Application of Computational Geometry to Multiuser Detection in CDMA

Girish Manglani and A. K. Chaturvedi, *Senior Member, IEEE*

***Abstract*—The maximum-likelihood multiuser detection problem in code-division multiple-access is known to be an optimization problem with an objective function that is required to be optimized over a combinatorial decision region. Conventional suboptimal detectors relax the combinatorial decision region by a convex region, without altering the objective function to be optimized. We take an approach wherein the objective function is reduced to a form appropriate for the application of a polynomial complexity algorithm in computational geometry, while keeping the decision region combinatorial. The resulting detector allows a tradeoff between performance and computational complexity. The bit-error rate performance of the detector has been found to be better than the decorrelator and the linear minimum mean-square error detectors, for the same level of complexity.**

***Index Terms*—Complexity, computational geometry (CG), multiuser detection (MUD), optimization methods.**

I. Introduction

THE maximum-likelihood (ML) multiuser detection (MUD) problem is equivalent to the problem of optimization of a quadratic function over the corners of a hypercube. The ML decision for the estimated vector $\hat{\mathbf{b}}$ is [1, pp. 162]

$$\hat{\mathbf{b}} = \arg \min_{\mathbf{b}\in\{\pm 1\}^K} \mathbf{b}^T\mathbf{H}\mathbf{b} - 2\mathbf{b}^T\mathbf{A}\mathbf{y} \tag{1}$$

where $\mathbf{H} = \mathbf{ARA}$ is the unnormalized correlation matrix, $\mathbf{R}$ is the normalized correlation matrix, $\mathbf{A}$ is a diagonal matrix which gives the received amplitudes of the signals of various users, $\mathbf{y}$ is a vector consisting of the matched-filter outputs, $\mathbf{b}$ is the vector containing the information bits of the users, and K is the number of bits to be detected. The problem in (1) is NP-hard for arbitrary correlation matrices, thereby requiring the use of suboptimal detectors [2].

The MUD problem has two facets: the objective function $\mathbf{b}^T\mathbf{H}\mathbf{b} - 2\mathbf{b}^T\mathbf{A}\mathbf{y}$, which is the function to be minimized, and the constraint set $\mathbf{b} \in \{\pm 1\}^K$, which is the region over which the objective function is to be minimized. Conventional suboptimal detectors, e.g., decorrelator, soft interference canceler, and minimum mean-square error (MMSE) detector, solve the problem by relaxing the constraint set to various convex regions, without modifying the objective function. Thus the problem is reduced to the optimization of a convex function over a convex set, and can be solved using polynomial complexity algorithms [3]. Contrary to this, we propose an approach wherein the objective function is reduced to a form suitable for the application of a computational geometry (CG) algorithm, while keeping the constraint set unchanged. This also results in a detector which can solve the MUD problem with polynomial complexity, while offering an interesting tradeoff between complexity and performance. However, the detector is limited to binary phase-shift keying (BPSK) and quaternary phase-shift keying (QPSK) modulations. Section II explains a CG algorithm, and Section III converts the MUD problem to a form suitable for solution using this CG algorithm. Section IV quantifies the computational complexity of the resulting detector, while Section V illustrates the simulation results, and Section VI gives the concluding remarks.

Paper approved by A. Host-Madsen, the Editor for Multiuser Communications of the IEEE Communications Society. Manuscript received December 10, 2003; revised October 19, 2004 and June 14, 2005.

G. Manglani was with the Department of Electrical Engineering, Indian Institute of Technology–Kanpur, Kanpur 208016, India. He is now with Reliance Infocomm, Mumbai 400709, India (e-mail:girish.manglani@relianceinfo.com).

A. K. Chaturvedi is with the Department of Electrical Engineering, Indian Institute of Technology–Kanpur, Kanpur 208016, India (e-mail:akc@iitk.ac.in).

Digital Object Identifier 10.1109/TCOMM.2005.863793

II. Computational Geometry Algorithm

We consider a CG algorithm which uses a solution procedure along the lines of the one given in [4]. The problem under consideration is a quadratic maximization problem in $\{-1, 1\}$ variables, whose standard form is

$$\max\ f(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x}, \quad \text{subject to } \mathbf{x} \in \{-1,\ 1\}^n \tag{2}$$

where $\mathbf{Q}$ is a real, nonnegative definite (ND) matrix of size $n \times n$ and rank r_q. $\mathbf{Q}$ can be decomposed into a summation of rank-one matrices such that

$$\mathbf{Q} = \sum_{i=1}^{r_q} \mathbf{v}_i\mathbf{v}_i^T = \mathbf{V}^T\mathbf{V} \tag{3}$$

where $\mathbf{v}_i\mathbf{v}_i^T$, $1 \le i \le r_q$, is a matrix of rank one, and $\mathbf{v}_i^T$ is the ith row of $\mathbf{V}$ which is of size $r_q \times n$. Equation (2) now becomes

$$\max\ f(\mathbf{x}) = \mathbf{x}^T\mathbf{Q}\mathbf{x} = \mathbf{x}^T\mathbf{V}^T\mathbf{V}\mathbf{x} = \left\|\mathbf{V}\mathbf{x}\right\|^2 = \sum_{i=1}^{r_q}(\mathbf{v}_i, \mathbf{x})^2 \tag{4}$$

subject to $\mathbf{x} \in \{-1, 1\}^n$, where $(\mathbf{p}, \mathbf{q})$ denotes the inner product of two vectors, $\mathbf{p}$ and $\mathbf{q}$.

Consider the hypercube $\mathbf{x} \in [-1, 1]^n$. The image of the hypercube under the mapping $R^n \to R^{r_q} : \mathbf{z} = \mathbf{V}\mathbf{x}$ is a convex polytope of dimension r_q, which we denote by Z. This is also known as a zonotope. Following the argument given in [4], the optimal value $\hat{f}$ of (4) can be expressed as

$$\hat{f} = \max_{\mathbf{x}\in\{-1,1\}^n} \sum_{i=1}^{r_q}(\mathbf{v}_i, \mathbf{x})^2 = \max_{\mathbf{x}\in[-1,1]^n} \sum_{i=1}^{r_q}(\mathbf{v}_i, \mathbf{x})^2.$$

Replacing $(\mathbf{v}_i, \mathbf{x})$ by z_i, we get

$$\hat{f} = \max_{\mathbf{z}\in Z} \sum_{i=1}^{r_q} z_i^2. \tag{5}$$

Let us denote the set of extreme points of Z by S. Then (5) can be written as

$$\hat{f} = \max_{\mathbf{z}\in S} \sum_{i=1}^{r_q} z_i^2. \tag{6}$$

The set of extreme points of the hypercube corresponding to the extreme points of the zonotope is denoted by set T. Hence, the solution of the quadratic optimization problem can be obtained using the following equation :

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in T} \sum_{i=1}^{r_q} (\mathbf{v}_i, \mathbf{x})^2. \tag{7}$$

Set T can be obtained with $O(n^{(r_q-1)})$ computational complexity [4]. Thus, the smaller the value of r_q, the less is the complexity.

III. Multiuser Detection and the Computational Geometry Algorithm

The ML MUD problem is converted to a quadratic form, as in (2), for the application of the CG algorithm by rewriting (1) as [5]

$$\tilde{c}\tilde{\mathbf{b}} = \arg \max_{\mathbf{x}\in\{\pm 1\}^n} \mathbf{x}^T\mathbf{N}\mathbf{x} \tag{8}$$

where $n = K+1$, $c \in \{\pm 1\}$, $\mathbf{x} = (\mathbf{b}^T c)^T$ and

$$\mathbf{N} = \begin{pmatrix} -\mathbf{H} & \mathbf{Ay} \\ (\mathbf{Ay})^T & 0 \end{pmatrix}. \tag{9}$$

The estimated bit vector is given by

$$\hat{\mathbf{b}} = \tilde{c}\tilde{\mathbf{b}}. \tag{10}$$

However, to be able to use the CG algorithm to solve the problem in (8), it is required that $\mathbf{N}$ be ND, as can be observed from (2).

A. *Nonnegative Definiteness of* $\mathbf{N}$

Unfortunately, $\mathbf{N}$ is not ND, as can be seen from the following proposition.

Proposition 1: The matrix $\mathbf{N}$, as specified in (9), is not ND.

Proof: For any K-vector $\mathbf{p} = (p_1, p_2, \ldots, p_K)^T$ [1, pp. 20]

$$\mathbf{p}^T\mathbf{H}\mathbf{p} = \left\| \sum_{i=1}^{K} p_i A_i \mathbf{s}_i \right\|^2 \geq 0 \tag{11}$$

where A_i is the amplitude of the ith signal and $\mathbf{s}_i$ is the corresponding spreading sequence. Therefore, $\mathbf{H}$ is ND. Since $\mathbf{H}$ is ND, its eigenvalues are nonnegative [6, pp. 402]. The eigenvalues of $-\mathbf{H}$ are nonpositive. Let the eigenvalues of $-\mathbf{H}$ be arranged in ascending order

$$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{n-1}.$$

The eigenvalues of $\mathbf{N}$ are also arranged in ascending order

$$\tilde{\lambda}_1 \leq \tilde{\lambda}_2 \leq \cdots \leq \tilde{\lambda}_n.$$

Using the interlacing eigenvalues theorem for bordered matrices [6, pp. 185–186], we get

$$\tilde{\lambda}_1 \leq \lambda_1 \leq \tilde{\lambda}_2 \leq \lambda_2 \leq \cdots \leq \tilde{\lambda}_{n-1} \leq \lambda_{n-1} \leq \tilde{\lambda}_n. \tag{12}$$

Since the trace of $-\mathbf{H}$ and $\mathbf{N}$ is same, and the trace of a matrix is equal to the sum of its eigenvalues, it can be seen from (12) that unless $\tilde{\lambda}_1 = \lambda_1, \tilde{\lambda}_2 = \lambda_2, \ldots, \tilde{\lambda}_{n-1} = \lambda_{n-1}$, $\mathbf{N}$ will have one positive eigenvalue $\tilde{\lambda}_n$, and this will also be the only positive eigenvalue. □

Although $\mathbf{N}$ is not ND, it is Hermitian. The following proposition proves that by adding a diagonal matrix to $\mathbf{N}$, it can be transformed to an ND matrix without changing the maximum of the optimization problem stated in (8).

Proposition 2: Matrix $\mathbf{N}$ can be transformed to an ND form without changing the maximum of the optimization problem.

Proof: Since $\mathbf{N}$ is Hermitian, its eigenvalues are real [6, pp. 170–171]. Denote the lowest of the eigenvalues by $\tilde{\lambda}_{\min}$. Construct a matrix $\mathbf{M} = (\mathbf{N} - \tilde{\lambda}_{\min}\mathbf{I})$, by adding a diagonal matrix to $\mathbf{N}$. Let $\mathbf{g}_i$ be an eigenvector of $\mathbf{N}$, and $\tilde{\lambda}_i$ be the corresponding eigenvalue. The eigenvalues of $\mathbf{M}$ are obtained from

$$\mathbf{M}\mathbf{g}_i = (\mathbf{N} - \tilde{\lambda}_{\min}\mathbf{I})\mathbf{g}_i = (\tilde{\lambda}_i - \tilde{\lambda}_{\min})\mathbf{g}_i \tag{13}$$

where $i = 1, 2, \ldots, n$. From (13), it can be observed that all eigenvalues of $\mathbf{M}$ are nonnegative. By replacing $\mathbf{N}$ with $\mathbf{M}$ in (8), it can be seen that the maximum of the optimization problem does not change. □

$\mathbf{M}$ obtained through the application of *Proposition 2* to $\mathbf{N}$ will be an ND matrix of rank $(n-1)$. Now we are ready to introduce suboptimality, and thereby provide the tradeoff between complexity and performance. For this, we replace $\mathbf{M}$ by a lower rank matrix whose Frobenius norm is closest to $\mathbf{M}$. The complexity of construction of set T and evaluation of (7) will be exponential in the rank of this lower rank matrix.

B. *Reduction of* $\mathbf{M}$ *to* $\mathbf{Q}$

The singular value decomposition (SVD) of $\mathbf{M}$ can be used to obtain the desired lower rank matrix $\mathbf{Q}$ which is closest to $\mathbf{M}$ with respect to the Frobenius norm [6, pp. 427–428]. Since $\mathbf{M}$ is Hermitian, SVD is equivalent to eigenvalue decomposition. The eigenvalue decomposition of $\mathbf{M}$ is given by

$$\mathbf{M} = \sum_{i=1}^{n-1} \rho_i \mathbf{e}_i \mathbf{e}_i^T \tag{14}$$

where $\mathbf{e}_i$ is the ith eigenvector of $\mathbf{M}$ and $\rho_i = \tilde{\lambda}_i - \tilde{\lambda}_{\min}$ is the corresponding eigenvalue. The rank of $\mathbf{M}$ is $n-1$. Considering the largest $r_q \leq (n-1)$ eigenvalues and their corresponding eigenvectors, we obtain $\mathbf{Q}$ as

$$\mathbf{Q} = \sum_{i=1}^{r_q} \rho_i \mathbf{e}_i \mathbf{e}_i^T = \sum_{i=1}^{r_q} \mathbf{v}_i \mathbf{v}_i^T = \mathbf{V}^T\mathbf{V} \tag{15}$$

where $\mathbf{v}_i = \sqrt{\rho_i}\mathbf{e}_i$, $i = 1, 2, \ldots, r_q$. Since $1 \leq r_q \leq (n-1)$, $(n-1)$ different $\mathbf{Q}$'s can be obtained from $\mathbf{M}$, each giving a different tradeoff between complexity and performance. Such a $\mathbf{Q}$ can be substituted in (4) to solve the MUD problem. In Section II, it has been shown that the maximum of the problem in (4) is the same as the maximum of the problem in (7) obtained over the set T. Once the set T has been obtained, we have two options: either the eigenvectors (scaled by the square root of the corresponding eigenvalues) of the reduced matrix $\mathbf{Q}$, or the appropriately scaled eigenvectors of the unreduced matrix $\mathbf{M}$, can be used in (4). In the latter case, using the scaled eigenvectors of $\mathbf{M}$ from (14) in (4), we get

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in T} \sum_{i=1}^{n-1} (\mathbf{v}_i, \mathbf{x})^2 \tag{16}$$

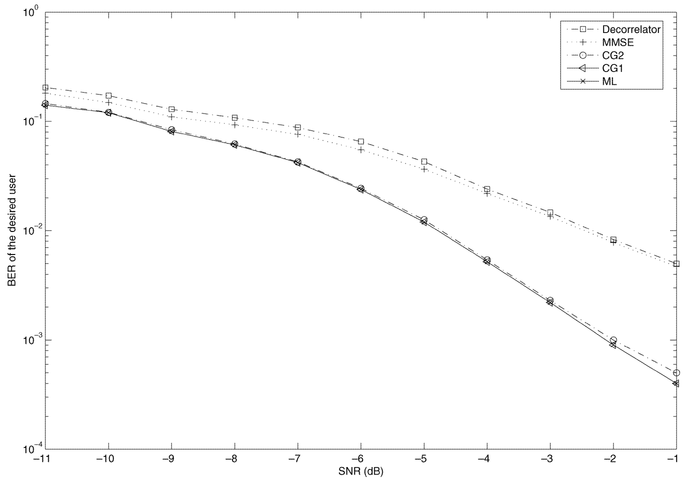

Fig. 1. Improvement in BER performance due to increase in r_q for an asynchronous system. Spreading sequence: PN, $L = 15$, $B = 3$, $U = 3$, $K = BU = 9$, and $n = 10$; CG1 is for $r_q = 4$ and CG2 is for $r_q = 3$. Curves for CG1 and ML multiuser detector completely overlap.

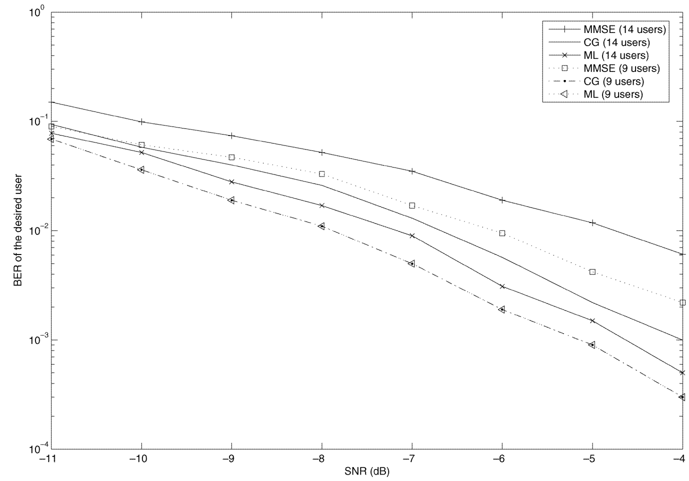

Fig. 2. Effect of number of bits to be detected on the BER in case of a synchronous system. Spreading sequence: random, $L = 31, K = U = 9, 14$; $n = 10, 15$ and $r_q = 3$. Curves for CG and ML multiuser detector completely overlap in the case of nine users.

where $\mathbf{v}_i = \sqrt{\rho_i}\mathbf{e}_i$, $i = 1, 2, \ldots, (n-1)$. Let us represent $\hat{\mathbf{x}}$ as $\hat{\mathbf{x}} = (\hat{\mathbf{x}}_{1n}^T \hat{x}_n)^T$. Hence, using (10), the estimated bit vector can be expressed as

$$\hat{\mathbf{b}} = \hat{x}_n \hat{\mathbf{x}}_{1n} \tag{17}$$

where $\hat{\mathbf{x}}_{1n} = \tilde{\mathbf{b}} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{n-1})^T$, $\hat{x}_n = \tilde{c}$, and $n = K + 1$. In this letter, we consider only the latter option.

IV. Computational Complexity of the Detector

The input matrix $\mathbf{N}$ has to be converted to an ND matrix $\mathbf{M}$, and subsequently reduced to a matrix $\mathbf{Q}$ of rank $r_q \leq (n-1)$ before the CG algorithm can be applied to the problem. This is necessary, irrespective of the rank of $\mathbf{N}$. An ND $\mathbf{M}$ can be reduced to $\mathbf{Q}$ using eigenvalue decomposition with a complexity of $O(n^3)$. This gives a lower bound on the complexity of the proposed detector.

Since $\mathbf{V}$ is of dimension $(n-1) \times n$, the complexity of evaluating (16) is of $O(n^{r_q})$. This can be shown easily using the analysis given in [7, pp. 286–288] and the fact that the number of extreme points of Z, and hence, the cardinality of set T, is of $O(n^{(r_q - 1)})$ [7, Ch. 1]. Thus for $r_q \geq 3$, the complexity is $O(n^{r_q})$ and equals the complexity of the ML multiuser detector, as well as its performance, when $r_q = (n-1)$. From the viewpoint of applications, the range of r_q that will be of interest is $r_q \leq 4$.

V. Simulation Results and Conclusions

Simulations have been carried out in an additive white Gaussian noise (AWGN) channel, keeping the noise power constant at -4 dB and the power of all interfering users constant at 0 dB. The power of the desired user is varied to attain the desired value of signal-to-noise ratio (SNR). The number of users is denoted by U; B denotes the number of bits per frame, and L denotes the length of the spreading sequence.

The difference between the Frobenius norms of $\mathbf{M}$ and $\mathbf{Q}$ decreases as the rank of $\mathbf{Q}$ is increased. This leads to better results, as is shown in Fig. 1, which gives the performance of the detector for an asynchronous system using PN sequences. The delays of different users have been chosen randomly and remain constant for a frame. The curves for the ML multiuser detector and the proposed detector (denoted by CG in all the curves) with $r_q = 4$ (CG1, complexity of $O(n^4)$) overlap, while the bit error rate (BER) performance when $r_q = 3$ (CG2, complexity of $O(n^3)$) is slightly inferior. Both CG1 and CG2 outperform the decorrelator and the linear MMSE, and also demonstrate the tradeoff involved between performance and complexity.

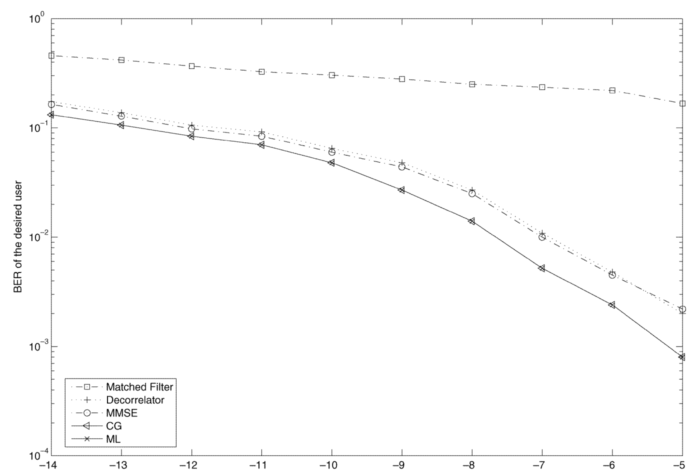

Fig. 3. BER performance for a synchronous system. Spreading sequence: Gold, $L = 31$, $K = U = 14$, $n = 15$, and $r_q = 3$. Curves for CG and ML multiuser detector completely overlap.

For a fixed rank of $\mathbf{Q}$, the difference between the Frobenius norms of $\mathbf{M}$ and $\mathbf{Q}$ increases as the size of $\mathbf{M}$ increases, and hence the BER performance gets worse as the number of bits to be detected increases. This is illustrated in Fig. 2 for a synchronous system, which also demonstrates the superior performance of the proposed detector over the linear MMSE, for the same complexity. The spreading sequences have been obtained using a random number generator and are different for every bit period for different users.

We have also simulated using Gold sequences for a synchronous system, and the BER performance is shown in Fig. 3. The curves for the proposed detector and the ML multiuser detector overlap, and this performance is better than the corresponding performance in Fig. 2. Better performance in the case of Gold sequences can be attributed to an eigenvalue of large magnitude, which leads to lesser difference between the Frobenius norms of $\mathbf{M}$ and $\mathbf{Q}$, as compared with the case when random spreading sequences are used.

REFERENCES

[1] S. Verdu, *Multiuser Detection*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[2] ——, "Computational complexity of optimum multiuser detection," *Algorithmica*, vol. 4, no. 4, pp. 303–312, 1989.

[3] A. Yener, R. D. Yates, and S. Ulukus, "CDMA multiuser detection: A nonlinear programming approach," *IEEE Trans. Commun.*, vol. 50, no. 6, pp. 1016–1024, Jun. 2002.

[4] K. Allemand, K. Fukuda, T. Liebling, and E. Steiner, "A polynomial case of unconstrained zero-one quadratic optimization," *Math. Program.*, vol. 91, no. 1, pp. 49–52, 2001.

[5] W.-K. Ma, T. Davidson, K. M. Wong, Z.-Q. Luo, and P.-C. Ching, "Quasi-maximum likelihood multiuser detection using semi-definite relaxation with application to synchronous CDMA," *IEEE Trans. Signal Process.*, vol. 50, no. 4, pp. 912–922, Apr. 2002.

[6] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.

[7] H. Edelsbrunner, *Algorithms in Combinatorial Geometry*. New York: Springer-Verlag, 1987.