Reduced Complexity Sphere Decoding

Boyu Li and Ender Ayanoglu

Center for Pervasive Communications and Computing
Department of Electrical Engineering and Computer Science
The Henry Samueli School of Engineering
University of California - Irvine
Irvine, California 92697-2625
Email: boyul@uci.edu, ayanoglu@uci.edu

***Abstract*—In Multiple-Input Multiple-Output (MIMO) systems, Sphere Decoding (SD) can achieve performance equivalent to full search Maximum Likelihood (ML) decoding with reduced complexity. Several researchers reported techniques that reduce the complexity of SD further. In this paper, a new technique is introduced which decreases the computational complexity of SD substantially, without sacrificing performance. The reduction is accomplished by deconstructing the decoding metric to decrease the number of computations and exploiting the structure of a lattice representation. Simulation results show that this approach achieves substantial gains for the average number of real multiplications and real additions needed to decode one transmitted vector symbol. As an example, for a 4×4 MIMO system, the gains in the number of multiplications are 85% with 4-QAM and 90% with 64-QAM, at low SNR.**

***Index Terms*—MIMO, ML Decoding, SD, Low Computational Complexity.**

I. Introduction

Multiple-Input Multiple-Output (MIMO) systems have drawn substantial research and development because they offer high spectral efficiency and performance in a given bandwidth. In such systems, the goal is to minimize the Bit Error Rate (BER) for a given Signal-to-Noise Ratio (SNR). A number of different MIMO systems exist. Most of these systems result in optimum decoding techniques that are complicated. Therefore a number of decoding algorithms with different complexity-performance tradeoffs have been introduced. Linear detection methods such as Zero-Forcing (ZF) or Minimum Mean Squared Error (MMSE) provide linear complexity, however their performance are suboptimal. Ordered successive interference cancellation decoders such as Vertical Bell Laboratories Layered Space-Time (V-BLAST) algorithm, show slightly better performance compared to ZF and MMSE, but suffer from error propagation and are still suboptimal [1]. It is well-known that Maximum Likelihood (ML) detection is the optimum method. However, the complexity of the ML algorithm in MIMO systems increases exponentially with the number of possible constellation points for the modulation scheme, making the algorithm unsuitable for practical purposes [2]. Sphere Decoding (SD), on the other hand, is proposed as an alternative for ML that provides optimal or near-optimal performance with reduced complexity [3].

Although the complexity of SD is much smaller than ML decoding, there is room for complexity reduction in conventional SD. To that end, several complexity reduction techniques for SD have been proposed. In [4] and [5], attention is drawn to initial radius selection strategy, since an inappropriate initial radius can result in either a large number of lattice points to be searched or a large number of restart actions. In [6], this complexity is attacked by making a proper choice to update the sphere radius. In [7], the Schnorr-Euchner (SE) strategy is applied to SD, which executes intelligent enumeration of candidate symbols at each level to reduce the number of visited nodes when the system dimension is small [8]. Channel reordering techniques can also be applied to reduce the number of visited nodes [8], [9], [10]. Other methods, such as the K-best lattice decoder [11], [12], can significantly reduce the complexity at low SNR, but with the tradeoff of BER performance degradation.

In this paper, the complexity of SD is efficiently improved by reducing the number of operations required at each node to obtain the ML solution for flat fading channels. This complexity reduction is achieved by deconstructing the decoding metric in order to reduce the number of computations and exploiting the structure of a lattice representation of SD [9], [10]. In simulations, 2×2 and 4×4 MIMO systems with 4-QAM and 64-QAM have been studied. In these systems, the reduction in the number of real additions is in the range of $40\% - 75\%$, and the reduction in the number of real multiplications is in the range of $70\% - 90\%$, without any change in performance. The complexity gains increase with the MIMO system dimension or the modulation alphabet size.

The remainder of this paper is organized as follows: In Section II, the problem definition is introduced and a brief review of conventional SD algorithm is presented. In Section III, a new technique to implement the SD algorithm with low computational complexity is proposed, and the mathematical derivations for the complexity reduction are carried out. In Section IV, complexity comparisons for different number of antennas or modulation schemes are provided. Finally, a conclusion is provided in Section V.

978-1-4577-9538-2/11/$26.00 ©2011 IEEE

II. CONVENTIONAL SPHERE DECODER

In this paper, MIMO systems using square Quadrature Amplitude Modulation (QAM) with N_t transmit and N_r receive antennas are considered, and the channel is assumed to be flat fading. Then, the input-output relation is given by

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{v}, \tag{1}$$

where $\mathbf{r} \in \mathbb{C}^{N_r}$ is the N_r dimensional received vector symbol and $\mathbb{C}$ denotes the set of complex numbers; $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ is the channel matrix whose channel coefficients are independent and identically distributed (i.i.d.) zero-mean, unit-variance complex Gaussian random variables; $\mathbf{s} \in \mathbb{C}^{N_t}$ is an N_t dimensional transmitted complex vector with each element in square QAM format; and $\mathbf{v} \in \mathbb{C}^{N_r}$ is a zero-mean white Gaussian noise vector with variance matrix $\sigma^2\mathbf{I}$.

Assuming $\mathbf{H}$ is known at the receiver, ML detection is

$$\hat{\mathbf{s}} = \arg \min_{\mathbf{s} \in \chi^{N_t}} \|\mathbf{r} - \mathbf{H}\mathbf{s}\|^2 \tag{2}$$

where χ denotes the sample space for QAM modulation scalar symbols. For example, $\chi = \{-3, -1, 1, 3\}^2$ for 16-QAM.

Solving (2) is known to be NP-hard, given that a full search over the entire lattice space is performed [13]. SD, on the other hand, solves (2) by searching only lattice points that lie inside a sphere of radius δ centering around the received vector $\mathbf{r}$.

A frequently used solution for the QAM-modulated signal model is to decompose the N_r-dimensional complex-valued problem (1) into a $2N_r$-dimensional real-valued problem, which can be written as

$$\begin{bmatrix} \Re\{\mathbf{r}\} \\ \Im\{\mathbf{r}\} \end{bmatrix} = \begin{bmatrix} \Re\{\mathbf{H}\} & -\Im\{\mathbf{H}\} \\ \Im\{\mathbf{H}\} & \Re\{\mathbf{H}\} \end{bmatrix} \begin{bmatrix} \Re\{\mathbf{s}\} \\ \Im\{\mathbf{s}\} \end{bmatrix} + \begin{bmatrix} \Re\{\mathbf{v}\} \\ \Im\{\mathbf{v}\} \end{bmatrix}, \tag{3}$$

where $\Re\{\mathbf{r}\}$ and $\Im\{\mathbf{r}\}$ denote the real and imaginary parts of $\mathbf{r}$ respectively [3], [13]. Let

$$\mathbf{y} = \left[\Re\{\mathbf{r}\}^T \ \Im\{\mathbf{r}\}^T\right]^T, \tag{4}$$

$$\bar{\mathbf{H}} = \begin{bmatrix} \Re\{\mathbf{H}\} & -\Im\{\mathbf{H}\} \\ \Im\{\mathbf{H}\} & \Re\{\mathbf{H}\} \end{bmatrix}, \tag{5}$$

$$\mathbf{x} = \left[\Re\{\mathbf{s}\}^T \ \Im\{\mathbf{s}\}^T\right]^T, \tag{6}$$

$$\mathbf{n} = \left[\Re\{\mathbf{v}\}^T \ \Im\{\mathbf{v}\}^T\right]^T, \tag{7}$$

then (3) can be written as

$$\mathbf{y} = \bar{\mathbf{H}}\mathbf{x} + \mathbf{n}. \tag{8}$$

Assuming $N_t = N_r = N$ in the sequel, and using the QR decomposition of $\bar{\mathbf{H}} = \mathbf{Q}\mathbf{R}$, where $\mathbf{R}$ is an upper triangular matrix, and the matrix $\mathbf{Q}$ is unitary, SD solves

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \xi} \|\bar{\mathbf{y}} - \mathbf{R}\mathbf{x}\|^2 \tag{9}$$

where $\bar{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$. Let Ω denote the sample space for one dimension of QAM-modulated symbols, e.g., $\Omega = \{-3, -1, 1, 3\}$ for 16-QAM, then ξ denotes a subset of Ω^{2N} whose elements satisfy $\|\bar{\mathbf{y}} - \mathbf{R}\mathbf{x}\|^2 < \delta^2$.

The SD algorithm can be viewed as a pruning algorithm on a tree of depth $2N$, whose branches correspond to elements drawn from the set Ω [9], [10], [13]. Conventional SD implements a Depth-First Search (DFS) strategy in the tree, which can achieve ML performance.

Conventional SD starts the search process from the root of the tree, and then searches down along branches until the total weight of a node exceeds the square of the sphere radius, δ^2. At this point, the corresponding branch is pruned, and any path passing through that node is declared as improbable for a candidate solution. Then the algorithm backtracks and proceeds down a different branch. Whenever a valid lattice point at the bottom level of the tree is found within the sphere, δ^2 is set to the newly-found point weight, thus reducing the search space for finding other candidate solutions. In the end, the path from the root to the leaf that is inside the sphere with the lowest weight is chosen to be the estimated solution $\hat{\mathbf{x}}$. If no candidate solutions can be found, the tree is searched again with a larger initial radius.

III. PROPOSED SPHERE DECODER

The complexity of SD is measured by the number of operations required per visited node multiplied by the number of visited nodes throughout the search procedure [13]. The complexity can be reduced by either reducing the number of visited nodes or the number of operations to be carried out at each node, or both. Making a judicious choice of initial radius to start the algorithm with [4], [5], executing a proper sphere radius update strategy [6], applying an improved search strategy [7], and exploiting channel reordering [8], [9], [10] can all reduce the number of visited nodes. In this paper, our focus is on reducing the average number of operations required at each node for SD.

The node weight is given by [9], [10],

$$w(\mathbf{x}^{(l)}) = w(\mathbf{x}^{(l+1)}) + w_{pw}(\mathbf{x}^{(l)}), \tag{10}$$

with $l = 2N, \cdots, 1$, $w(\mathbf{x}^{(2N+1)}) = 0$, and $w_{pw}(\mathbf{x}^{(2N+1)}) = 0$, where $\mathbf{x}^{(l)}$ denotes the partial vector symbol at layer l. The partial weight corresponding to $\mathbf{x}^{(l)}$ is written as

$$w_{pw}(\mathbf{x}^{(l)}) = |\bar{y}_l - \sum_{k=l}^{2N} R_{l,k} x_k|^2, \tag{11}$$

where $R_{i,j}$ denotes the $(i,j)^{th}$ element of $\mathbf{R}$, and x_i denotes the i^{th} element of $\mathbf{x}$.

A. Check-Table $\mathbb{T}$

Note that for one channel realization, both $\mathbf{R}$ and Ω are independent of time. In other words, to decode different received symbols for one channel realization, the only term in (11) which depends on time is $\bar{y}_l$. Consequently, a check-table $\mathbb{T}$ is constructed to store all terms of $R_{i,j}x$, where $R_{i,j} \neq 0$ and $x \in \Omega$, before starting the tree search procedure. Equations (10) and (11) imply that only one real multiplication is needed instead of $2N - l + 2$ for each node to calculate the node weight by using $\mathbb{T}$. As a result, the number of real multiplications can be significantly reduced.

Taking the square QAM lattice structure into consideration, Ω can be divided into two smaller sets Ω_1 with negative elements and Ω_2 with positive elements. Take 16-QAM for example, $\Omega = \{-3, -1, 1, 3\}$, then $\Omega_1 = \{-3, -1\}$ and $\Omega_2 = \{1, 3\}$. Any negative element in Ω_1 has a positive element with the same absolute value in Ω_2. Consequently, in order to build $\mathbb{T}$, only terms in the form of $R_{i,j}x$, where $R_{i,j} \neq 0$ and $x \in \Omega_1$, need to be calculated and stored. Hence, the size of $\mathbb{T}$ is

$$|\mathbb{T}| = \frac{N_R|\Omega|}{2}, \tag{12}$$

where N_R denotes the number of non-zero elements in matrix $\mathbf{R}$, and $|\Omega|$ denotes the size of Ω.

In order to build $\mathbb{T}$, both the number of terms that need to be stored and the number of real multiplications required are $|\mathbb{T}|$. Since the channel is assumed to be flat fading, only one $\mathbb{T}$ needs to be built in one burst. If the burst length is very long, its computational complexity can be neglected.

B. Intermediate Node Weights

Define

$$M(\mathbf{x}^{(l+1)}) = \bar{y}_l - \sum_{k=l+1}^{2N} R_{l,k}x_k, \tag{13}$$

where $M(\mathbf{x}^{(2N+1)}) = 0$, then (11) can be rewritten as

$$w_{pw}(\mathbf{x}^{(l)}) = |M(\mathbf{x}^{(l+1)}) - R_{l,l}x_l|^2. \tag{14}$$

Equation (13) shows that $M(\mathbf{x}^{(l+1)})$ is independent of x_l, which means for any node not in the last level of the search tree, all children nodes share the same $M(\mathbf{x}^{(l+1)})$. In other words, for these nodes, their $M(\mathbf{x}^{(l+1)})$ values need to be calculated only once to get the whole set of weights for their children nodes. Consequently, the number of operations will be reduced if $M(\mathbf{x}^{(l+1)})$ values are stored at each node, except nodes of the last level, until the whole set of their children are visited. Based on (10), (13), and (14), by storing the $M(\mathbf{x}^{(l+1)})$ values, the number of real additions needed to get all partial weights of the children nodes at layer l, for a parent node of layer $l+1$, reduces to $2N - l + |\Omega|$ from $(2N - l + 1)|\Omega|$. Note that after implementing the check-table $\mathbb{T}$, storing $M(\mathbf{x}^{(l+1)})$ values does not affect the number of real multiplications.

C. New Lattice Representation

In our previous work [9], [10], a new lattice representation was proposed for (8) that enables decoding the real and imaginary parts of each complex symbol independently. Also, a near ML decoding algorithm, which combines DFS, K-best decoding, and quantization, was introduced. In this work, a different application of the lattice representation, which achieves no performance degradation, is employed.

For the new lattice representation, (4)-(7) become

$$\mathbf{y} = \left[\Re\{r_1\}\ \Im\{r_1\}\ \cdots\ \Re\{r_N\}\ \Im\{r_N\}\right]^T, \tag{15}$$

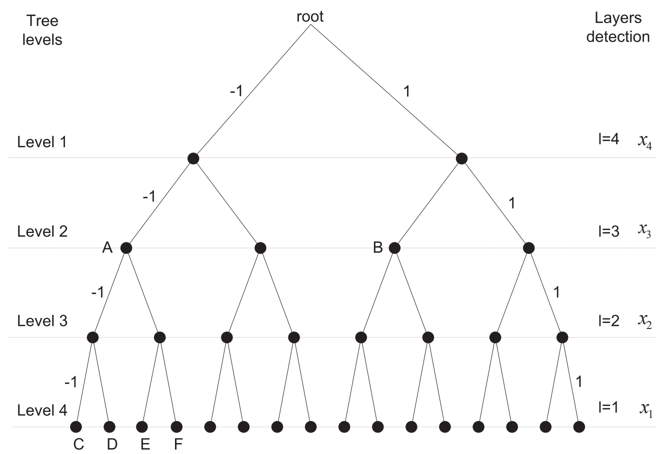

Fig. 1. Tree structure for a 2x2 system employing 4-QAM

$$\bar{\mathbf{H}} = \begin{bmatrix} \Re\{H_{1,1}\} & -\Im\{H_{1,1}\} & \cdots & \Re\{H_{1,N}\} & -\Im\{H_{1,N}\} \\ \Im\{H_{1,1}\} & \Re\{H_{1,1}\} & \cdots & \Im\{H_{1,N}\} & \Re\{H_{1,N}\} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \Re\{H_{N,1}\} & -\Im\{H_{N,1}\} & \cdots & \Re\{H_{N,N}\} & -\Im\{H_{N,N}\} \\ \Im\{H_{N,1}\} & \Re\{H_{N,1}\} & \cdots & \Im\{H_{N,N}\} & \Re\{H_{N,N}\} \end{bmatrix}, \tag{16}$$

$$\mathbf{x} = \left[\Re\{s_1\}\ \Im\{s_1\}\ \cdots\ \Re\{s_N\}\ \Im\{s_N\}\right]^T, \tag{17}$$

$$\mathbf{n} = \left[\Re\{v_1\}\ \Im\{v_1\}\ \cdots\ \Re\{v_N\}\ \Im\{v_N\}\right]^T. \tag{18}$$

The structure of the new lattice representation (15)-(18) becomes advantageous after applying the QR decomposition to $\bar{\mathbf{H}}$. By doing so, and due to the special form of orthogonality between each pair of columns, all elements $R_{k,k+1}$ for $k = 1, 3, \ldots, 2N-1$, in the upper triangular matrix $\mathbf{R}$ become zero. The locations of these zeros introduce orthogonality between the real and imaginary parts of every detected symbol, which can be taken advantage of to reduce the computational complexity of SD. The following example is provided to explain this.

Example: Consider a MIMO system having $N_r = N_t = N = 2$, and employing 4-QAM. Then, SD constructs a tree with $2N = 4$ levels, where the branches coming out from each node represent the real values in the set $\Omega = \{-1, 1\}$. This tree is shown in Fig. 1. Now using the real-valued lattice representation (15)-(18), and applying the QR decomposition to the channel matrix, the input-output relation is given by

$$\begin{bmatrix} \bar{y}_1 \\ \bar{y}_2 \\ \bar{y}_3 \\ \bar{y}_4 \end{bmatrix} = \begin{bmatrix} R_{1,1} & 0 & R_{1,3} & R_{1,4} \\ 0 & R_{2,2} & R_{2,3} & R_{2,4} \\ 0 & 0 & R_{3,3} & 0 \\ 0 & 0 & 0 & R_{4,4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix}. \tag{19}$$

Based on (11) and (19), calculating partial node weights for the first level and the second level are independent, similar to the third level and the forth level, because of the additional zero locations in the $\mathbf{R}$ matrix. For instance, the partial weights of node A and node B only depend on x_3 but x_4, and the partial weights of node C, node D, node E, and node F, depend on x_4, x_3, and x_1 except x_2. In other words, the partial weights of node A and node B are equal, and only need to be calculated once. Similarly, partial weights of node C and node D can be used when calculating the partial weights of node E and node F, respectively.

SD is then modified because of this feature. Once the tree is searched in layer l, where l is an odd number, partial weights of this node and all of its sibling nodes are computed, stored, and recycled when calculating partial node weights with the same grandparent node of layer $l+2$ but with different parent nodes of layer $l+1$.

By applying the modification, further complexity reduction is achieved beyond the reduction due to the check-table $\mathbb{T}$ and intermediate $M(\mathbf{x}^{(l+1)})$ values. For a node of layer $l+2$, where l is an odd number, let $\alpha \in [0, |\Omega|]$ denote the number of non-pruned branches for its children nodes of layer $l+1$. If $\alpha = 0$, which means all branches of its children nodes of layer $l+1$ are pruned, the number of operations needed stay the same. If $\alpha \neq 0$, to get all partial weights of its grandchildren nodes in layer l, the number of real multiplications and real additions reduce further from $(\alpha+1)|\Omega|$ to $2|\Omega|$, and $(\alpha+1)(2N-l-1+|\Omega|)+\alpha$ to $2(2N-l-1+|\Omega|)$, respectively.

IV. SIMULATION RESULTS

To verify the proposed technique, simulations are carried out for 2×2 and 4×4 systems using 4-QAM and 64-QAM. Assuming a sequence of vector symbols are transmitted, and considering multiple channel realizations for each simulation, the average number of real multiplications and real additions for decoding one transmitted vector symbol are calculated. In the figures, conventional SD is denoted by CSD and proposed SD by PSD. In our simulations, $\delta^2 = 2\sigma_n^2 N$ is chosen as the square of initial radius. A lattice point lies inside a sphere of this radius with high probability [7].

Fig. 2 and Fig. 3 show comparisons for the number of operations between conventional SD and proposed SD for 2×2 systems using 4-QAM and 64-QAM. For 4-QAM, the complexity gains for the average numbers of real multiplications and real additions are around 70% and 45% respectively at high SNR. Corresponding numbers are 75% and 40% respectively at low SNR. For the 64-QAM case, gains increase to around 70% and 65% at high SNR respectively, while they are around 85% and 60% at low SNR respectively.

Similarly, Fig. 4 and Fig. 5 show complexity comparisons using 4-QAM and 64-QAM for 4×4 systems. For 4-QAM, gains for the average number of real multiplications and real additions are around 80% and 50% respectively at high SNR, while they are around 85% and 45% respectively at low SNR. For 64-QAM, gains rise up to around 80% and 75% respectively at high SNR, while they are around 90% and 70% respectively at low SNR.

Simulation results show that proposed SD reduces the complexity significantly compared to conventional SD, particularly for real multiplications, which are the most expensive operations in terms of machine cycles, and the reduction becomes larger as the system dimension or the modulation alphabet size increases. An important property of our proposed SD is that the substantial complexity reduction achieved causes no performance degradation. The proposed technique can be combined with other techniques which reduce the number of visited nodes such as SE, and other near-optimal techniques such as K-best.

V. CONCLUSIONS

A simple and general technique to implement the SD algorithm with low computational complexity is proposed in this paper. The focus of the technique is on reducing the average number of operations required at each node for SD. The BER performance of the proposed SD is the same as conventional SD. Moreover, a substantial complexity reduction is achieved. Simulation results are provided for 2×2 and 4×4 systems employing 4-QAM and 64-QAM. The complexity gains for the average numbers of real multiplications and real additions are substantial, ranging from 70% to 90% and 40% to 75% respectively, based on the number of antennas and the constellation size of modulation schemes. These complexity gains become larger as the system dimension or the modulation alphabet size increases.

REFERENCES

[1] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, "VLSI Implementation of MIMO Detection Using the Sphere Decoding Algorithm," *IEEE J. Solid-State Circuits*, vol. 40, pp. 1566–1577, Jul. 2005.

[2] E. Zimmermann, W. Rave, and G. Fettweis, "On the Complexity of Sphere Decoding," in *Proc. WPMC 2004*, Abano Terme, Italy, Sep. 2004.

[3] J. Jaldén and B. Ottersten, "On the Complexity of Sphere Decoding in Digital Communications," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1474–1484, Apr. 2005.

[4] B. Cheng, W. Liu, Z. Yang, and Y. Li, "A New Method for Initial Radius Selection of Sphere Decoding," in *Proc. IEEE ISCC 2007*, Aveiro, Portugal, Jul. 2007, pp. 19–24.

[5] H. G. Han, S. K. Oh, S. J. Lee, and D. S. Kwon, "Computational Complexities of Sphere Decoding According to Initial Radius Selection Schemes and an Efficient Initial Radius Reduction Scheme," in *Proc. IEEE GLOBECOM 2005*, vol. 4, St. Louis, MO, USA, Nov. 2005, pp. 2354–2358.

[6] B. Hassibi and H. Vikalo, "On the Sphere-Decoding Algorithm I. Expected Complexity," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2806–2818, Aug. 2005.

[7] M. O. Damen, H. E. Gamel, and G. Caire, "On Maximum-Likelihood Detection and the Search for the Closest Lattice Point," *IEEE Trans. Inf. Theory*, vol. 49, no. 8, pp. 2389–2402, Oct. 2003.

[8] G. Rekaya and J. Belfiore, "On the Complexity of ML Lattice Decoders for Decoding Linear Full-Rate Space-Time Codes," in *Proc. IEEE ISIT 2003*, Pacifico Yokohama, Kanagawa, Japan, Jul. 2003.

[9] L. Azzam and E. Ayanoglu, "Reduced Complexity Sphere Decoding for Square QAM via a New Lattice Representation," in *Proc. IEEE GLOBECOM 2007*, Washington, D.C., USA, Nov. 2007, pp. 4242–4246.

[10] ——, "Reduced Complexity Sphere Decoding via a Reordered Lattice Representation," *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2564–2569, Sep. 2009.

[11] K. Wong, C. Tsui, and R. S. C. W. Mow, "A VLSI Architecture of a K-Best Lattice Decoding Algorithm for MIMO Channels," in *Proc. IEEE ISCAS 2002*.

[12] T. Huynh, D. Hoang, M. R. Islam1, and J. Kim, "Two-Level-Search Sphere Decoding Algorithm for MIMO Detection," in *Proc. IEEE ISWCS 2008*, Reykjavik, Iceland, Oct. 2008, pp. 448–452.

[13] B. Hassibi and H. Vikalo, "On the Expected Complexity of Integer Least-Squares Problems," in *Proc. IEEE ICASSP 2002*, vol. 2, Orlando, FL, USA, May 2002, pp. 1497–1500.

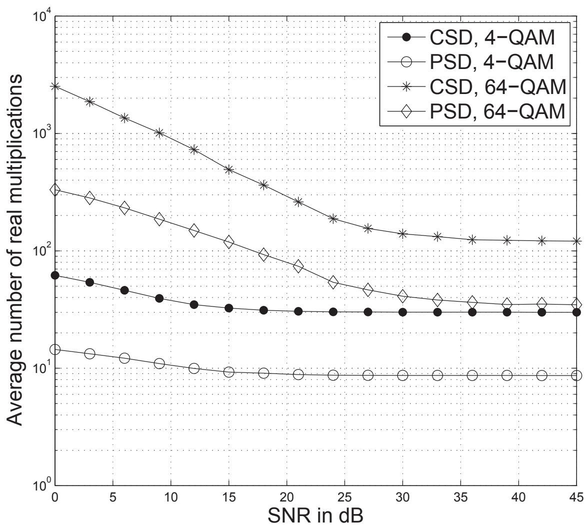

Fig. 2. Average number of real multiplications vs. SNR for conventional SD and proposed SD over a 2×2 MIMO flat fading channel.

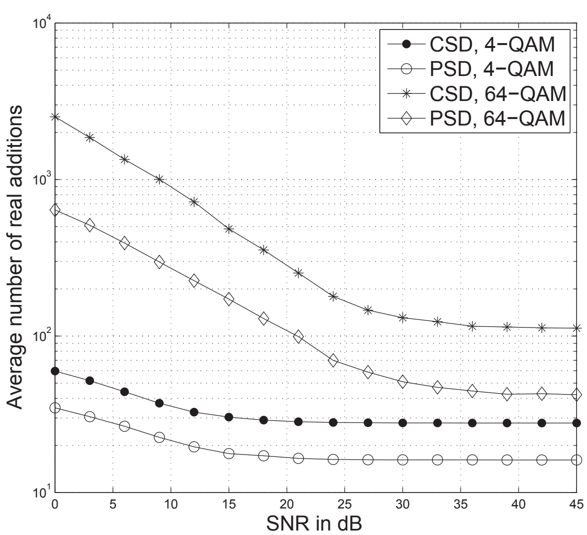

Fig. 3. Average number of real additions vs. SNR for conventional SD and proposed SD over a 2×2 MIMO flat fading channel.

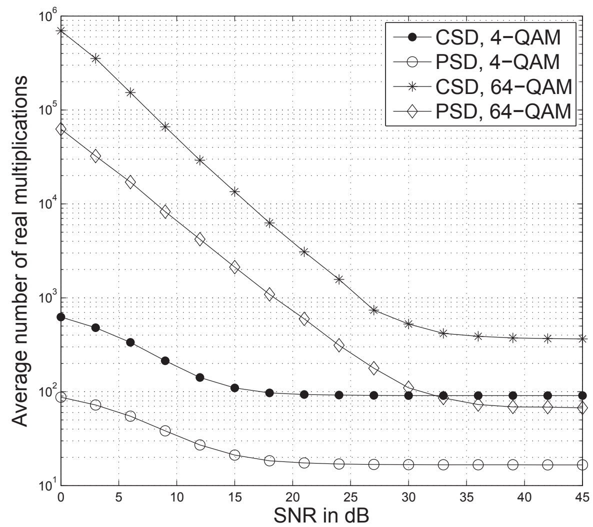

Fig. 4. Average number of real multiplications vs. SNR for conventional SD and proposed SD over a 4×4 MIMO flat fading channel.

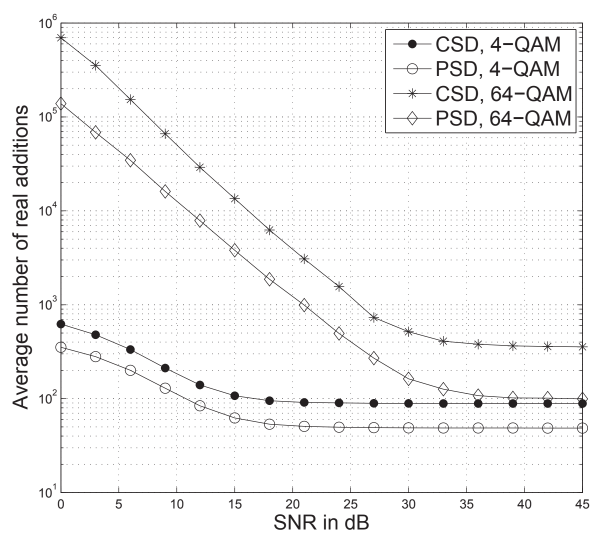

Fig. 5. Average number of real additions vs. SNR for conventional SD and proposed SD over a 4×4 MIMO flat fading channel.