR in (1) is the Signal to Noise Ratio (SNR) per receive antenna.

Noting $\tilde{x}_i \in \mathcal{S}$, for $i = 1, \cdots, \tilde{N}$ we have

$$\tilde{x}_i = u_i(1)\tilde{s}_1 + u_i(2)\tilde{s}_2 + \cdots + u_i(K)\tilde{s}_K, \tag{2}$$

where

$$u_i(j) \in \{0, 1\} \text{ and } \sum_{j=1}^{K} u_i(j) = 1 \; \forall\, i = 1, \cdots, \tilde{N}. \tag{3}$$

Let $\mathbf{u} = \left[u_1(1) \;\cdots\; u_1(K) \;\cdots\; u_N(1) \;\cdots\; u_N(K)\right]^T$ and $N = \tilde{N}$. Using the equations in (2) and (3), the transmitted vector is expressed as

$$\tilde{\mathbf{x}} = \tilde{\mathbf{S}}\mathbf{u}, \tag{4}$$

where $\tilde{\mathbf{S}} = \mathbf{I}_N \otimes [\tilde{s}_1, \cdots, \tilde{s}_K]$ is an $N \times NK$ matrix of coef£cients, $\mathbf{I}_N$ is an $N \times N$ Identity matrix, $\otimes$ is the tensor product, and $\mathbf{u}$ is an $NK \times 1$ binary vector such that $\mathbf{Au} = \mathbf{e}_N$, where $\mathbf{A} = \mathbf{I}_N \otimes \mathbf{e}_K^T$ and $\mathbf{e}_N$ is an $N \times 1$ vector of all ones. This constraint states that among each K components of the binary vector $\mathbf{u}$, i.e. $u_i(1), \cdots, u_i(K)$, there is only one element equal to "1".

To avoid using complex matrices, the system model (1) is represented by real matrices in (5).

$$\begin{bmatrix} \Re(\tilde{\mathbf{y}}) \\ \Im(\tilde{\mathbf{y}}) \end{bmatrix} = \sqrt{\frac{SNR}{\tilde{M}\tilde{E}_{s_{av}}}} \begin{bmatrix} \Re\left(\tilde{\mathbf{H}}\right) & \Im\left(\tilde{\mathbf{H}}\right) \\ -\Im\left(\tilde{\mathbf{H}}\right) & \Re\left(\tilde{\mathbf{H}}\right) \end{bmatrix} \begin{bmatrix} \Re(\tilde{\mathbf{x}}) \\ \Im(\tilde{\mathbf{x}}) \end{bmatrix} + \begin{bmatrix} \Re(\tilde{\mathbf{n}}) \\ \Im(\tilde{\mathbf{n}}) \end{bmatrix} \Rightarrow \mathbf{y} = \mathbf{Hx} + \mathbf{n}, \quad (5)$$

where $\Re(.)$ and $\Im(.)$ denote the real and imaginary parts of a matrix, respectively, $\mathbf{y}$ is the *received vector*, and $\mathbf{x}$ is the *input vector*.

Let $\mathbf{S}$ denotes the real matrix $\begin{bmatrix} \Re\left(\tilde{\mathbf{S}}\right) \\ \Im\left(\tilde{\mathbf{S}}\right) \end{bmatrix}$; therefore,

$$\mathbf{y} = \mathbf{HSu} + \mathbf{n} \quad (6)$$

expresses the MIMO system model by real matrices and the input binary vector, $\mathbf{u}$.

The maximum-likelihood (ML) detector in MIMO systems is equivalent to

$$\hat{\mathbf{x}} = \arg \min_{s.t.\ x_i \in \mathcal{S}} \|\hat{\mathbf{y}} - \mathbf{Hx}\|^2 \quad (7)$$

In [4], it is shown that the decoding minimization problem can be expressed as:

$$\begin{aligned} \min\ & \mathbf{u}^T\mathbf{Qu} + 2\mathbf{c}^T\mathbf{u} \\ s.t.\ & \mathbf{Au} = \mathbf{e}_N \\ & u_i \in \{0,1\}^n, \end{aligned} \quad (8)$$

where $n = NK$, $\mathbf{Q} = \mathbf{S}^T\mathbf{H}^T\mathbf{HS}$, $\mathbf{c} = -\mathbf{S}^T\mathbf{H}^T\hat{\mathbf{y}}$, and $\mathbf{s} = [s_1, \cdots, s_K]^T$.

3 SEMI-DEFINITE RELAXATION SOLUTION

In [1], a preliminary SDP relaxation of the minimization problem has obtained by removing a rank-one constraint in the problem and using Lagrangian duality [5]. This relaxation has many redundant constraints and no strict interior for the feasible set. There are numerical dif£culties in £nding the solution for a problem without an interior point.

To overcome this drawback, the feasible set has been projected onto a face of the semi-de£nite cone [1] and based on the identi£ed redundant constraints, another form of the relaxation has obtained. The resulting relaxation has strict interior, and; therefore, one can compute the solution of the problem by an interior point method. By investigating on the structure of the feasible set, the relaxation has strengthen by a set of new constraints that impose a zero pattern for the solution.

Let

$$\mathcal{L}_{\mathbf{Q}} = \left[\begin{array}{c|c} 0 & \mathbf{c}^T \\ \hline \mathbf{c} & \mathbf{Q} \end{array}\right], \quad \mathbf{V}_K = \left[\begin{array}{c} \mathbf{I}_{K-1} \\ \hline -\mathbf{e}_{K-1}^T \end{array}\right],$$

$$\hat{\mathbf{V}} = \left[\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \frac{1}{K}(\mathbf{e}_{KN} - (\mathbf{I}_N \otimes \mathbf{V}_K)\mathbf{e}_{(K-1)N}) & \mathbf{I}_N \otimes \mathbf{V}_K \end{array}\right]. \quad (9)$$

The resulting relaxation problem is [4]

$$\begin{aligned} \min\ & \text{trace}(\hat{\mathbf{V}}^T\mathcal{L}_{\mathbf{Q}}\hat{\mathbf{V}})\mathbf{R} \\ \text{s.t.}\ & \mathcal{G}_{\bar{J}}(\hat{\mathbf{V}}\mathbf{R}\hat{\mathbf{V}}^{\mathbf{T}}) = \mathbf{E}_{00} \\ & \mathbf{R} \succeq 0, \end{aligned} \quad (10)$$

where $\mathbf{R}$ is our variable matrix of dimension $(N(K-1)+1)\times(N(K-1)+1)$, $\mathbf{E}_{00}$ is an $(NK+1)\otimes(NK+1)$ all zero matrix except one element equal to 1 in its $(0,0)$th entry, and $\bar{J}$ is the set of indices in (11) [4]

$$\begin{aligned} \bar{J} = \{ & (i,j):\ i = K(p-1)+q,\ j = K(p-1)+r, \\ & q<r, q,r \in \{1,\cdots,K\}, p \in \{1,\cdots,N\}\} \\ & \bigcup \{(0,0)\}. \end{aligned} \quad (11)$$

The operator $\mathcal{G}$ is known as the *gangster operator*, and for a matrix $\mathbf{Y}$ and a given set of indices J, it is de£ned by

$$(\mathcal{G}_J(\mathbf{Y}))_{ij} = \begin{cases} Y_{ij} & \text{if } (i,j) \text{ or } (j,i) \in J \\ 0 & \text{otherwise.} \end{cases} \quad (12)$$

The relaxation in (10) is further tightened by considering the *non-negativity constraints* [6]. All the elements of the matrix $\mathbf{Y}$ which are not covered by the equality constraints in (10) are equal to or greater than zero. These inequalities can be added to the set of constraints that we have in (10) [4].

$$\begin{aligned} \min\ & \text{trace}(\hat{\mathbf{V}}^T\mathcal{L}_{\mathbf{Q}}\hat{\mathbf{V}})\mathbf{R} \\ \text{s.t.}\ & \mathcal{G}_{\bar{J}}(\hat{\mathbf{V}}\mathbf{R}\hat{\mathbf{V}}^{\mathbf{T}}) = \mathbf{E}_{00} \\ & \mathcal{G}_{\hat{J}}(\hat{\mathbf{V}}\mathbf{R}\hat{\mathbf{V}}^{\mathbf{T}}) \geq 0 \\ & \mathbf{R} \succeq 0, \end{aligned} \quad (13)$$

where the set $\hat{J}$ indicates those indices which are not covered by $\bar{J}$. Note that there is a trade-off between the Bit Error Rate (BER) performance and the complexity of the decoding methods built on the introduced models. The decoding method based on the model in (13), compared to the model in (10), performs better in the sense of BER, but with more computational complexity.

The most common method for solving SDP problems of a moderate size is the Interior-Point Method (IPM). The computational complexity of IPM is polynomial. Nowadays, there are several IPM based solvers for solving SDP problems, e.g., DSDP, SeDuMi, SDPA, etc. In our numerical experiments, we implement SeDuMi and SDPA software packages.

Solving the minimization problem (10) or (13) results in a solution for the matrix $\mathbf{R}$ which is transformed to the matrix $\mathbf{Y}$ by

$$\mathbf{Y} = \hat{\mathbf{V}}\mathbf{R}\hat{\mathbf{V}}^T. \tag{14}$$

It can be shown [4] that if the matrix $\mathbf{Y}$ is restricted to be rank-one, then the optimal solution $\mathbf{u}$ for (8) can be found by

$$\mathbf{Y} = \begin{bmatrix} 1 \\ \mathbf{u} \end{bmatrix} \left[1\ \mathbf{u}^T\right] = \left[\begin{array}{c|c} 1 & \mathbf{u}^T \\ \hline \mathbf{u} & \mathbf{u}\mathbf{u}^T \end{array}\right]. \tag{15}$$

The optimization problems in (10) and (13) result in a solution that is not necessarily a binary rank-one matrix. This solution is changed to a 0–1 rank-one matrix through a randomization algorithm. We change the conventional randomization algorithms to £t our problem. Also, a new randomization procedure, with a better decoding performance as that in the common methods, is introduced which £nds the optimal binary rank-one solution with fewer iterations compared to the conventional methods.

4 RANDOMIZATION PROCEDURE

As mentioned in Section III, the SDP relaxation models (10) and (13) lead to a matrix $\mathbf{Y}$ whose elements are between 0 and 1. In order to change this matrix to an appropriate solution for (8), the simplest way is to use the properties of the optimal case.

Since the entries of $\mathbf{u}$ are binary numbers, the £rst row/column of the symmetric matrix in the optimal case of (15) is equal to its diagonal. In [4], it is shown that this statement is valid for any matrix $\mathbf{Y}$ resulted from the relaxation problems (10) or (13). Therefore, the vector $\mathbf{u}$ is approximated by rounding off the elements of the £rst row/column/diagonal of the matrix $\mathbf{Y}$. This transformation results in a loose upper bound for the BER performance. However, the transformation of $\mathbf{Y}$ to a rank-one matrix through a randomization procedure results in a better solution for (8). In the following, two randomization algorithms are considered for this transformation.

Algorithm I: Goemans and Williamson [7] introduced an algorithm that randomly rounds the solution to a semi-de£nite programming relaxation. This approach is followed in [3] for the quasi maximum likelihood decoding of a PSK signalling. This technique is based on expressing the BPSK symbols by $\{-1, 1\}$ elements. After solving the relaxation problem in [3], the Cholesky factorization is applied to the $n \times n$ matrix $\mathbf{Y}$ and the Cholesky factor $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n]$ is computed, i.e. $\mathbf{Y} = \mathbf{V}\mathbf{V}^T$. In [3], it is observed that one can approximate the solution of the distance minimization problem, $\mathbf{u}$, using $\mathbf{V}$, i.e. u_i is approximated using $\mathbf{v}_i$. Thus, the assignment of -1 or 1 to the vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ is equivalent to specifying the elements of $\mathbf{u}$. Fig. 1 shows how the method in [7] works. It is shown that norm

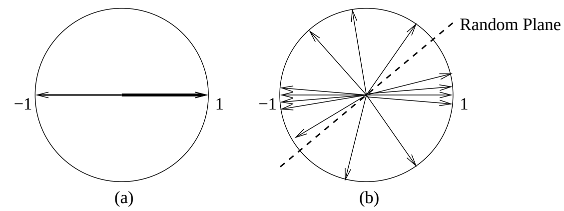

Fig. 1. Graphic representation for the randomization algorithm in [7]

of the vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ is one, and they are inside an n–dimensional unit sphere [3]. In the ideal case, these vectors should be classi£ed in two different groups corresponding to 1 and -1, see Fig. 1.(a). However, Fig. 1.(b) shows the general case for vectors $\mathbf{v}_i$ when $\mathbf{Y}$ is not a rank-one matrix. In order to assign -1 or 1 to these vectors, the randomization procedure generates a random vector uniformly distributed on this sphere. This vector de£nes a plane crossing the origin. Among given vectors $\mathbf{v}_i$, $i = 1, \ldots n$, all the vectors at one side of the plane are assigned to 1 and the remaining vectors are assigned to -1. This procedure is repeated several times and the vector $\mathbf{u}$ resulting in the lowest objective function is selected as the answer.

In our proposed approach, the variables are binary numbers. In order to implement the randomization procedure of [7], we bijectively map the computed solution of the $\{0, 1\}$ SDP formulation to the solution of the corresponding $\{-1, 1\}$ SDP formulation. More precisely, we use the following mapping:

$$\mathbf{T} = \left[\begin{array}{c|ccc} 1 & 0 & \cdots & 0 \\ \hline -1 & 2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & 0 & \cdots & 2 \end{array}\right],$$

$$\mathbf{Y}_{\{-1,1\}} = \mathbf{T}\mathbf{Y}_{\{0,1\}}\mathbf{T}^T. \tag{16}$$

Now, the solution for (8) can be found using a similar randomization method as in [3]. The computational complexity of this randomization algorithm is polynomial time [3].

Algorithm II: Most communication applications deal with binary elements. Also, formulating the problem with $\{-1, 1\}$ elements is not always a simple task. Here, we propose a new zero-one randomization procedure following the randomization procedure in [7]. This algorithm can be applied to $\{0, 1\}$ problem formulation directly. Therefore, the complexity of the whole randomization procedure is reduced, since the preprocessing step - bijective mapping form one to another model formulation - is omitted.

After solving the relaxation problem (10) or (13), the cholesky factorization of $\mathbf{Y}$ results in a matrix $\mathbf{V} = [\mathbf{v_1}, \ldots, \mathbf{v_n}]$ such that $\mathbf{Y} = \mathbf{V}\mathbf{V}^T$. In the ideal case of (15), the elements of $\mathbf{u}$, is either zero or one. Since the solution of the relaxation problem does not lead to a rank-one binary matrix $\mathbf{Y}$, the norm of resulting vectors $\mathbf{v}_i$ is between zero and one. These vectors are depicted in Fig. 2. It is clear that a sphere with a random radius uniformly distributed between zero and one has the same functionality as the random plane in Fig. 1. Now, in order to assign 0 or 1 to these vectors, the randomization procedure generates a random number uniformly distributed between 0 and 1. Among given vectors $\mathbf{v}_i$, $i = 1, \ldots n$, all the vectors whose norms are larger than this number are assigned to 1 and the remaining vectors are assigned to 0. This procedure is repeated several times and the vector $\mathbf{u}$ resulting in the lowest objective function is selected as the answer. Simulation results con£rms that the proposed method results in a better Symbol Error Rates (SER) performance for the lattice decoding problem compared to SER resulting from the method in [7]. Also, the computational complexity of the randomization algorithm is decreased, due to the removal of the preprocessing step.

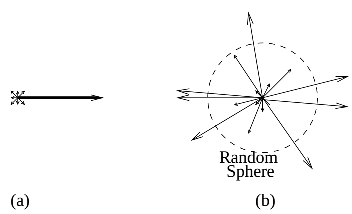

Fig. 2. Graphic representation for the proposed randomization algorithm

5 SIMULATION RESULTS

In our simulations, we have considered a MIMO system with 4 transmit and 4 receive antennas employing 16QAM. We show the SER of the proposed algorithms and ML decoding vs. the signal to noise ratio per bit, E_b/N_0. To solve the minimization problem in (10), we use either SDPA or DSDP packages. One of the main advantages of the model (10) is that all the constraints represented by the gangster operator can be represented by rank-one constant matrices. Both SDPA and DSDP packages have special considerations for these kinds of constraints. Specially, in the DSDP package, the complexity of solving a problem having rank-one constant matrices is considerably decreased. The randomization procedure implemented for this model is based on algorithm II. Fig. 3 shows the effect of using randomization algorithm I and II on the relaxation model (10).

The solution of the relaxation model in (13), for the most tested instances, corresponds to the optimal solution of the original problem (8). In the other words, because the model in (13) is strong enough, there is no need for the randomization algorithm. Comparison between the SER of the proposed algorithm and the ML decoding for the relaxation model (13) is presented in Fig. 3 [1]. Note that there is a trade-off between the complexity of using the non-negative constraints and that in the randomization procedure. Several compromises for improving the performance of the ML decoding can be done, e.g., including only some of the non-negative constraints in (13) and/or using a randomization procedure with a fewer number of iterations.

REFERENCES

[1] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "An Ef£cient Quasi-Maximum Likelihood Decoding for Finite Constellations," in *Conference on Information Sciences and Systems (CISS) 2005*, March 16-18 2005.

[2] E. Telatar, "Capacity of multi-antenna gaussian channels," *European Trans. on Telecomm. ETT*, vol. 10, no. 6, pp. 585–596, November 1999.

[3] 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.

[4] A. Mobasher, M. Taherzadeh, R. Sotirov, and A. K. Khandani, "An Ef£cient Quasi-Maximum Likelihood Decoding for Finite Constellations," Department of E&CE, University of Waterloo, Tech. Rep. UW-E&CE 2005-01, 2005, available via the WWW site at http://www.cst.uwaterloo.ca/∼amin.

[5] H. Wolkowicz, R. Saigal, and L. Vandenberghe, *Handbook of Semide£nite Programming: Theory, Algorithms, and Applications*. Kluwer, 2000.

[6] R. Sotirov and F. Rendl, "Bounds for the Quadratic Assignment Problem Using the Bundle Method," Department of Mathematics, University of Klagenfurt, Austria, Tech. Rep., 2003, available at http://www.ms.unimelb.edu.au/∼rsotirov.

[7] M. Goemans and D. Williamson, "Improved approximation algorithms for maximum cut and satisfability problem using semide£nite programming," *Journal of ACM*, vol. 42, pp. 1115–1145, 1995.

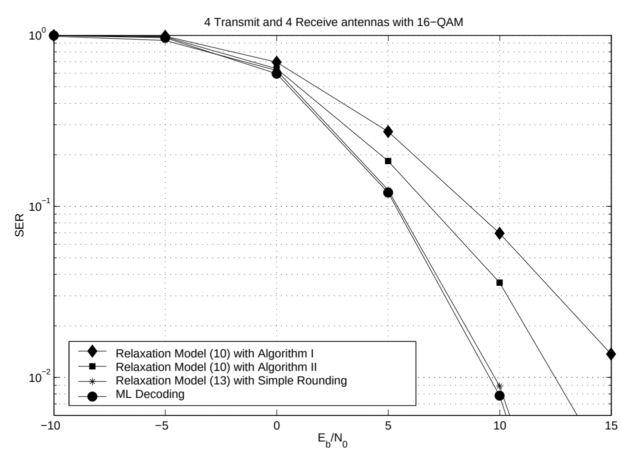

Fig. 3. Symbol Error Rates for the proposed algorithm and ML Decoding for different relaxation models and randomization algorithms