An Efficient Regularization Approach for Underdetermined MIMO System Decoding

Xiao-Wen Chang
School of Computer Science
McGill University
Montreal, Quebec, Canada H3A 2A7
chang@cs.mcgill.ca

Xiaohua Yang
School of Computer Science
McGill University
Montreal, Quebec, Canada H3A 2A7
xyang35@cs.mcgill.ca

ABSTRACT

An efficient regularization approach is proposed for decoding underdetermined multiple input multiple output (MIMO) systems. The main idea is to transform an underdetermined integer least squares problem to an equivalent overdetermined integer least squares problem by using part of the transmit vector to do a regularization. Some strategies are proposed to enhance the efficiency of this approach. Specifically, we discuss how many entries of the transmit vector should be chosen and how to choose them when we do the regularization. An empirical formula for the regularization parameter is presented. Simulation results indicate that this modified approach can be much more efficient than current approaches for any square constellation higher than 4QAM.

Categories and Subject Descriptors

E.4 [**Coding and Information Theory**]: Formal models of communication; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems—*Computation of transforms, Computations on matrices*; G.1.6 [**Numerical Analysis**]: Optimization—*Constrained optimization, Least squares methods*

General Terms

Algorithms Performance

Keywords

regularization approach, underdetermined integer least squares, generalized sphere decoding, MIMO

1. INTRODUCTION

In Gaussian multi-input multi-output (MIMO) linear channels systems, the received signal vector is given by a linear combination of the flat-fading channel corrupted by additive noise. The relation between the received signal vector and

the transmit signal vector can be written as a complex linear system

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{H}}\tilde{\boldsymbol{x}} + \tilde{\boldsymbol{v}} \tag{1}$$

where $\tilde{\boldsymbol{H}} \in \mathbb{C}^{N_r \times N_t}$ represents the flat-fading channel with N_t transmitter antennas and N_r receiver antennas, and the elements of $\tilde{\boldsymbol{H}}$ are complex i.i.d Gaussian variables with (normalized) distribution $CN(0,1)$, $\tilde{\boldsymbol{v}} \in \mathbb{C}^{N_r}$ is the white Gaussian noise vector with distribution $CN(\boldsymbol{0}, 2\sigma^2\boldsymbol{I})$, and $\tilde{\boldsymbol{x}} \in \mathbb{C}^{N_t}$ is the unknown signal vector and its elements are odd numbers in the finite set

$$\tilde{\mathcal{X}}_k = \{k_1 + k_2 j\}$$

where $k_1, k_2 = \pm 1, \pm 3, \cdots, \pm(2^k - 3), \pm(2^k - 1)$, and $j^2 = -1$. Note that $k = 1, 2, 3$ corresponds to QPSK (i.e., 4QAM), 16QAM, 64QAM constellations, respectively. The decoding problem is to estimate $\tilde{\boldsymbol{x}}$ in (1).

In order to avoid complex computations, we first transform the complex linear system (1) to a real one. For each vector or matrix in (1) we separate its real part and imaginary part:

$$\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{y}}^R + j\tilde{\boldsymbol{y}}^I, \qquad \tilde{\boldsymbol{H}} = \tilde{\boldsymbol{H}}^R + j\tilde{\boldsymbol{H}}^I,$$
$$\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{x}}^R + j\tilde{\boldsymbol{x}}^I, \qquad \tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}^R + j\tilde{\boldsymbol{v}}^I.$$

Then it is easy to show that (1) is equivalent to the following real linear system

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v} \tag{2}$$

where

$$\boldsymbol{y} = \begin{bmatrix} \tilde{\boldsymbol{y}}^R \\ \tilde{\boldsymbol{y}}^I \end{bmatrix}, \quad \boldsymbol{H} = \begin{bmatrix} \tilde{\boldsymbol{H}}^R & -\tilde{\boldsymbol{H}}^I \\ \tilde{\boldsymbol{H}}^I & \tilde{\boldsymbol{H}}^R \end{bmatrix}, \boldsymbol{x} = \begin{bmatrix} \tilde{\boldsymbol{x}}^R \\ \tilde{\boldsymbol{x}}^I \end{bmatrix}, \quad \boldsymbol{v} = \begin{bmatrix} \tilde{\boldsymbol{v}}^R \\ \tilde{\boldsymbol{v}}^I \end{bmatrix}.$$

Obviously $\boldsymbol{H} \in \mathbb{R}^{m \times n}$ with $m \triangleq 2N_r$, $n \triangleq 2N_t$ and $h_{ij} \sim N(0, 1/2)$, $\boldsymbol{v} \sim N(0, \sigma^2\boldsymbol{I}_m)$, and $\boldsymbol{x} \in \mathcal{X}_k^n$ with

$$\mathcal{X}_k \triangleq \{\pm 1, \pm 3, \cdots, \pm(2^k - 3), \pm(2^k - 1)\}. \tag{3}$$

In order to estimate the complex transmit vector $\tilde{\boldsymbol{x}}$ in (1) or the real "transmit" vector $\boldsymbol{x}$ in (2), one solves the following minimization problem

$$\min_{\boldsymbol{x} \in \mathcal{X}_k^n} \|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}\|_2^2, \tag{4}$$

which we refer to as a *box constrained integer least squares (ILS) problem*. This is also referred to as maximum likelihood decoding. When the matrix $\boldsymbol{H}$ has full column rank, a regular sphere decoding (SD) algorithm can be employed to

find the optimal solution of (4) , see, e.g., [1], [2], [3], [4] and [5]. In this paper, we are interested in the case that $\boldsymbol{H}$ has full row rank. One such application is the multiple-antenna communication systems where there are more transmitting antennas than receiving antennas. For this case, a regular SD algorithm cannot be applied directly. To solve this problem, Damen et al [6] proposed the first generalized sphere decoding (GSD) algorithm. Later, Dayal and Varanasi [7] proposed another GSD algorithm, to be called Algorithm DV in this paper for convenience, which can significantly reduce the computational complexity by partitioning the candidate set into disjoint ordered subsets. Recently, Yang et al [8] proposed a new GSD algorithm, to be referred to as Algorithm YLH, which is usually faster than Algorithm DV. More recently, Chang and Yang [9] proposed a recursive GSD algorithm, to be called Algorithm CY, which modified Algorithm DV and incorporated a column permutation strategy in reduction (or preprocessing), and is (much) faster than Algorithms DV and YLH. All the above algorithms mainly consider how to generate a sequence of determined sub-ILS problems. In [10], Cui and Tellambura proposed a different approach, which transforms the underdetermined problem (4) to an equivalent overdetermined problem so that a regular SD algorithm can then be applied. We refer to this approach as a regularization approach and the corresponding algorithm as Algorithm CT for convenience. The idea of this approach is first transforming the entire vector $\boldsymbol{x}$ into a new vector $\bar{\boldsymbol{x}}$ whose entries are either 1 or -1, and then transforming the underdetermined problem to an overdetermined problem by adding a constant term which involves the 2-norm of the entire vector $\bar{\boldsymbol{x}}$ to the objective function of the problem. If $k = 1$ in the constraint $\mathcal{X}_k$ in (3), our simulations indicated that Algorithm CT is the most efficient one among all the above decoding algorithms. However, for $k \geq 2$, our simulations showed that Algorithm CT is slower than Algorithm CY. For computational efficiency, some algorithms have been proposed to find a sub-optimal solution to (4), see, e.g., [11] and [12]. Such algorithms will not be discussed in this paper.

In this paper, we propose to modify Algorithm CT. The key idea is to choose part of the transmit vector $\boldsymbol{x}$ to do the transformation and then to do the regularization. This can greatly reduce the computational complexity for $k \geq 2$. We will discuss some issues involved in this approach and propose our strategies.

The rest of this contribution is organized as follows. In Section 2 we introduce the main ideas of the generalized sphere decoding approach. In Section 3 we present our efficient regularization approach. In Section 3 we compare our method with Algorithm CT and discuss a few efficiency issues. Section 5 gives simulation results to show the efficiency of this approach. Finally a summary is given in Section 6.

2. THE GSD APPROACH

In this section we introduce basic ideas of the recent GSD algorithms for solving the underdetermined problem (4).

To solve (4), one transforms the problem to a new problem. This process is called reduction and can be realized by the QR factorization of the matrix $\boldsymbol{H}$: $\boldsymbol{H} = \boldsymbol{QR}$ where $\boldsymbol{Q} \in \mathbb{R}^{m \times m}$ is orthogonal and $\boldsymbol{R} \in \mathbb{R}^{m \times n}$ is an upper trapezoidal matrix. This QR factorization can be computed by the Householder transformations, see, e.g., [13, Sec 5.2]. Then with $\tilde{\boldsymbol{y}} \triangleq \boldsymbol{Q}^T \boldsymbol{y}$, the objective function in (4) can be

written as $\|\boldsymbol{y} - \boldsymbol{Hx}\|_2^2 = \|\boldsymbol{Q}^T \boldsymbol{y} - \boldsymbol{Rx}\|_2^2 = \|\tilde{\boldsymbol{y}} - \boldsymbol{Rx}\|_2^2$. So the problem (4) is reduced to

$$\min_{\boldsymbol{x} \in \mathcal{X}_k^n} \|\tilde{\boldsymbol{y}} - \boldsymbol{Rx}\|_2^2.$$

Partition the upper trapezoidal matrix $\boldsymbol{R} \in \mathbb{R}^{m \times n}$ and the vector $\boldsymbol{x} \in \mathbb{R}^n$ as

$$\boldsymbol{R} = [\underset{m}{\boldsymbol{R}_1}, \underset{n-m}{\boldsymbol{R}_2}], \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}^{(1)} \\ \boldsymbol{x}^{(2)} \end{bmatrix} \begin{matrix} m \\ n-m \end{matrix}. \qquad (5)$$

Notice that $\boldsymbol{R}_1$ is nonsingular upper triangular. Then we have

$$\min_{\boldsymbol{x} \in \mathcal{X}_k^n} \|\tilde{\boldsymbol{y}} - \boldsymbol{Rx}\|_2^2$$

$$= \min_{\boldsymbol{x}^{(2)} \in \mathcal{X}_k^{n-m}} \left(\min_{\boldsymbol{x}^{(1)} \in \mathcal{X}_k^m} \|(\tilde{\boldsymbol{y}} - \boldsymbol{R}_2 \boldsymbol{x}^{(2)}) - \boldsymbol{R}_1 \boldsymbol{x}^{(1)}\|_2^2 \right). \qquad (6)$$

The GSD algorithm in [6] first fixed $\boldsymbol{x}^{(2)}$, then employs a conventional SD to solve the bracketed minimization problem in (6). This is done by exhaustively trying every possible $\boldsymbol{x}^{(2)}$. Thus the GSD algorithm in [6] has an exponential complexity in $n - m$ independent of the signal to noise ratio (SNR). Obviously it is time-consuming to try each possible candidate for $\boldsymbol{x}^{(2)}$. In fact, it is not necessary to try each one. Motivated by this observation, [7] [8] [9] proposed different strategies to improve the efficiency.

3. EFFICIENT REGULARIZATION APPROACH

In this section, based on the regularization approach given in [10], we will present a modified approach to improve efficiency.

Partition $\boldsymbol{H}$ and $\boldsymbol{x}$ as follows

$$\boldsymbol{H} = [\underset{n-l}{\boldsymbol{H}_1}, \underset{l}{\boldsymbol{H}_2}] \, m \, , \qquad \boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{bmatrix} \begin{matrix} n-l \\ l \end{matrix} \qquad (7)$$

where we take $l = n - m + 1$. Following [10], we can write $\boldsymbol{x}_2 \in \mathcal{X}_k^l$ as a linear combination of $\boldsymbol{x}_2^{(i)} \in \mathcal{X}_1^l$ for $0 \leq i \leq k - 1$:

$$\boldsymbol{x}_2 = \sum_{i=0}^{k-1} 2^i \boldsymbol{x}_2^{(i)}. \qquad (8)$$

Define

$$\bar{\boldsymbol{H}}_2 \triangleq [\boldsymbol{H}_2, 2\boldsymbol{H}_2, \cdots, 2^{k-1} \boldsymbol{H}_2] \in \mathbb{R}^{m \times kl},$$
$$\bar{\boldsymbol{x}}_2 \triangleq \begin{bmatrix} \boldsymbol{x}_2^{(0)} \\ \boldsymbol{x}_2^{(1)} \\ \vdots \\ \boldsymbol{x}_2^{(k-1)} \end{bmatrix} \in \mathbb{R}^{kl}. \qquad (9)$$

Notice that $\|\bar{\boldsymbol{x}}_2\|_2^2 = kl$, so $\|\bar{\boldsymbol{x}}_2\|_2^2$ is a constant. Then from (7)-(9), we see that the original ILS problem (4) is equivalent to

$$\min_{\boldsymbol{x}_1 \in \mathcal{X}_k^{n-l}, \bar{\boldsymbol{x}}_2 \in \mathcal{X}_1^{kl}} \left\| \boldsymbol{y} - [\boldsymbol{H}_1, \bar{\boldsymbol{H}}_2] \begin{bmatrix} \boldsymbol{x}_1 \\ \bar{\boldsymbol{x}}_2 \end{bmatrix} \right\|_2^2 + \alpha^2 \|\bar{\boldsymbol{x}}_2\|_2^2 \qquad (10)$$

where we refer to α as a regularization parameter and its

choice will be discussed later. Therefore with

$$\bar{\boldsymbol{H}} \triangleq \begin{bmatrix} \boldsymbol{H}_1 & \bar{\boldsymbol{H}}_2 \\ \boldsymbol{0} & \alpha \boldsymbol{I} \end{bmatrix} \in \mathbb{R}^{(m+kl)\times(n+(k-1)l)}, \qquad (11)$$

$$\bar{\boldsymbol{x}} \triangleq \begin{bmatrix} \boldsymbol{x}_1 \\ \bar{\boldsymbol{x}}_2 \end{bmatrix} \in \mathbb{R}^{n+(k-1)l}, \qquad \bar{\boldsymbol{y}} \triangleq \begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{0} \end{bmatrix} \in \mathbb{R}^{m+kl}, \quad (12)$$

$$\bar{\mathcal{X}} \triangleq \{ [\begin{smallmatrix} \boldsymbol{x}_1 \\ \bar{\boldsymbol{x}}_2 \end{smallmatrix}] : \boldsymbol{x}_1 \in \mathcal{X}_k^{n-l}, \ \bar{\boldsymbol{x}}_2 \in \mathcal{X}_1^{kl} \}, \qquad (13)$$

the problem (10) can be rewritten as

$$\min_{\bar{\boldsymbol{x}} \in \bar{\mathcal{X}}} \| \bar{\boldsymbol{y}} - \bar{\boldsymbol{H}} \bar{\boldsymbol{x}} \|_2^2. \qquad (14)$$

This is a box constrained overdetermined ILS problem and can be solved by any regular SD algorithm which can handle the constraints. We will apply the Schnorr-Euchner strategy based search algorithm given in [5] and use the V-BLAST column reordering strategy in the reduction (or preprocessing) process (see [2] and [5]). Note that the reduction process is to transform $\bar{\boldsymbol{H}}$ to an upper triangular matrix whose dimension is $(n+(k-1)l)\times(n+(k-1)l)$ by a QR factorization with column reordering.

4. EFFICIENCY ISSUES

In the following, we discuss a few issues with the approach proposed in Sec. 3.

Unlike Algorithm CT in which $\boldsymbol{x}_2$ was chosen as the entire vector $\boldsymbol{x}$ (i.e., in (7) $l = n$), in our modified approach, $\boldsymbol{x}_2$ is part of $\boldsymbol{x}$ with dimension $l = n - m + 1$, thus the matrix $\bar{\boldsymbol{H}}$ in (11) is $(m+k(n-m+1))\times(n+(k-1)(n-m+1))$, while it is $(m+kn)\times(n+(k-1)n)$ in Algorithm CT. When $k = 1$, the numbers of columns of $\bar{\boldsymbol{H}}$ in the two approaches are the same, while the number of rows is smaller in the modified approach. Thus the reduction process will cost less for the modified approach, although this is not a big deal, since the search process dominates the cost of the entire algorithm. When $k \geq 2$, the number of columns of $\bar{\boldsymbol{H}}$ in the modified approach is smaller than that in Algorithm CT, while the constraint set for each entry of $\boldsymbol{x}_1$ is still $\mathcal{X}_k$, which is larger than $\mathcal{X}_1$, the constraint set for each entry of the unknown vector $\bar{\boldsymbol{x}}$ in Algorithm CT. In other words, compared with Algorithm CT, when $k \geq 2$, our modified approach decreases the dimension of the overdetermined ILS problem, while increases the ranges of part of the constraint sets. Our simulations (see Section 5) showed that this strategy can significantly decrease the computational complexity. This is related the nature of the search process. The Schnorr-Euchner strategy based search process can be regarded as a depth-first search (DFS) process on a tree, and our strategy actually decreases the height of the search tree, making the search process more efficient, see, e.g., [14, Chaps 12 and 13]) for explanations.

In our approach, the dimension l of $\boldsymbol{x}_2$ (see (7)) is chosen to be $n - m + 1$. Note that in order to obtain an overdetermined ILS problem (14), l has to be at least $n - m$. If the dimension of $\boldsymbol{x}_2$ were chosen to be $n - m$, then in the reduction process, the bottom right corner of the $\boldsymbol{R}$ factor of the QR factorization of $\bar{\boldsymbol{H}}$ in (11) would usually be diagonal even if we use V-BLAST column reordering strategy in computing the QR factorization. This structure would make the search algorithm less efficient. Choosing l to be $n - m + 1$ avoids the above problem. Certainly, choosing a larger dimension for $\boldsymbol{x}_2$ would also avoid this problem, but it would increase the dimension of the overdetermined ILS problem.

In (7) we uses the last $n - m + 1$ entries of $\boldsymbol{x}$ to define $\boldsymbol{x}_2$. But choosing a different $\boldsymbol{x}_2$ from $\boldsymbol{x}$ may make the efficiency of the algorithm to solve (14) different. Thus we would like to discuss how to choose good $\boldsymbol{x}_2$ from $\boldsymbol{x}$, or how to reorder the columns of the matrix $\boldsymbol{H}$ so that good $\boldsymbol{x}_2$ can be chosen. From simulations we found that sorting the columns of $\boldsymbol{H}$ in decreasing order with respect to the 2-norm helps to reduce the computational complexity of the search process when $k \geq 2$. The reason is that such ordering is likely to make the strict upper triangular part of the $\boldsymbol{R}$ factor of the QR factorization of $\bar{\boldsymbol{H}}$ smaller, which can make the search process more efficient, see, e.g., [15] and [16], for some discussion on this issue.

In the ILS problem (10), α is a parameter and its choice has an effect on the efficiency of the algorithm to solve the problem. It appears difficult if not impossible to find an optimal α that leads to the lowest computational complexity. In [10], α was chosen to be 1 or σ (the standard deviation of the noise $\boldsymbol{v}$ in the linear model (2)) in the simulations, and there was no other suggestion about how to choose a good α. We found from simulations that the optimal α in our modified approach depends strongly on σ and weakly on the dimension difference $n - m$ and k. If $n \leq 2m$ and $k \leq 3$ (this is often the case in applications), our simulations suggest that $\alpha = 2^{7/4}\sigma$ is a good choice.

5. SIMULATIONS

In this section we compare the computational cost of our algorithm outlined in Section 3 with Algorithm CT in [10] and Algorithm CY in [9]. For convenience, our modified algorithm is referred to as Algorithm MCT. All the simulations were performed in MATLAB 7.0.

The flat-fading channel matrix $\boldsymbol{H}$ and the receiver vector $\boldsymbol{y}$ were generated according to the real linear system (2). All the three algorithms employ the same regular sphere decoding algorithm for solving any overdetermined or determined box constrained ILS problems. This regular sphere decoding algorithm uses the Schnorr-Euchner strategy in the search process and the V-BLAST column reordering strategy in the reduction process, see [5] for details. The complexity of the three algorithms is measured by the number of flops. Only the flops of the search algorithm are counted without considering the cost of the reduction process, which is relatively negligible. For Algorithm CT, parameter α was set to 1 or σ as in [10], for Algorithm MCT, α was set to $2^{7/4}\sigma$. For each case, we performed 100 runs and counted average flops.

Fig. 1, Fig. 2 and Fig. 3 display the average flops of the three algorithms versus $N_t - N_r$ for 4QAM ($k = 1$), 16QAM ($k = 2$) and 64QAM ($k = 3$), respectively, and in all of these cases $\boldsymbol{v} \sim N(\boldsymbol{0}, 0.05^2 \boldsymbol{I}_m)$ and $N_r = 4$. For 4QAM, we see that there is small difference between the cost of Algorithm CT (for either $\alpha = 1$ or $\alpha = \sigma$) and the cost of Algorithm MCT. This is because the numbers of columns of $\bar{\boldsymbol{H}}$ in the two overdetermined ILS problems (see (14)) solved by the two algorithms, respectively, are the same when $k = 1$. When $N_t - N_r = 1, 2$, both Algorithm CT and Algorithm MCT cost more than Algorithm CY, and when $N_t - N_r = 3, 4$, the former cost (much) less than the latter. For 16QAM and 64QAM, Algorithm MCT is the most efficient one among the three decoding algorithms. For example, in Fig. 2, when $N_t - N_r = 1$, the average flops costed by Algorithm MCT and Algorithm CT with $\alpha = \sigma$ are 1.2×10^3 and 1.3×10^4, i.e., Algorithm MCT is about 11 times as fast

as Algorithm CT; and in Fig. 3, when $N_t - N_r = 1$, the average flops costed by Algorithm MCT and Algorithm CT with $\alpha = 1$ are 1.1×10^4 and 2.4×10^5, i.e., Algorithm MCT is about 22 times as fast as Algorithm CT. If $N_t - N_r$ is fixed, when k increases, Fig. 1–Fig. 3 indicate that Algorithm MCT becomes more and more efficient than Algorithm CT. This is because the difference $(k-1)(m-1)$ between the numbers of columns of the matrices $\bar{H}$ in the two overdetermined ILS problems solved by Algorithm CT and MCT respectively becomes larger and larger.

Fig. 4 shows the average flops of the three algorithms versus different SNR for 16QAM, $N_r = 4$ and $N_t = 5$. Here, for M-QAM, SNR is defined by SNR $= 10 \log_{10} \frac{(M-1)/3}{2\sigma^2}$. When SNR increases, Fig. 4 indicates that Algorithm MCT becomes more and more efficient than Algorithm CT. For SNR $= 12$dB and SNR $= 27$dB, the average flops costed by Algorithm CT with $\alpha = \sigma$ is about 7 times and 10 times as high as that by Algorithm MCT, respectively.

In Section 3, we introduced two strategies to improve the efficiency of our approach. One is about the choice of the dimension of x_2 and the other is about reordering the columns of H to find x_2. Fig. 5 shows the effects of these two strategies on efficiency (again here $v \sim N(\mathbf{0}, 0.05^2 I_m)$ and $N_r = 4$). All the three curves represent the cost of Algorithm MCT. The top curve is for the case that the dimension l of x_2 was set to $n - m$ and there was no column reordering of H in finding x_2; the middle curve is for the case that l was set to $n - m + 1$ and there was no reordering in finding x_2 either; and the bottom curve is for the case that l was set to $n - m + 1$ and x_2 was found by the reordering strategy. From Fig. 5, we see that indeed those two strategies can reduce the computational complexity of Algorithm MCT.

In order to see how different regularization parameter α in (14) affect the efficiency of Algorithms CT and MCT, Fig. 6 shows the costs of these two algorithms versus α with different setting. We observe that Algorithm CT is more sensitive to α than Algorithm MCT when $\alpha > 2$. When $\sigma = 0.02, 0.05, 0.5$, our choice $\alpha = 2^{7/4}\sigma \approx 0.07, 0.17, 1.7$. We see from Fig. 6 that these values are close to the optimal values. This suggests that $\alpha = 2^{7/4}\sigma$ is a good choice. From Fig. 6, we find that for the same setting ($\sigma = 0.05$, 16QAM), Algorithm MCT with $\alpha = 0.17$ costs much less than Algorithm CT even with the optimal α, which appears to be around 0.1.

6. SUMMARY

A modified decoding approach has been proposed to improve the efficiency of the regularization approach given in [10] for underdetermined MIMO systems. We showed how to decrease the dimension of the overdetermined ILS problem obtained by [10]. Some strategies were proposed to enhance the efficiency of our approach. We also suggested an empirical formula for the regularization parameter. Our simulations indicated that this formula gave a good approximation to the optimal regularization parameter, which is difficult to be found. Numerical results indicated that our new approach is much more efficient than the approach presented in [10] for a square constellation higher than 4QAM.
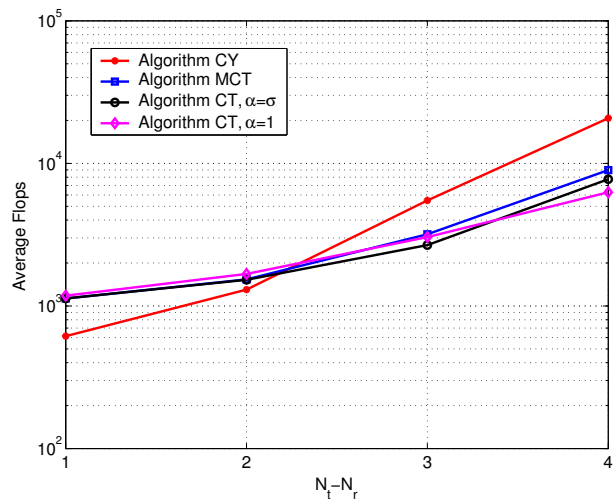
7. ACKNOWLEDGMENTS

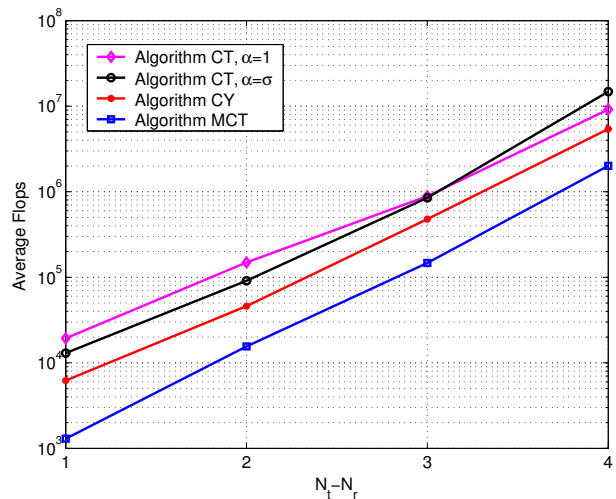Figure 1: Average flops vs $N_t - N_r$ (4QAM)

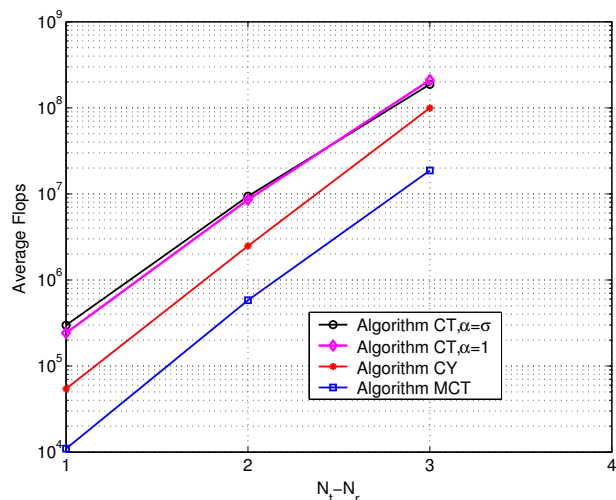Figure 2: Average flops vs $N_t - N_r$ (16QAM)

Figure 3: Average flops vs $N_t - N_r$ (64QAM)
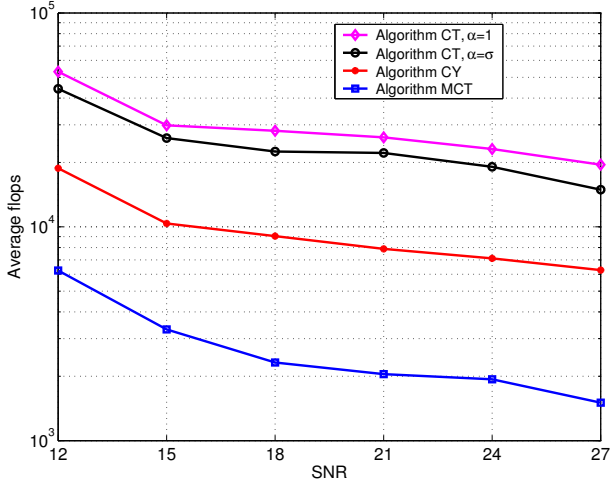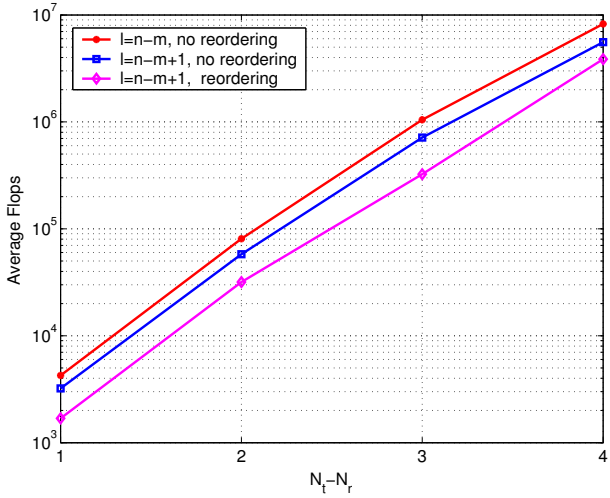
Figure 4: Average flops vs SNR (16QAM)

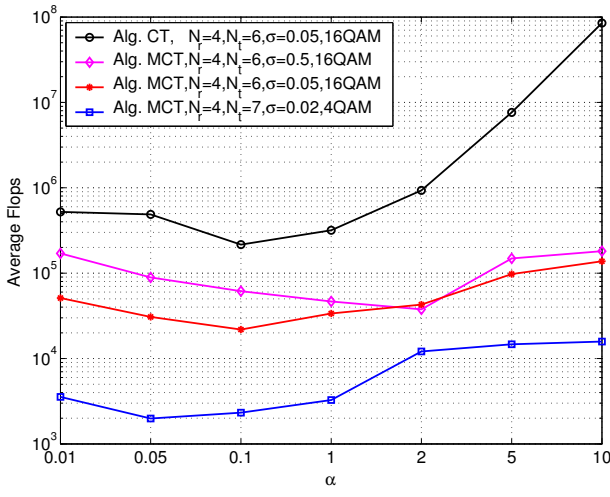**Figure 5: Average flops vs $N_t - N_r$ (16QAM)**

Figure 6: Average flops vs α

8. REFERENCES

[1] M. Damen, A. Chkeif, and J.C. Belfiore. Lattice code decoder for space-time codes. *IEEE Commun. Lett.*, 4(5):161–163, 2000.

[2] M. Damen, H. El Gamal, and G. Caire. On maximum-likelihood detection and the search for the closest lattice point. *IEEE Trans. Inf. Theory*, 49(10):2389–2402, Oct. 2003.

[3] L. Brunel and J. Boutros. Lattice decoding of joint detection in direct-sequence CDMA systems. *IEEE Trans. Info. Theory*, 49(4):1030–1037, 2003.

[4] L. Brunel. Multiuser detection techniques using maximum likelihood sphere decoding in multicarrier CDMA systems. *IEEE Trans. Wireless Commu.*, 3(3):949–957, 2004.

[5] X.-W. Chang and Q. Han. Solving box-constrained integer least squares problems. *IEEE Trans. Wireless Commun.*, to appear, 2007.

[6] M. Damen, K. Abed-Meraim, and J. Belfiore. Generalized sphere decoder for asymmetrical space-time communication architecture. *Electron. Lett.*, 36:166–167, 2000.

[7] P. Dayal and M. Varanasi. A fast generalized sphere decoder for optimum decoding of under-determined MIMO systems. In *41st Annu. Allerton Conf. Communication, Control, and Computing*, pages 1216–1225, Monticello, IL, Oct. 2003.

[8] Z. Yang, C. Liu, and J. He. A new approach for fast generalized sphere decoding in MIMO systems. *IEEE Sig. Proc. Letters*, 12(1):41–44, 2005.

[9] X.-W. Chang and X. Yang. A new fast generalized sphere decoding algorithm for underdetermined MIMO systems. In *23rd Biennial Symposium on Communications*, pages 18–21, Kingston, Ontario, Canada, 2006.

[10] T. Cui and C. Tellambura. An efficient generalized sphere decoder for rank-deficient MIMO systems. *IEEE Commun. Lett.*, 9(5):423–425, 2005.

[11] M.O. Damen, H. El Gamel, and G. Caire. MMSE-GDFE lattice decoding for solving under-determined linear systems with integer unknowns. In *Proc. IEEE Int. Symp. Inform. Theory*, Chicago, USA, June-July 2004.

[12] P. Wang and T. Le-Ngoc. A low complexity generalized sphere decoding approach for underdetermined MIMO systems. In *Proc. IEEE Int. Conf. on Communications*, pages 4266-4271, June 2006.

[13] G.H. Golub and C.F. Van Loan. *Matrix computations*. The Johns Hopkins University Press, Baltimore, Maryland, 3rd edition.

[14] T. H. Cormen, C. E. Leiserson, and C. Rivest. *Introduction to algorithms*. MIT Press, 2001.

[15] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger. Closest point search in lattices. *IEEE Trans. Inf. Theory*, 48(8):2201–2214, 2002.

[16] A. D. Murguan, H. El Gamal, M. Damen, and G. Caire. A unified framework for tree search decoding: rediscovering the sequential decoder. *IEEE Trans. Inf. Theory*, 52(3):933–953, 2006.