

# Selection of Design Parameters for Generalized Sphere Decoding Algorithms

Ping WANG, Tho LE-NGOC

*Department of ECE, McGill University, 3480 University, Montréal, Canada*

*Email: pingwang.mcgill@gmail.com, tho.le-ngoc@mcgill.ca*

*Received November 11, 2009; revised December 15, 2009; accepted January 7, 2010*

## Abstract

Various efficient generalized sphere decoding (GSD) algorithms have been proposed to approach optimal ML performance for underdetermined linear systems, by transforming the original problem into the full-column-rank one so that standard SD can be fully applied. However, their design parameters are heuristically set based on observation or the possibility of an ill-conditioned transformed matrix can affect their searching efficiency. This paper presents a better transformation to alleviate the ill-conditioned structure and provides a systematic approach to select design parameters for various GSD algorithms in order to high efficiency. Simulation results on the searching performance confirm that the proposed techniques can provide significant improvement.

**Keywords:** Sphere Decoding (SD), Generalized Sphere Decoding (GSD), Maximum-likelihood (ML), Integer Least-Square (ILS), MIMO,  $\lambda$ -GSD, Multi-User Detection (MUD), CDMA, MC-CDMA

## 1. Introduction

Sphere decoding (SD) is an efficient searching method to obtain maximum-likelihood (ML) solution for NP-hard integer least-square (ILS) problems. SD can have polynomial time average complexity for many practical communication problems [1]. It offers large complexity reduction over the exhaustive search method in numerous applications, e.g., lattice decoding [2], multi-input, multi-output (MIMO) detection [3] and multi-user detection (MUD) [4]. However, when the problem is underdetermined, zero elements appear in the diagonal terms of the upper-triangular matrix generated by QR or Cholesky decomposition before searching, and the standard SD searching cannot apply. Such underdetermined ILS problems arise in many areas, e.g., MIMO detection with the number of transmit antennas larger than that of receiver antennas; MIMO detection with strongly correlated channel gains [5] or MUD for overloaded CDMA-related systems [8]. To solve such problems efficiently, generalized SD (GSD) algorithms, fully or partly based on SD, have been developed [6–12] for underdetermined or rank-deficient MIMO systems.

The GSD algorithms in [6,7] modify the underdetermined problem for constant-modulus constellation (e.g., QPSK) into a full-column-rank one by introducing a design parameter purely based on observations and then

uses SD on the modified problem. For non-constant-modulus ones, e.g., 16/64QAM, they have to be transformed into multiple QPSKs, leading to larger dimension and hence increased complexity. To avoid this problem and obtain better efficiency, the  $\lambda$ -GSD algorithm proposed in [8], performs transformation without expanding the size of M-ary QAMs. Unlike the GSD algorithms in [6,7], the design parameter can be is upper-bounded [8,13] to guarantee near-ML performance for high QAMs and has little effect on the efficiency of  $\lambda$ -GSD [14].

In this paper, we first present an improved version of the  $\lambda$ -GSD algorithm to alleviate the ill-conditioned problem in the transformed matrix of [8]. Then we study the setting of the design parameter for the improved version of  $\lambda$ -GSD, as well as GSD in [6,7]. A more systematic approach is proposed to select design parameters for the above GSD algorithms to achieve high efficiency.

The remainder of the paper is as follows. Section 2 summarizes transformations employed in several popular GSD algorithms. Section 3 presents the improved version of  $\lambda$ -GSD over [8] and briefly discusses its performance characteristics; then we proceed to propose a systematic approach to choose the design parameter for it and also for that of [6,7] in Section 4. Section 5 provides illustrative results under flat-fading underdetermined MIMO scenarios. Section 6 concludes the paper.

## 2. Transformations by GSD Algorithms

The objective of ILS problem is to find the least-square (LS) solution for a linear system with integer unknowns. When the transmitted vectors (the unknowns) are from a finite set  $A$  in communications, i.e.,  $\mathbf{x} \in A \subset \mathbb{Z}^n$ , where  $\mathbb{Z}^n$  denotes the  $n$ -dim vector space with integer entries, the objective function is

$$\min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2, \text{ for } \mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}, \quad (1)$$

where  $\mathbf{y} \in \mathfrak{R}^m$  is a  $m \times 1$  vector for a real-valued system<sup>1</sup>,  $\mathbf{H} \in \mathfrak{R}^{m \times n}$  the system matrix typically with full column rank (i.e.,  $\text{rank}(\mathbf{H}) = n$ ) and  $\mathbf{v}$  a  $m \times 1$  zero-mean Gaussian noise vector,  $\mathbf{v} \sim N(0, \sigma^2 \mathbf{I})$ . It's well known that SD algorithm and its variants are efficient approaches to solve such problems: it reduces the complexity of exhaustive-search greatly by conducting search within a hyper-sphere but still guarantees the ML optimality. One advantage of SD algorithms is that the search can be divided into layered enumeration of integer values in *one-dimensional* interval, so that it can efficiently decide which points are inside the hyper-sphere. However, SD algorithm was originally designed for full-column-rank system only. For underdetermined ILS problem when  $\mathbf{H} \in \mathfrak{R}^{m \times n}$  has  $\text{rank}(\mathbf{H}) = m, m < n$  in (1), the original SD algorithms can not proceed efficiently, as on certain searching layers, the candidate point set is no longer in a 1D-interval. [6–8] introduce transformations for underdetermined problem so that SD can be utilized on the transformed full-column-rank system. The transformations used are as follows.

The original underdetermined problem for constant-modulus constellation (e.g., QPSK) was modified into a full-column-rank system [6] and the equivalent new objective function is

$$\min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left( \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda^2 \|\mathbf{x}\|_2^2 \right), \quad (2)$$

where  $\lambda \neq 0$  is a design parameter. (2) is equivalent to

$$\min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} \mathbf{H} \\ \lambda \mathbf{I}_{n \times n} \end{pmatrix} \mathbf{x} \right\|_2^2, \quad (3)$$

where  $\begin{pmatrix} \mathbf{H} \\ \lambda \mathbf{I}_{n \times n} \end{pmatrix}$  in (3) is full-column-ranked with  $\lambda \neq 0$ . If  $\mathbf{x}$  is from constant-modulus constellations,  $\|\mathbf{x}\|_2^2$  is constant and (3) is the same as (1); and when applying SD on the new problem (3), ML performance can still be achieved. For high QAMs, in order to guarantee ML performance, high QAMs are transformed into multiple QPSKs in [6] before using the above transfor-

mation, e.g., 16QAM vector  $\mathbf{x}$  can be represented as  $\left[ \sqrt{2}x_1 \sqrt{2}x_2 / 2 \right]$  where  $x_1, x_2$  are from QPSK. The 16QAM system is transformed into a QPSK one with

$$\mathbf{y} = \left[ \sqrt{2} \mathbf{H} \quad \frac{\sqrt{2}}{2} \mathbf{H} \right] \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} + \mathbf{v}. \quad (4)$$

Hence the transformation in (3) can be used on (4) and SD can then be applied efficiently. Despite the fact that the complexity of the GSD algorithm in [6] depends on  $\lambda$ , the parameter is set arbitrarily in [6].

The transformation of high QAMs into multiple QPSK ones in [6] unavoidably lead to larger problem size for the ILS problem and increased search complexity. To avoid this problem, another transformation is utilized in [8] for both the constant-modulus and non-constant modulus constellations, i.e.,

$$\min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left( \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda^2 \|\bar{\mathbf{x}}_2\|_2^2 \right) = \min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ \mathbf{0} & \lambda \mathbf{I}_{(n-m) \times (n-m)} \end{pmatrix} \begin{bmatrix} \bar{\mathbf{x}}_1 \\ \bar{\mathbf{x}}_2 \end{bmatrix} \right\|_2^2 \quad (5)$$

where  $\bar{\mathbf{x}}_2$  is subset of  $\mathbf{x} = [\bar{\mathbf{x}}_1 \bar{\mathbf{x}}_2]$  with length  $(n-m)$ . SD searching is applied on (5) directly for both QPSK and high QAMs. Consequently, exact ML performance can be achieved for QPSK constellation with arbitrary value of  $\lambda$  whilst close-to-ML performance can be achieved with sufficient small  $\lambda$  [8]. Moreover, an upper bound is provided in [8] for  $\lambda$  in high QAMs to approach ML performance with negligible loss and it's also shown that the efficiency is quite insensitive to  $\lambda$  [14] for  $\lambda$ -GSD, unlike GSD in [6]. Therefore, the choice of  $\lambda$  can be randomly set for QPSK and judiciously set to be within the upper bound for high QAMs.

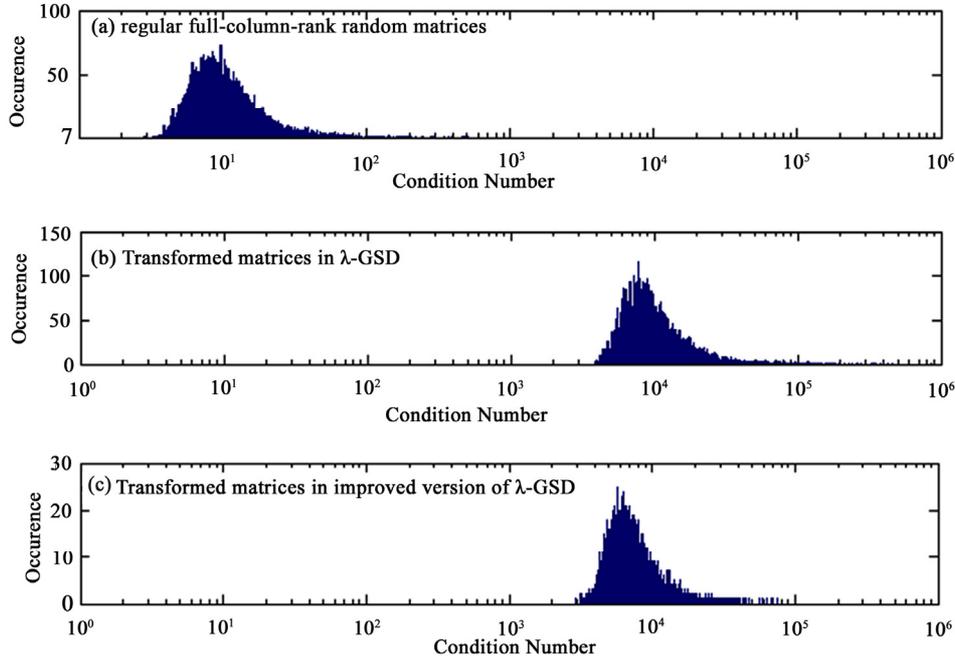
Later in [7], by noticing the structure of the transformation in (5) is less efficient<sup>2</sup> and also targeting to alleviate the size expansion problem in [6,7] considers to utilize a transformation of adding the  $\bar{\mathbf{x}}_2$  part of  $\mathbf{x} = [\bar{\mathbf{x}}_1 \bar{\mathbf{x}}_2]$  with length  $(n-m+1)$  instead of  $(n-m)$ , and then apply SD to achieve ML performance for constant-modulus constellations. The resulted new ILS problem is

$$\min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left( \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda^2 \|\bar{\mathbf{x}}_2\|_2^2 \right) = \min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ \mathbf{0} & \lambda \mathbf{I}_{(n-m+1) \times (n-m+1)} \end{pmatrix} \begin{bmatrix} \bar{\mathbf{x}}_1 \\ \bar{\mathbf{x}}_2 \end{bmatrix} \right\|_2^2 \quad (6)$$

For non-constant-modulus constellations, [7] adopts the same strategy as [6], i.e., to transform high QAMs into multiple QPSKs in order to achieve exact-ML performance in a similar way as [6,7] also suggests a value

<sup>1</sup>For complex-valued communication systems, linear transformations are generally used to obtain real-valued ones and apply SD.

<sup>2</sup>to be discussed in Section 3.



**Figure 1. Histogram of condition number distribution of MIMO system matrices over  $10^4$  runs @12dB, QPSK. (a) 12x12 real matrices of regular i.i.d. 6x6 complex-valued matrices; (b) transformed 12x12 real matrices of  $\lambda$ -GSD at  $\lambda=10^{-3}$ ; (c) transformed 14x12 real matrices of improved  $\lambda$ -GSD at  $\lambda=10^{-3}$ .**

of  $\lambda$  purely based on experimental observations though apparently the efficiency depends on  $\lambda$ .

### 3. Improved Transformation Structure to Speed up $\lambda$ -GSD Algorithm

For a negligible performance loss,  $\lambda$ -GSD keeps the problem size intact for high QAMs as mentioned, and hence it is more efficient than [6,7] for high QAMs. However, despite of the high efficiency of  $\lambda$ -GSD compared to its counterparts, the efficiency of  $\lambda$ -GSD is still significantly higher than SD on regular full-column-rank system. Ill-conditioned structure of the transformed matrix of  $\lambda$ -GSD partly accounts for the inefficiency. It is known that well-conditioned matrix generally leads to more efficient searching than ill-conditioned ones [16]. Figure 1 illustrates histograms for the occurrence of the condition number of the three categories of matrices over  $10^4$  runs for a QPSK MIMO system at 12dB, where (a) is for the 12x12 real-valued matrices of the i.i.d. regular 6x6 complex-valued system with full-column-rank; (b) is for the transformed 12x12 real matrices of  $\lambda$ -GSD in a 4x6 complex-valued system. (c) is for an improved version of  $\lambda$ -GSD to be proposed in this section. Comparing Figure 1(a) and (b), we can see that over  $10^4$  runs, the condition number of transformed matrices of  $\lambda$ -GSD is centered at around  $8 \times 10^3$  whilst that of the regular full-column-rank ones is only centered at around 10. Next we will propose an improved version of  $\lambda$ -GSD,

which combines the transformation in [7] and the idea of keeping problem size intact of  $\lambda$ -GSD. The modified version of  $\lambda$ -GSD to alleviate the ill-conditioned problem can be presented as follows.

Consider a real-valued underdetermined system  $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$  in (1) where  $2 < \text{rank}(\mathbf{H}) = m < n$ . Partition  $\mathbf{H} = [\bar{\mathbf{H}}_1 \ \bar{\mathbf{H}}_2]$  where  $\bar{\mathbf{H}}_1, \bar{\mathbf{H}}_2$  are  $m \times (m-2)$  and  $m \times (n-m+2)$  matrices respectively, instead of  $m \times m$  and  $m \times (n-m)$  in the original  $\lambda$ -GSD (5). Then partition  $\mathbf{x} = [\bar{\mathbf{x}}_1^T \ \bar{\mathbf{x}}_2^T]^T$  so that  $\mathbf{y} = [\bar{\mathbf{H}}_1 \ \bar{\mathbf{H}}_2][\bar{\mathbf{x}}_1^T \ \bar{\mathbf{x}}_2^T]^T + \mathbf{v}$ . Adding an equation  $\mathbf{0} = \lambda \bar{\mathbf{x}}_2 - \lambda \bar{\mathbf{x}}_2$  where  $\mathbf{0}$  is a zero-vector with length  $(n-m+2)$  and  $\lambda > 0$  is a weighting factor, the transformed new ILS problem becomes,

$$\begin{aligned} \min_{\mathbf{x} \in A \subset \mathbb{Z}^n} & \left\| \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} - \begin{bmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ 0 & \lambda \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{x}}_1 \\ \bar{\mathbf{x}}_2 \end{bmatrix} \right\|_2^2 \\ & = \min_{\mathbf{x} \in A \subset \mathbb{Z}^n} \left( \|\mathbf{y} - \mathbf{H}\mathbf{x}\|_2^2 + \lambda^2 \|\bar{\mathbf{x}}_2\|_2^2 \right) \end{aligned} \quad (7)$$

Again, SD algorithms then can be used on (7) to solve the transformed full-column-rank ILS problem. Note that the transformation in (7) appears very analogous<sup>3</sup> to that

<sup>3</sup>The only difference in transformation is the length of  $(n-m+2)$  in  $\bar{\mathbf{x}}_2$  is used in improved version of  $\lambda$ -GSD, instead of  $(n-m+1)$  in [7]. Using  $(n-m+2)$  in real-valued ILS problem is actually equivalent to using  $((n-m)/2+1)$  in corresponding complex-valued problem; i.e., “+1” is assumed to operate in complex-valued domain, whilst [7] applies “+1” directly on real-valued domain.

used in [2], i.e., (6), yet they are distinct for the case of high QAMs: the improved version of  $\lambda$ -GSD keeps the problem size intact for high QAMs and hence SD algorithm is directly applied on (7) without further transformation. Performance of the modified approach can be analyzed following the analysis for  $\lambda$ -GSD in [13], and the conclusions are similar. Specifically, for constant-modulus modulations, the modified GSD can achieve exact ML performance and for high QAMs, the choice of  $\lambda$  has to be sufficiently small to keep the performance close to optimal. Here we omit the detailed upper bound analysis for simplicity and only provide the conclusions for a  $R_x \times T_x$  ( $R_x < T_x$ ) complex linear system using uncoded square  $M$ -QAM with average symbol energy  $E_s$ : Approximately, assuming  $\varepsilon$  denotes the difference in the upper bound error probability between  $\lambda$ -GSD and ML performance, the upper bound for  $\lambda$  is  $\lambda \leq (32\varepsilon)^{1/4} [3(T_x - R_x + 1)E_s / \sigma^2]^{-1/2}$ .

Compared to original  $\lambda$ -GSD, we expect the improved version is more efficient as it has better transformed matrix structure, which can be seen from the illustrative histogram of condition number in Figure 1(c) for the transformed  $14 \times 12$  real-valued matrices in the improved version of  $\lambda$ -GSD over  $10^4$  runs for a QPSK MIMO system. We can see that the condition number in this example is centered at around  $6 \times 10^3$ . Compared to  $8 \times 10^3$  in Figure 1(b), the ill-conditioned situation gets alleviated. The complexity efficiency of the improved version of  $\lambda$ -GSD will be illustrated in Section 5 via simulations.

#### 4. Selection of Design Parameters for the GSD Algorithms

It was mentioned in [6,7] that there may exist an optimal value for the design parameter in their GSD algorithms. It is very hard, if not impossible, to derive a closed-form expression for the optimal value in a strict sense. In this section, alternatively, we propose one criterion to choose design parameters for these GSD algorithms, which enables the underlying algorithms with high efficiency. The proposed criterion also works for the improved version of  $\lambda$ -GSD algorithm in QPSK systems, since the underlying transformation shares similarity with that for GSD in [7]. Therefore, we mainly derive the design parameter for the improved version of  $\lambda$ -GSD in QPSK systems for simplicity, which can be applied to GSD in [7] as well. Note that for high QAMs, the selection of the design parameter for the improved version of  $\lambda$ -GSD is limited by the upper bound as mentioned in previous session.

For SD searching, if the first point found is closer to ML point, the search tends to be faster [15]. For the improved version of  $\lambda$ -GSD and GSD in [6,7], the first point obtained in search is different for various values of  $\lambda$ . Let us consider the transformation (5) of GSD [6] first.

Since the first point using Shnorr-Euchner enumeration<sup>4</sup> is the midpoint, hence, the elements of the first point can be written as [15]

$$x_{init\_k} = \left[ (\bar{y}_k - \sum_{i=k+1}^n R_{k,i} x_{init\_i}) / R_{k,k} \right], \text{ for } k = n, \dots, 1, \quad (8)$$

where  $x_{init\_k}$  is the ZF-DFE point [15] for the transformed ILS problem;  $\bar{y}_k = (\mathbf{Q}^T \tilde{\mathbf{y}})_k$  where  $(\mathbf{Q}, \mathbf{R})$  are the resulted matrices after QR decomposition of the transformed matrix;  $R_{i,j}$  is the  $(i,j)$ -th element of  $\mathbf{R}$  and  $\tilde{\mathbf{y}}$  is the equivalent received vector for the transformed ILS problem. Then at  $k=n$ , based on (5), we have

$$\begin{aligned} x_{init\_n} &= \left[ \left( \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ \lambda \mathbf{I}_{n \times n} \end{pmatrix}^T \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ \lambda \mathbf{I}_{n \times n} \end{pmatrix} \right)^{-1} \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ \lambda \mathbf{I}_{n \times n} \end{pmatrix}^T \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \right]_n \\ &= \left[ \left( \mathbf{H}^T \mathbf{H} + \lambda^2 \mathbf{I}_{n \times n} \right)^{-1} \mathbf{H}^T \mathbf{y} \right]_n \end{aligned} \quad (9)$$

Clearly,  $x_{init\_n}$  depends on  $\lambda$ . From (8) and (9), we

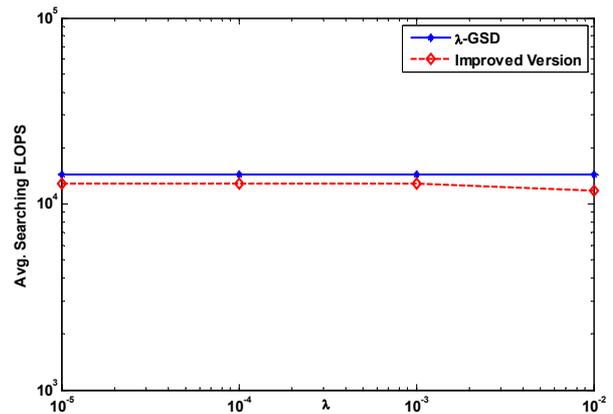


Figure 2. Avg. searching FLOPS vs.  $\lambda$  @ 2x4 MIMO, 16QAM, 28dB,  $10^4$  runs.

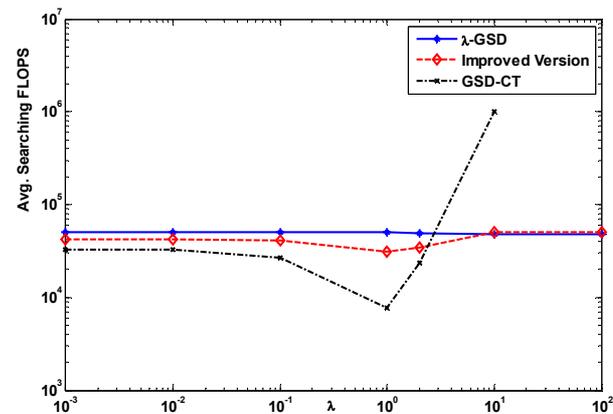


Figure 3. Avg. searching FLOPS vs.  $\lambda$  @ 4x8 MIMO, QPSK, 12dB,  $10^4$  runs.

know that when  $\lambda$  is 0, the first element of (9),  $x_{init\_n}$ , is the first element of ZF-DFE point of the original ILS, i.e.,  $x_{init\_n} = [x_{zf-dfe}]_n$ ; Thus, the first point,  $\mathbf{x}_{init}$ , is its ZF-DFE point. On the other hand, when  $\lambda$  is equal to  $1/\sqrt{\gamma_s}$ , where  $\gamma_s$  represents the SNR ratio, (9) becomes  $x_{init\_n} = \left[ \left( \mathbf{H}^T \mathbf{H} + \mathbf{I}_{n \times n} / \gamma_s \right)^{-1} \mathbf{H}^T \mathbf{y} \right]_n$  which is the first element of MMSE-DFE point for the original ILS, i.e.,  $x_{init\_n} = [x_{mmse-dfe}]_n$  and  $\mathbf{x}_{init}$  is the MMSE-DFE point. It's known that MMSE detector is the best linear estimator to the ILS problem based on mean-square error criterion and MMSE-DFE is generally closer to the optimal ML point than ZF-DFE; Using MMSE-DFE point as the first point in the searching procedure could lead to a fast convergence [15]. Therefore, to achieve high efficiency, we suggest using  $\lambda = 1/\sqrt{\gamma_s}$  in GSD [6], so that the first valid point is the MMSE-DFE point of the original ILS.

Next, we continue to derive the suggested value of  $\lambda$  for the improved version of  $\lambda$ -GSD for QPSK system as follows. From (8), we obtain  $x_{init\_n}$  in the improved version of  $\lambda$ -GSD:

$$\begin{aligned} x_{init\_n} &= \\ & \left[ \left( \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ 0 & \lambda \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{pmatrix}^T \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ 0 & \lambda \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{pmatrix} \right)^{-1} \begin{pmatrix} \bar{\mathbf{H}}_1 & \bar{\mathbf{H}}_2 \\ 0 & \lambda \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{pmatrix}^T \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \right]_n \\ &= \left[ \begin{pmatrix} \mathbf{H}_1^T \mathbf{H}_1 & \mathbf{H}_1^T \mathbf{H}_2 \\ \mathbf{H}_2^T \mathbf{H}_1 & \mathbf{H}_2^T \mathbf{H}_2 + \lambda^2 \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{pmatrix}^{-1} \mathbf{H}^T \mathbf{y} \right]_n \end{aligned} \quad (10)$$

Comparing with the first element of MMSE-DFE point,

$$\begin{aligned} x_{init\_n} &= \left[ \left( \mathbf{H}^T \mathbf{H} + \frac{\mathbf{I}_{n \times n}}{\gamma_s} \right)^{-1} \mathbf{H}^T \mathbf{y} \right]_n \\ &= \left[ \begin{pmatrix} \mathbf{H}_1^T \mathbf{H}_1 + \frac{\mathbf{I}_{(m-2) \times (m-2)}}{\gamma_s} & \mathbf{H}_1^T \mathbf{H}_2 \\ \mathbf{H}_2^T \mathbf{H}_1 & \mathbf{H}_2^T \mathbf{H}_2 + \frac{\mathbf{I}_{(n-m+2) \times (n-m+2)}}{\gamma_s} \end{pmatrix}^{-1} \mathbf{H}^T \mathbf{y} \right]_n \end{aligned} \quad (11)$$

We would like to choose  $\lambda$  such that (10) and (11) are as similar as possible. Thus, we propose to choose  $\lambda$  such that the two following matrices,  $\mathbf{A}_1 \triangleq \begin{pmatrix} \mathbf{H}_1^T \mathbf{H}_1 & \mathbf{H}_1^T \mathbf{H}_2 \\ \mathbf{H}_2^T \mathbf{H}_1 & \mathbf{H}_2^T \mathbf{H}_2 + \lambda^2 \mathbf{I}_{(n-m+1) \times (n-m+1)} \end{pmatrix}$  in (10) and

$$\mathbf{A}_2 \triangleq \begin{pmatrix} \mathbf{H}_1^T \mathbf{H}_1 + \mathbf{I}_{(m-2) \times (m-2)} / \gamma_s & \mathbf{H}_1^T \mathbf{H}_2 \\ \mathbf{H}_2^T \mathbf{H}_1 & \mathbf{H}_2^T \mathbf{H}_2 + \mathbf{I}_{(n-m+2) \times (n-m+2)} / \gamma_s \end{pmatrix} \text{ in}$$

(11) are as similar as possible. One way to measure the similarity between the two matrices is the norm of the difference. Here we use Frobenius norm to measure the difference and choose  $\lambda$  to minimize  $\|\mathbf{A}_1 - \mathbf{A}_2\|_F$ , i.e.,

$$\begin{aligned} \min_{\lambda > 0} \|\mathbf{A}_1 - \mathbf{A}_2\|_F &= \\ & \left\| \begin{pmatrix} \mathbf{I}_{(m-2) \times (m-2)} / \gamma_s & \mathbf{0} \\ \mathbf{0} & (1/\gamma_s - \lambda^2) \mathbf{I}_{(n-m+2) \times (n-m+2)} \end{pmatrix} \right\|_F \\ &= \frac{n}{\gamma_s} - \lambda^2 (n-m+2) \end{aligned} \quad (12)$$

Clearly, when  $\lambda = \sqrt{n / [(n-m+2)\gamma_s]}$ ,  $\|\mathbf{A}_1 - \mathbf{A}_2\|_F$  achieves the minimum, zero.

## 5. Illustrative Results

This section illustrates the merits of the improved ver-

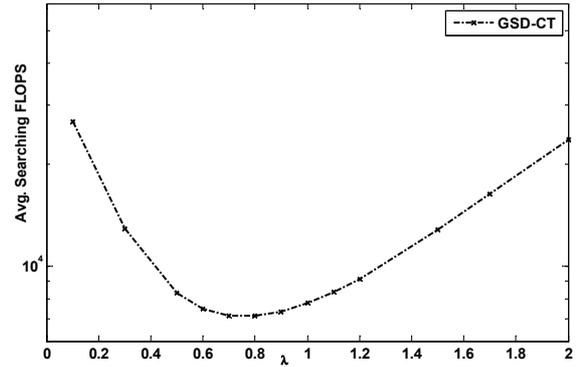


Figure 4. Avg. searching FLOPS vs.  $\lambda$  @ 4x8 MIMO, QPSK, 12dB,  $10^4$  runs.

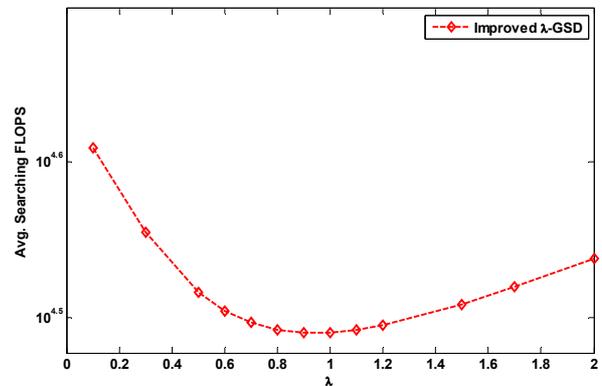


Figure 5. Avg. searching FLOPS vs.  $\lambda$  @ 4x8 MIMO, QPSK, 12dB,  $10^4$  runs.

<sup>4</sup>Shnorr-Euchner enumeration refers to the ordering of the integer points in each layer in a zig-zag manner, which is more efficient than natural ordering and recommended to be used for SD searching recently.

sion of  $\lambda$ -GSD and verify the efficiency of the selection of design parameter for the improved version of  $\lambda$ -GSD and GSD [6] via simulation in a flat fading underdetermined MIMO channel with  $n$  transmit and  $m$  receive antennas ( $m < n$ ). In simulations, we use the “accelerated” standard SD algorithm in [17] as the sub-algorithm for GSDs. The received signal can be represented as  $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$ , where  $\mathbf{x}$  is the transmitted vector from M-QAM and  $\mathbf{v}$  is an  $m \times 1$  AWGN noise vector with variance  $E(\mathbf{v}^H \mathbf{v}) = 2\sigma^2 \mathbf{I}_m$ .  $\mathbf{H} = [h_{ij}]$  is an underdetermined  $m \times n$  matrix where  $h_{ij}$  follows Rayleigh fading model  $\sim CN(0,1)$ . We define  $SNR = 10 \log_{10}(nE_s / (2q\sigma^2))$  dB, where  $E_s$  is the average M-ary QAM symbol energy. Note that the complex model would be transformed into an equivalent real model with  $2n$  unknowns (from  $L = \sqrt{M}$ -PAM equivalently) before using  $\lambda$ -GSD. The average searching FLOPS are counted as complexity measurement for simulation comparison for the GSDs.

Figure 2 provides the average searching FLOPS comparison between the original  $\lambda$ -GSD and the improved version for various  $\lambda$  values under a 2x4 16QAM flat-fading MIMO scenario at 28dB. Note that the upper bound for  $\lambda$  in these two GSD algorithms with  $\epsilon=1\%$  is 0.0167 and 0.0149 respectively. We can see that the complexity is reduced by about 5% with the better structure. Besides, Figure 2 indicates that in the improved version of  $\lambda$ -GSD for 16QAM, the complexity is quite insensitive to the choice of the design parameter  $\lambda$  as long as the value of  $\lambda$  is within the upper bound, similar to the case of the original  $\lambda$ -GSD.

Figure 3 plots the complexity comparison of three GSD algorithms for 4x8 QPSK flat-fading MIMO scenarios at SNR=12dB, including the original  $\lambda$ -GSD, the improved version and GSD in [6] (i.e., GSD-CT). Figure 3 shows the average searching FLOPS w.r.t. various values for the design parameter in all the three GSD algorithms for QPSK system. We can see that the complexity of the original  $\lambda$ -GSD is insensitive to the choice of the design parameter; yet the other two GSD algorithms are more sensitive to the design parameter and apparently there is an optimal value for the lowest complexity as expected.

Thus, we calculate the  $\lambda = 1/\sqrt{\gamma_s}$  value for a 4x8 MIMO QPSK system at 12dB and the resulted value for  $\lambda$  is  $\lambda_0=0.710468$ . Then we simulate the complexity for different  $\lambda$  values in the range (0.1, 2) in Figure 4, which indicates that the lowest complexity is achieved approximately at  $\lambda_0=0.710468$ , i.e., choosing  $\lambda$  equal to  $\lambda_0$  can lead to very good efficiency. Correspondingly, we calculate the  $\lambda = \sqrt{n/[(n-m+2)\gamma_s]}$  value of the improved version of  $\lambda$ -GSD for the same 4x8 MIMO QPSK system at 12dB and the resulted value for  $\lambda$  is  $\lambda_0=0.8987$ .

Then we simulate its corresponding complexity for different  $\lambda$  value in the range of (0.1, 2). Figure 5 indicates that the lowest complexity is achieved for  $\lambda$  around  $\lambda_0$ , i.e., choosing  $\lambda$  equal to the suggested value of  $\lambda_0=0.8987$  also can lead to very good efficiency for the improved version of  $\lambda$ -GSD.

## 6. Conclusions

An improved version of  $\lambda$ -GSD algorithm was first proposed for underdetermined linear systems by transforming the original problem into a full-column-rank one with better structure, so that standard SD can be efficiently applied. As the introduced transformation maintains the original problem dimension and has better system structure than the original one, lower complexity can be obtained compared to its counterparts, especially for high QAMs. Selection of design parameter  $\lambda$  was then studied for the improved version of  $\lambda$ -GSD. The resulted systematic approach to select the design parameter can also be used for GSD in [6], [7] in which the design parameters were only selected randomly or based on observations. Simulation results confirmed the high efficiency achieved by such choices of design parameters for these GSDs.

## 7. References

- [1] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, “Closest point search in lattices,” *IEEE Transactions Information Theory*, pp. 2201–2214, August 2002.
- [2] O. Damen, A. Chkeif, and J. C. Belfiore, “Lattice code decoder for space-time codes,” *IEEE Communication Letters*, pp. 161–163, May 2000.
- [3] H. Vikalo, B. Hassibi, and T. Kailath, “Iterative decoding for MIMO channels via modified sphere decoding,” *IEEE Transactions on Wireless Communication*, Vol. 3, No. 6, pp. 2299–2311, November 2004.
- [4] L. Brunel, “Multiuser detection techniques using maximum likelihood sphere decoding in multicarrier CDMA systems,” *IEEE Transactions on Wireless Communication*, Vol. 3, No. 3, pp. 949–957, May 2004.
- [5] T. Cui and C. Tellambura, “An efficient generalized sphere decoder for rank-deficient MIMO systems,” *IEEE VTC. Fall*, 2004.
- [6] T. Cui and C. Tellambura, “An efficient generalized sphere decoder for rank-deficient MIMO systems,” *IEEE Communication Letters*, Vol. 9, No. 5, pp. 423–425, May 2005.
- [7] X. Chang and X. Yang, “An efficient regularization approach for underdetermined MIMO system decoding,” *IWCMC, Hawaii, USA*, August, 2007.
- [8] P. Wang and T. Le-Ngoc, “An efficient multi-user detection scheme for overloaded group-orthogonal MC-CDMA systems,” *IET Communications*, Vol. 2, No. 2, pp.346–352, February 2008.

- [9] M. Damen, K. A. Meraim, and J. C. Belfiore, "Generalised sphere decoder for asymmetrical space-time communication architecture," *Electronics Letters*, Vol. 36, No. 2, pp. 166–167, January 2000.
- [10] P. Dayal and M. K. Varanasi, "A fast generalized sphere decoder for optimum decoding of under-determined MIMO systems," 41st Annual Allerton Conference on Communication Control and Computer, October 2003.
- [11] Z. Yang, C. Liu, and J. He, "A new approach for fast generalized sphere decoding in MIMO systems," *IEEE Signal Processing Letters*, Vol. 12, No. 1, pp. 41–44, January 2005.
- [12] X. W. Chang and X. H. Yang, "A new fast generalized sphere decoding algorithm for underdetermined sphere decoding algorithm for underdetermined MIMO systems," 23<sup>rd</sup> Biennial symposium on Communications, Kingston, ON, Canada, pp. 18–21, June 2006.
- [13] P. Wang and T. Le-Ngoc, "A low complexity generalized sphere decoding approach for underdetermined linear communication systems: performance and complexity evaluation," *IEEE Transactions on Communications*, Vol. 57, No. 11, pp. 3376–3388, November 2009.
- [14] P. Wang and T. Le-Ngoc, "On the expected complexity analysis of a generalized sphere decoding algorithm for underdetermined linear communication systems," *IEEE International Conference Communication (ICC)*, Glasgow, June 2007.
- [15] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Transactions on Information Theory*, Vol. 49, No. 10, pp. 2389–2402, October 2003.
- [16] H. Artes, D. Seethaler, and F. Hlawatsch, "Efficient detection algorithms for MIMO channels: a geometrical approach to approximate ML detection," *IEEE Transactions on Signal Processing*, Vol. 51, No. 11, pp. 2808–2820, November 2003.
- [17] J. Boutros, N. Gresset, L. Brunel, and M. Fossorier, "Soft-input soft-output lattice sphere decoder for linear channels," *IEEE Globecom 2003*, pp. 1583–1587, 2003.