Approximate ML Detection for MIMO Systems Using Multistage Sphere Decoding

Tao Cui, *Student Member, IEEE,* and Chintha Tellambura, *Senior Member, IEEE*

***Abstract*—We derive a new multistage sphere decoding (MSD) algorithm, which is a generalization of the conventional sphere decoder (SD). This new MSD exploits that many higher order signal constellations can naturally be decomposed into several lower order constellations. We develop a two-stage SD for a 16-ary quadrature amplitude modulation (16QAM) multi-input multi-output (MIMO) system by decomposing 16QAM into two 4QAM constellations. The first stage generates a list of 4QAM vectors. For each of these, the second stage computes an optimal 4QAM vector. In the low signal-to-noise ratio (SNR) region, our MSD performs close to the original (single-stage) SD, but it has a lower complexity. In the high SNR region, our MSD is not suitable for reaching near maximum likelihood (ML) performance.**

***Index Terms*—MIMO, sphere decoding.**

I. Introduction

SPACE-time processing and multi-input multi-output (MIMO) systems have emerged as promising high-capacity communication techniques. The sphere decoder (SD) [1], which is a computationally efficient decoding algorithm with maximum likelihood (ML) performance, has received considerable interest. The original SD has been used to decode lattice codes [2] and space-time codes [3]. For a MIMO system with m transmit and n receive antennas $(n \geq m)$, the input and output relationship is $\mathbf{y} = \mathbf{H}\mathbf{s}^* + \mathbf{n}$, where $\mathbf{s}^*$ is the transmitted signal vector, $\mathbf{H} \in \mathcal{C}^{n \times m}$ is the channel perfectly known to the receiver $\mathbf{y} \in \mathcal{C}^n$, and $\mathbf{n}$ is the additive noise vector. The basic detection problem is the discrete least-squares optimization

$$\hat{\mathbf{s}}_{\mathrm{ml}} = \arg \min_{\mathbf{s} \in \mathcal{Q}^m} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \qquad (1)$$

where $\| \cdot \|$ denotes the Euclidean norm, $\mathcal{Q}$ is a finite complex set called the signal constellation $s_k \in \mathcal{Q}$, and $\mathbf{s} = [s_1, s_2, \ldots, s_m]^T$. Typically, the computational complexity for solving $\hat{\mathbf{s}}_{\mathrm{ml}}$ grows exponentially with m and the size of $\mathcal{Q}$. Remarkably, the SD achieves polynomial complexity in the high signal-to-noise ratio (SNR) region (however, its complexity is exponential in the low SNR region). Nevertheless, the complexity of the SD may be too high for certain applications.

Manuscript received August 3, 2004; revised October 4, 2004. This work has been supported in part by the Natural Sciences and Engineering Research Council of Canada, Informatics Circle of Research Excellence, and Alberta Ingenuity Fund. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Geert Leus.

The authors are with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 2V4 Canada (e-mail: taocui@ece.ualberta.ca; chintha@ece.ualberta.ca).

Digital Object Identifier 10.1109/LSP.2004.842263

Suboptimal detectors for (1) include the zero-forcing (ZF) detector and the vertical Bell Labs layered space time (V-BLAST) detector (nulling and cancelling) [4]. The complexity of the SD is substantially higher than that of V-BLAST in the low SNR region, but the SD performs much better than V-BLAST. Attempts have been made to develop other suboptimal algorithms that have performance/complexity advantages over SD and V-BLAST (see [5] and [6]).

Multilevel codes and multistage decoding allow the use of component codes and their sequential decoding [7]. This approach has been widely used to construct complicated codes. Each stage uses the decisions from the previous stages and decodes one component code. Multistage sequential decoding facilitates this approach since ML decoding of large codes can be prohibitively complex. To the best of our knowledge, such an approach has not been developed for uncoded MIMO systems. Spectrally efficient, higher order constellations such as 16QAM, which are popular with MIMO systems, can naturally be decomposed into several lower order constellations [8]. Thus, the uncoded MIMO detection problem can be viewed as a multistage decoding problem.

This letter develops a multistage sphere decoder using such decompositions. We explain the key idea by a specific example and let 2^q-ary quadrature amplitude modulation (QAM) constellation and all QAM $N \times 1$ vectors be $\mathcal{Q}_{2^q}$ and $\mathcal{Q}_{2^q}^N$. Specifically, we know that if $x \in \mathcal{Q}_{16}$, there exist $x_1, x_2 \in \mathcal{Q}_4$ such that $x = \sqrt{2}x_1 + \sqrt{2}/2x_2$. This means a vector $\mathbf{s}$ in $\mathcal{Q}_{16}^m$ can be uniquely mapped into two component vectors $\mathbf{s}_1$ and $\mathbf{s}_2$ in $\mathcal{Q}_4^m$. We can then envision two-stage decoding in which $\mathbf{s}_1$ and $\mathbf{s}_2$ are detected sequentially by two SDs. The second SD uses the detected value $\hat{\mathbf{s}}_1$ for interference cancellation. However, due to error propagation, passing hard decisions between stages can result in poor performance. To overcome this, we use a list SD (LSD) [9] to generate a list of candidates $\hat{\mathbf{s}}_1 \in \mathcal{Q}_4^m$ in the first stage. For each candidate, a second SD computes the optimal candidate in $\mathcal{Q}_4^m$ (see Fig. 1). Depending on the list size, our multistage sphere decoding (MSD) performs close to the original single-stage SD (SSD) but reduces computational complexity. It can readily be generalized to other constellations. However, its complexity needed to achieve near ML performance increases with increasing SNR.

Notation: Bold symbols denote matrices or vectors. $(\cdot)^T$ and $(\cdot)^H$ denote transpose and conjugate transpose. The set of all complex $K \times 1$ vectors is denoted by $\mathcal{C}^K$ and $j = \sqrt{-1}$. We use the terms SSD(M, m) and LSD(M, m) to denote a conventional, SSD, and a list SD over the $\mathcal{Q}_M^m$ space. For $\mathbf{u} \in \mathcal{Q}_{16}^m$, we write $\mathbf{u} = \sqrt{2}\mathbf{u}_1 + \sqrt{2}/2\mathbf{u}_2$, where $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{Q}_4^m$.

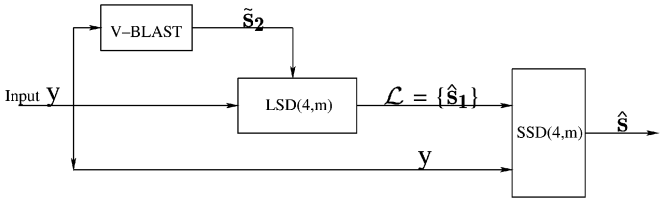

Fig. 1. Block diagram of the MSD for 16QAM systems.

II. MSD Algorithm

By transforming (1), the SSD of Fincke and Phost [1] solves the equivalent problem

$$\hat{\mathbf{s}}_{\mathrm{ml}} = \arg \min_{\mathbf{s} \in \mathcal{Q}^m} \|\mathbf{y}' - \mathbf{R}\mathbf{s}\|^2 \tag{2}$$

where $\mathbf{y}' = \mathbf{Q}_1^H \mathbf{y}$, the $m \times m$ upper triangular matrix $\mathbf{R}$, and the $n \times n$ orthogonal matrix $\mathbf{Q} = [\mathbf{Q}_1, \mathbf{Q}_2]$ are the QR factorization of $\mathbf{H}$. The matrices $\mathbf{Q}_1$ and $\mathbf{Q}_2$ represent the first m and last $n - m$ orthonormal columns of $\mathbf{Q}$. The SSD computes all s_ks that lie within a sphere of the given radius.

For brevity, we only show how to apply the MSD to 16QAM, and a more general algorithm is given later. An arbitrary 16QAM vector $\mathbf{s}$ can be uniquely expressed as $\mathbf{s} = \sqrt{2}\mathbf{s}_1 + \sqrt{2}/2\mathbf{s}_2$, where $\mathbf{s}_1, \mathbf{s}_2 \in \mathcal{Q}_4^m$. Similarly, let the true transmit vector be $\mathbf{s}^* = \sqrt{2}\mathbf{s}_1^* + \sqrt{2}/2\mathbf{s}_2^*$. The problem of detecting $\hat{\mathbf{s}}_{\mathrm{ml}}$ (2) is equivalent to detecting two 4QAM component vectors as follows:

$$[\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2] = \arg \min_{\mathbf{s}_1, \mathbf{s}_2 \in \mathcal{Q}_4^m} \left\| \mathbf{y}' - \mathbf{R}\left(\sqrt{2}\mathbf{s}_1 + \frac{\sqrt{2}}{2}\mathbf{s}_2\right) \right\|^2. \tag{3}$$

To begin, we need an initial approximation to the true signal $\mathbf{s}^*$. Let this be $\tilde{\mathbf{s}} = \sqrt{2}\tilde{\mathbf{s}}_1 + \sqrt{2}/2\tilde{\mathbf{s}}_2$. Using this, we do a partial interference cancellation as $\mathbf{y}_2 = \mathbf{y}' - \sqrt{2}/2\mathbf{R}\tilde{\mathbf{s}}_2$ in the first stage[1]. If $\mathbf{s}_2 = \mathbf{s}_2^*$, $\mathbf{y}_2$ is clearly sufficient to detect $\mathbf{s}_1^*$. We search $\mathbf{s}_1$ that minimizes

$$\|\mathbf{y}_2 - \sqrt{2}\mathbf{R}\mathbf{s}_1\|^2. \tag{4}$$

However, $\tilde{\mathbf{s}}_2 \neq \mathbf{s}_2^*$ in general, and minimizing (4) will likely give a wrong estimate. Therefore, we use an LSD [9] to generate a list $\mathcal{L}$ of the N_{cand} candidates $\hat{\mathbf{s}}_1$ that make (4) the smallest. The list size is between 4^m and 1 and is proportional to the probability that the true solution $\mathbf{s}_1^*$ falls in the list. With a properly chosen radius r, we can obtain $\mathcal{L}$ with N_{cand} candidates on average. To obtain a typical value of r, we note that for true $\mathbf{s}_1^*$

$$\|\mathbf{y}_2 - \sqrt{2}\mathbf{R}\mathbf{s}_1^*\|^2 = \left\| \frac{\sqrt{2}}{2}\mathbf{R}(\mathbf{s}_2^* - \tilde{\mathbf{s}}_2) + \mathbf{n} \right\|^2 \tag{5}$$

where $\mathbf{n}$ is the additive Gaussian noise vector with variance σ_n^2. Since $\tilde{\mathbf{s}}_2$ is correlated with $\mathbf{R}$ and $\mathbf{n}$, (5) cannot be treated as a chi-square random variable with $2m$ degrees of freedom. The expected value of this random variable is denoted by E, which can be obtained via simulation. As in [9], one possible choice of radius is $r^2 = kE$, where k is chosen so that the average length of the list is N_{cand}. For typical values of σ_n^2 and σ_h^2, r^2 corresponding to N_{cand} can be obtained from simulation and can be stored in memory for practical use.

In the second stage, for each candidate $\hat{\mathbf{s}}_1 \in \mathcal{L}$, the SSD (4, m) solves

$$\hat{\mathbf{s}}_2 = \arg \min_{\mathbf{s}_2 \in \mathcal{Q}_4^m} \left\| \mathbf{y}_1 - \frac{\sqrt{2}}{2}\mathbf{R}\mathbf{s}_2 \right\|^2 \tag{6}$$

where $\mathbf{y}_1 = \mathbf{y}' - \sqrt{2}\mathbf{R}\hat{\mathbf{s}}_1$. This process provides N_{cand} pairs of $[\hat{\mathbf{s}}_1, \hat{\mathbf{s}}_2]$, and the best among them is selected as the output. Each time a $\hat{\mathbf{s}}_2$ for (6) is found with a $\hat{\mathbf{s}}_1$, the search radius of the second stage is updated if $\|\mathbf{y}_1 - \sqrt{2}/2\mathbf{R}\mathbf{s}_2\|$ is less than the current radius.

Remarks:

- Our proposed MSD consists of an LSD (4, m) and an SSD (4, m), which computes N_{cand} points of 4QAM vector pairs. Is our approach less complex than an SSD (16, m)? That depends on the difference of the complexities of SSD (4, m) and SSD (16, m). For example, in the high SNR region, the difference of complexity is small, and our proposed MSD is not more efficient than the SSD(16, M), while in the low SNR region, the difference of complexity can be large, and our proposed algorithm is much more efficient.
- The parameter N_{cand} gives a tradeoff between complexity and performance. When N_{cand} is small, the probability that $\mathbf{s}_1^* \in \mathcal{L}$ is low and the bit error rate (BER) will increase, while the complexity will decrease.
- The initial estimate $\tilde{\mathbf{s}}_2$ can be obtained via ZF, minimum mean-square error (MMSE), or V-BLAST.
- The performance of our MSD may be further improved by using $\hat{\mathbf{s}}_2$ from the second stage SD to recompute the first stage output $\hat{\mathbf{s}}_1$. This will result in an iterative MSD and will increase computational complexity.

Our proposed MSD can be generalized to other constellations. For example, a 64QAM vector $\mathbf{s}$ can be uniquely expressed as $\mathbf{s} = 2\sqrt{2}\mathbf{s}_1 + \sqrt{2}\mathbf{s}_2 + \sqrt{2}/2\mathbf{s}_3$, where $\mathbf{s}_i \in \mathcal{Q}_4^m$, $i = 1, 2, 3$. Therefore, the MSD will have three stages. In the first stage, an LSD is used to generate a list of $\mathbf{s}_1$. Another LSD is used to generate a list of $\mathbf{s}_2$ in the second stage. In the last stage, an SSD is used to obtain the solution. The 4^q-QAM constellation can be represented as a weighted sum of q Quadrature Phase Shift Keying (QPSK) constellations [8]. That is, for $s \in M$-QAM and $s_i \in$ QPSK, $0 \leq i < q$, we have

$$s = \sum_{i=0}^{q-1} 2^i \left(\frac{\sqrt{2}}{2}\right) s_i. \tag{7}$$

Hence, the MSD for a 4^q-QAM system has q stages. For 2^q-PSK, the MSD has q stages and consists of $q - 1$ LSDs. For brevity, we do not give the algorithm in detail.

The complexity of a q-stage MSD $C_{\mathrm{MSD}} < (\prod_{i=1}^{q-1} N_{\mathrm{cand},i} + q - 1)C_{\mathrm{SD}}$, where $N_{\mathrm{cand},i}$ is the list size of the ith stage, and C_{SD} is the complexity of the SSD with 4QAM. Simulation results

[1]Note that we cancel $\tilde{\mathbf{s}}_2$ first since any errors in $\tilde{\mathbf{s}}_2$ will be attenuated by $\sqrt{2}$ [see (3)], whereas any errors in $\tilde{\mathbf{s}}_1$ will be magnified by $\sqrt{2}$.

show that the MSD achieves increased complexity savings for large constellations.

The MSD algorithm for 4^q-QAM can be summarized by the following steps.

Step 1) Compute the ZF solution $\tilde{\mathbf{s}}$ and decompose it to $\tilde{\mathbf{s}} = \sum_{i=1}^{q} \alpha_i \tilde{\mathbf{s}}_i$, where $\tilde{\mathbf{s}}_i \in \mathcal{Q}_4^m$.

Step 2) For $k = 1, \ldots, q-1, \mathbf{y}_k = \mathbf{y}' - \sum_{i \neq k} \alpha_i \mathbf{R}\tilde{\mathbf{s}}_i$. Solve $||\mathbf{y}_k - \alpha_k \mathbf{R}\hat{\mathbf{s}}_k||^2 < r^2$ with an LSD and insert each $\hat{\mathbf{s}}_k$ into a list $\mathcal{L}_k$.

Step 3) For each $(q-1)$ candidates $[\hat{\mathbf{s}}_1, \ldots, \hat{\mathbf{s}}_{q-1}]$, solve

$$\hat{\mathbf{s}}_q = \arg \min_{\mathbf{s}_q \in \mathcal{Q}_4^m} ||\mathbf{y}_q - \alpha_q \mathbf{R}\mathbf{s}_q||^2 \tag{8}$$

with an SSD, where $\mathbf{y}_q = \mathbf{y}' - \sum_{i=1}^{q-1} \alpha_i \mathbf{R}\hat{\mathbf{s}}_i$.

Step 4) Find the best q-tuple $[\hat{\mathbf{s}}_1, \ldots, \hat{\mathbf{s}}_q]$ and output $\hat{\mathbf{s}} = \sum_{i=1}^{q} \alpha_i \hat{\mathbf{s}}_i$.

III. Simulation Results

We now compare the MSD with the SSD for a 16QAM, uncoded MIMO system with four transmit and four receive antennas over a flat Rayleigh fading channel. This system is simulated using MATLAB V5.3. The MATLAB command "flops" is used to count the number of flops. Only the flops of the search algorithm are counted, without accounting for the preprocessing stage. The initial radius r is chosen according to the noise variance. Both the SSD and the MSD use the Schnorr–Euchner variant of SD [10]. The initial detection uses the ZF-VBLAST.

Fig. 2 compares the BER of the SSD with that of the MSD as a function of the number of candidates in the first stage N_{cand}. As N_{cand} increases, the MSD performs close to the SSD. As N_{cand} varies, its performance varies between those of V-BLAST and SSD. The complexity of the MSD increases as N_{cand} increases (see Fig. 3), and it is lower than that of the SSDs when the SNR is below a threshold. For instance, when $N_{\text{cand}} = 10$, this complexity crossover point is 17 dB. Fig. 3 also shows that the complexity of the MSD is almost constant with specific N_{cand}, suggesting that its complexity is polynomial for the whole SNR range, in contrast to the conventional SD. The major drawback of our MSD is that the complexity needed to achieve near-ML performance increases with increasing SNR. Thus, our MSD is suitable for the low SNR region, where it can be combined with an outer code to achieve low BER.

IV. Conclusion

In this letter, we developed a multistage SD by decomposing a large constellation into a sum of smaller constellations. This is particularly easy for QAM where a higher order QAM constellation can be readily resolved into lower order QAM components. A series of conventional LSDs and SSDs are used to search over the smaller constellation spaces. As a specific example, 16QAM is resolved into two 4QAM constellations. An LSD and an SSD are used to search over the 4QAM constellations. Simulation results show that in the low SNR region, our MSD performs close to the SSD and reduces complexity. On the other hand, the conventional SD is still a better choice in the high SNR region.

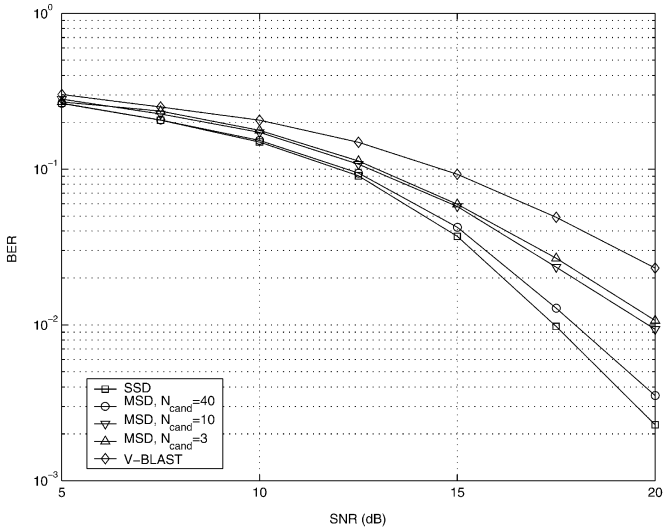

Fig. 2. BER comparison with different N_{cand} for a 16QAM MIMO system with four transmit and four receive antennas.

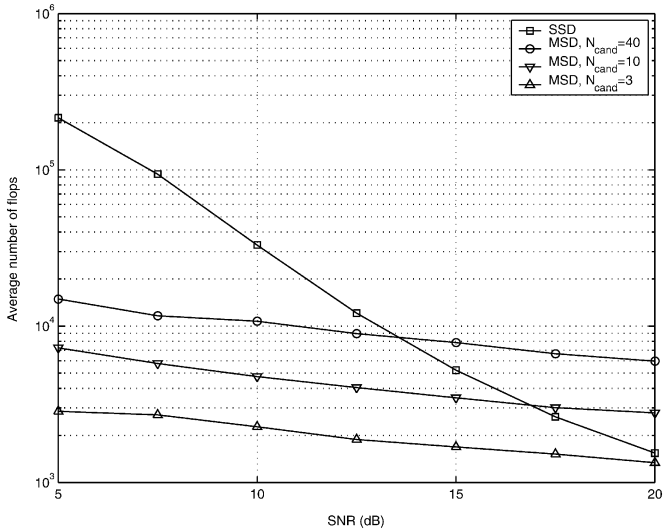

Fig. 3. Complexity comparison with different N_{cand} for a 16QAM MIMO system with four transmit and four receive antennas.

Acknowledgment

The authors would like to thank the anonymous reviewers for their critical comments that greatly improve this letter.

References

[1] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, inlcuding a complexity analysis," *Math. Computat.*, vol. 44, pp. 463–471, Apr. 1985.

[2] E. Viterbo and E. Biglieri, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 1639–1642, Jul. 1999.

[3] O. Damen, A. Chkeif, and J.-C. Belfiore, "Lattice code decoder for space-time codes," *IEEE Commun. Lett.*, vol. 4, no. 5, pp. 161–163, May 2000.

[4] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using the V-BLAST space-time communication architecture," *Electron. Lett.*, vol. 35, no. 1, pp. 14–15, Jan. 1999.

[5] M. Rupp, G. Gritsch, and H. Weinrichter, "Approximate ML detection for MIMO systems with very low complexity," in *Proc. IEEE Int. Conf. Speech, Acoust., Signal Process.*, May 2004.

[6] H. Artes, D. Seethaler, and F. Hlawatsch, "Efficient detection algorithms for MIMO channels: A geometrical approach to approximate ML detection," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2808–2820, Nov. 2003.
[7] A. R. Calderbank, "Multilevel codes and multistage decoding," *IEEE Trans. Commun.*, vol. 37, no. 3, pp. 222–229, Mar. 1989.
[8] B. Tarokh and H. Sadjadpour, "Construction of OFDM M-QAM sequences with low peak-to-average power ratio," *IEEE Trans. Commun.*, vol. 51, no. 1, pp. 25–28, Jan. 2003.
[9] B. Hochwald and S. T. Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.
[10] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Program.*, vol. 66, pp. 181–191, 1994.