

Dynamic K-Best Sphere Decoding Algorithms for MIMO Detection

Chengzhe Piao, Yang Liu, Kaihua Jiang, Xinyu Mao

School of electronic engineering and computer science, Peking University, Beijing, China

Email: 1000012736@pku.edu.cn, 1000012786@pku.edu.cn, 1000012726@pku.edu.cn, xymao@pku.edu.cn

Received May, 2013

ABSTRACT

Multiple Input Multiple Output (MIMO) technology is of great significance in high data rate wireless communication. The K-Best Sphere Decoding (K-Best SD) algorithm was proposed as a powerful method for MIMO detection that can approach near-optimal performance. However, some extra computational complexity is contained in K-Best SD. In this paper, we propose an improved K-Best SD to reduce the complexity of conventional K-Best SD by assigning K for each level dynamically following some rules. Simulation proves that the performance degradation of the improved K-Best SD is very little and the complexity is significantly reduced.

Keywords: Multiple Input Multiple Output (MIMO); Detection; K-Best Sphere Decoding (K-Best SD)

1. Introduction

Multiple-input multiple-output (MIMO) system, which utilizes the spatial diversity, has been extensively studied during the past decade, because it has high spectral efficiency [1]. The optimal performance of MIMO system detection can be achieved by maximum likelihood (ML) detection algorithm based on exhaustive search. However, the computational complexity is significantly affected by the size of modulation constellations and the antenna numbers. It increases exponentially with them. In order to decrease the computational complexity and apply in hardware, an improved algorithm, called the sphere detection algorithm (SD), is studied in [2,3]. The SD can offer a performance near maximum-likelihood algorithm with an acceptable complexity.

In the SD, detection is transformed into a depth-first tree search problem. This leads to a disadvantage that the computational complexity varies so obviously with different signals and channels that implementations of SD will be inconvenient. Hence the K-Best algorithm based on the breadth-first tree search strategy is put forward [4-5]. In the K-Best algorithm, only the best K candidates at each level of tree are reserved for further search in next level. This eliminates the disadvantage above. The fixed computational complexity makes implementation easily. However, comparing with ML algorithm, the K-best algorithm suffers performance degradation especially when K is small. It is because a larger value of K makes it less possible that the K-Best solution is not equal to the ML solution. The existing K-Best algorithm

cannot meet the increasing need for low computational complexity and high data rate of MIMO system.

In view of above, we proposed a new dynamic K-Best algorithm in this paper. In conventional K-best algorithm, a set consisted of K candidates, which have the smallest Euclidean distances, is kept for search in the next level. The difference between the smallest Euclidean distance and second smallest Euclidean distance is related to the possibility that the K-best solution is near to the ML solution. In the proposed algorithm, the value of K can be adjusted according to the difference so that the computational complexity is significantly reduced with small, even negligible, performance degradation.

The remainder of this paper is organized as follows. In section 2, the fundamental model of MIMO system is introduced. And we will review several basic decoding algorithms. In section 3, we will introduce the main ideas of sphere decoding (SD) algorithms. In section 4, we will describe our dynamic K-best algorithms in detail. In section 5, we will present our simulation results to demonstrate its complexity decrease. In section 6, the conclusion will be drawn.

2. Background Review

2.1. MIMO System Model

Consider a MIMO system with N_T transmit antennas and N_R receive antennas and assume that channel of this system is a Rayleigh fading channel. The relationship between the transmitted signal and received signal can be described as follows:

$$\tilde{\mathbf{y}} = \tilde{\mathbf{H}}\tilde{\mathbf{x}} + \tilde{\mathbf{n}} \quad (1)$$

in which $\tilde{\mathbf{y}} = [\tilde{y}_1, \tilde{y}_2 \dots \tilde{y}_{N_R}]^T$ denotes the N_R dimensional received signal vector and $\tilde{\mathbf{x}} = [\tilde{x}_1, \tilde{x}_2 \dots \tilde{x}_{N_T}]^T$ denotes the N_T dimensional transmitted signal vector. And the noise vector $\tilde{\mathbf{n}} = [\tilde{n}_1, \tilde{n}_2 \dots \tilde{n}_{N_R}]^T$ is a complex zero-mean additive Gaussian noise vector with variance σ^2 per dimension. Its powers of real parts and imaginary parts are equal. The flat fading channel is described by a complex $N_R \times N_T$ matrix $\tilde{\mathbf{H}}$.

$$\tilde{\mathbf{H}} = \begin{bmatrix} \tilde{h}_{11} & \tilde{h}_{12} & \dots & \tilde{h}_{1N_T} \\ \tilde{h}_{21} & \tilde{h}_{22} & \dots & \tilde{h}_{2N_T} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{h}_{N_R1} & \tilde{h}_{N_R2} & \dots & \tilde{h}_{N_RN_T} \end{bmatrix} \quad (2)$$

The element \tilde{h}_{ij} of channel matrix stand for the path gain from the j_{th} transmit antenna to the i_{th} receive antenna. It is supposed that channel matrix is fully known by receiver. Meanwhile, in order to simplify the calculation, we assume that $N_T = N_R = N$.

The complex channel matrix, transmit vector, receive vector, and noise vector can be transformed into its real representation.

$$\mathbf{H} = \begin{bmatrix} \Re(\tilde{\mathbf{H}}) & -\Im(\tilde{\mathbf{H}}) \\ \Im(\tilde{\mathbf{H}}) & \Re(\tilde{\mathbf{H}}) \end{bmatrix} \quad (3)$$

$$\mathbf{y} = \begin{bmatrix} \Re(\tilde{\mathbf{y}}) \\ \Im(\tilde{\mathbf{y}}) \end{bmatrix}, \mathbf{x} = \begin{bmatrix} \Re(\tilde{\mathbf{x}}) \\ \Im(\tilde{\mathbf{x}}) \end{bmatrix}, \mathbf{n} = \begin{bmatrix} \Re(\tilde{\mathbf{n}}) \\ \Im(\tilde{\mathbf{n}}) \end{bmatrix} \quad (4)$$

where $\Re(\cdot)$ stands for the real part and $\Im(\cdot)$ stands for the imaginary part.

Utilizing these real representations, (1) is transformed into

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \quad (5)$$

2.2. ZF Decoding

The Zero-Forcing (ZF) Decoding result is given by

$$\mathbf{x}_{ZF} = \mathbf{x} + \sqrt{\frac{M_T}{E_s}} \mathbf{H}^\dagger \mathbf{n} \quad (6)$$

where $\mathbf{H}^\dagger = (\mathbf{H}^H \mathbf{H})^{-1} \mathbf{H}^H$ is the Moore-Penrose inverse of the channel matrix \mathbf{H} [6]. ZF decoding transforms the joint decoding into N_T single stream decoding so as to reduce the complexity significantly. However, the reduction comes at the expense of large noise amplification leading to a much worse performance.

2.3. ML Decoding

The ML performance is optimal. For the receiver, the conditional probability density function $p_y(\mathbf{y}|\mathbf{x}, \mathbf{H})$ in the case of a Gaussian channel is given by

$$p_y(\mathbf{y}|\mathbf{x}, \mathbf{H}) = \frac{1}{(2 \cdot \pi)^N \cdot (\sigma_y^2)^N} \cdot \exp\left(-\frac{\mu}{2 \cdot \sigma_y^2}\right) \quad (7)$$

with $\mu = \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$ [7]. Therefore, maximizing the conditional pdf amounts is equal to minimizing $\mu = \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$ because the other terms are constant factors. So the mathematical model of ML detection is

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \Omega} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \quad (8)$$

where Ω is the set of all transmitted vectors.

For MIMO systems with N transmit antennas and the modulation order of m , Ω contains 2^{Nm} elements, and obviously ML Decoding using exhaustive search must search the whole set Ω to find the optimal solution $\hat{\mathbf{x}}$. Hence the complexity of ML Decoding increases exponentially with the product of m and N , so for large constellations and large N ML Decoding is impractical.

3. Sphere Decoding

3.1. Sphere Decoding

As is mentioned above, the ML detection based on exhaustive search is difficult in practice. Therefore a restriction of searching space is necessary. The SD is effective in reducing the complexity of ML detection by means of enumerating the lattice points within a distance lower than a given maximum R_0 named sphere radius. So the mathematical description of SD is

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in L(R_0)} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \quad (9)$$

Here $L(R_0)$ is a subset of the searching space of ML. But unfortunately, this kind of SD is not stable enough in that it cannot reduce much complexity if R_0 is too large, and on the other hand, it easily discards the ML solution at early stages of searching if R_0 is too small [8].

In order to simplify the searching, a decoding tree is usually employed. To create such a tree, an upper triangular structure \mathbf{R} of the channel matrix \mathbf{H} obtained by QR decomposition is required. Using QR decomposition, the MIMO channel model can be rewritten as below:

$$\mathbf{Q}^H \mathbf{y} = \mathbf{R}\mathbf{x} + \mathbf{Q}^H \mathbf{n} \quad (10)$$

or

$$\boldsymbol{\rho} = \mathbf{R}\mathbf{x} + \boldsymbol{\eta} \quad (11)$$

Since \mathbf{Q} is a unitary matrix, it can maintain the statistical characteristics of noise vector \mathbf{n} . Thus the new system is equivalent to the former one. In this way, the renewed expression of SD is

$$\begin{aligned} \hat{\mathbf{x}} &= \arg \min_{\mathbf{x} \in L(R_0)} \|\boldsymbol{\rho} - \mathbf{R}\mathbf{x}\|^2 \\ &= \arg \min_{\mathbf{x} \in L(R_0)} \sum_{j=1}^N (\rho_j - \sum_{i=j}^N x_i R_{ji})^2 \\ &= \arg \min_{\mathbf{x} \in L(R_0)} \sum_{j=1}^N e_j \end{aligned} \quad (12)$$

Here j is the level number of the tree and N is the number of antennas of the transmitter and the receiver. The nodes of the tree store the information of the partial solutions in order to find the optimal solution. The square of the Euclidean distance, namely $\|\rho - \mathbf{R}\mathbf{x}\|^2$, is calculated by computing partial Euclidean distance (PED) d_j for each level j , i.e.

$$d_j = \sqrt{\sum_{k=j}^N e_k} \quad (13)$$

Obviously

$$\|\rho - \mathbf{R}\mathbf{x}\| = d_N \quad (14)$$

3.2. Depth-First SD

The Depth-First Sphere Decoding algorithms contain forward and backward directions of searching the tree. It is carried out as below:

- 1) Set a large number as *optdist* (distance of the optimal solution);
- 2) Go forward to the first leaf of the tree;
- 3) Calculate the lowest distance of the nodes expanded by the same node with the leaf to update the *optdist*;
- 4) Go back to the former level and try the next node; if the end of nodes is reached, continue this step until there is a node available; if no such node exists, go to step 7.
- 5) If the PED of the node is larger than *optdist*, then go to step 4; else expand the node and go to step 6;
- 6) If the leaf is reached, go to step 3; else go to step 5.
- 7) Choose the solution whose PED is *optdist* as the optimal solution.

3.3. Fixed K-Best SD

Compared with exhaustive search, K-Best SD uses Breadth-First Search (BFS). BFS only allows forward direction and the information of nodes at current level will be kept before the nodes are expanded. K-Best algorithm keeps only K nodes at each level. Let j be the level number. The procedure of the algorithm is:

1. Let $j = N$;
2. Calculate the PEDs of all nodes at level j expanded by the former level and choose $\min(K, M_j)$ smallest ones, where M_j is the number of nodes at level j ;
3. Expand the chosen nodes, and $j = j - 1$;
4. If $j \neq 1$, go to step 2; else go to step 5;
5. Choose the smallest PED as the optimal solution.

4. Dynamic K-Best Algorithm

The fixed K-Best SD may bring extra complexity because for some cases the PEDs are more likely to be near the ML solution while for other cases they are low. So a method able to adjust the value of K according to how likely it is for the partial solution to be identical to the

ML solution is proposed. It is called Dynamic K-Best algorithm.

If the noise of the optimal solution is small, the PEDs of the solutions keeps unchanged, the PED difference between the optimal solution and the other solutions will be larger. A method adjusting the number K according to the difference between the PEDs of the optimal solution and the suboptimal solution is feasible.

For some cases where the differences are large, in other words, the partial solution is more likely to approach the ML solution, the value of K will be decreased in order to reduce the number of visiting nodes. Similarly, for other cases when the difference is small, the value of K will be increased to ensure the accuracy. The mathematical description of the algorithm is:

1. Give a set of thresholds $[a, b, c]$ ($a < b < c$) to each level;
2. Calculate the PEDs of the optimal and suboptimal solutions $d_{j_{opt}}$ and $d_{j_{subopt}}$.
3. Let $K = K_1$ if $\Delta d^2 = |d_{j_{opt}}^2 - d_{j_{subopt}}^2| \in [0, a]$;
else $K = K_2$ if $\Delta d^2 \in (a, b]$;
else $K = K_3$ if $\Delta d^2 \in (b, c]$;
else $K = K_4$;
go to step 3;
4. Keep K survivor nodes with the smallest K distances and expand them and $j = j - 1$;
5. If $j \neq 1$, go to step 2;
else go to step 5;
6. Find the solution with the smallest distance as the optimal solution x_{opt} .

5. Simulation Result

In this section, the BER performance and complexity of the Dynamic K-Best SD, compared with the conventional K-Best SD ($K=16$), are assessed. It is assumed that our simulation is based on a 4×4 MIMO system with 16QAM or 64QAM through a flat Rayleigh fading channel, and $[K_1, K_2, K_3, K_4]$ is assigned [8,4,12, 16].

Figures 1 and **2** are drawn when 16QAM is applied. **Figure 1** shows the BER performance of the Dynamic K-Best SD as well as the conventional K-Best SD. It is obvious that in **Figure 1** the interval between the two curves is almost negligible. By selecting appropriate threshold values of each level, the rates of BER increased approximately are 8.94%, 8.06%, 6.44%, 9.09%, 8.96% and 6.25% when SNR (dB) equals 0, 2, 4, 6, 8 and 10 respectively. All of the rates of BER increased by Dynamic K-Best SD are less than 10%, which means practically the deterioration of BER can be ignored.

The computational complexity is characterized by the average number of nodes visited in searching tree in this paper. In **Figure 2**, it is concluded that compared with 404 nodes when using the conventional K-best SD, the

computational complexity is reduced to less than 165 nodes when using Dynamic K-Best SD. To be explicit, the average number of visited nodes is 145.2, 164.3, 149.5, 142.4, 131.0 or 128.2 respectively as SNR (dB) rises from 0 to 10 equidistantly. It implies that on condition of the same BER, the computational complexity of Dynamic K-Best SD is 40% of that of conventional K-Best SD.

Figures 3 and 4 are drawn in 4x4 MIMO system with 64QAM. Correspond with Figures 1 and 2, Figures 3 and 4 show the BER performance and computational complexity separately. And the rates of BER increased

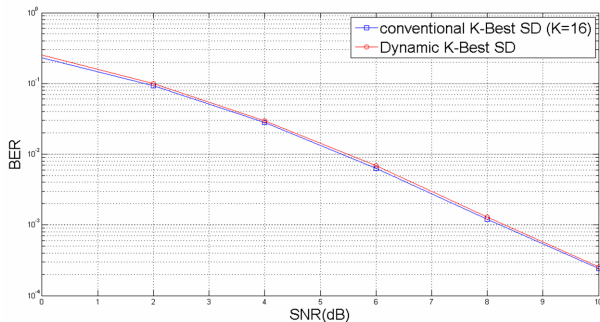


Figure 1. BER Performance Results of conventional KSD and Dynamic KSD in 4x4 MIMO System with 16QAM.

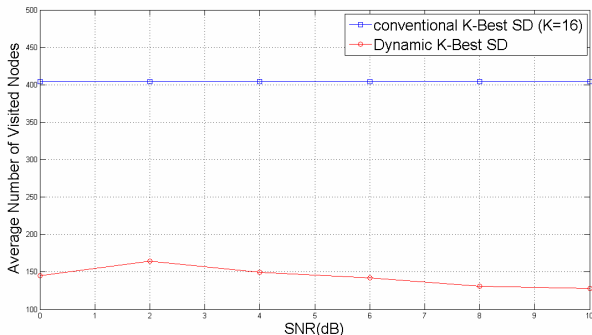


Figure 2. Computation Complexity of conventional KSD and Dynamic KSD in 4x4 MIMO System with 16QAM.

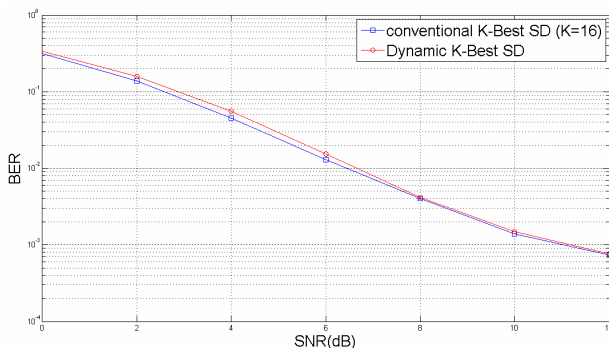


Figure 3. BER Performance Results of conventional KSD and Dynamic KSD in 4x4 MIMO System with 64QAM.

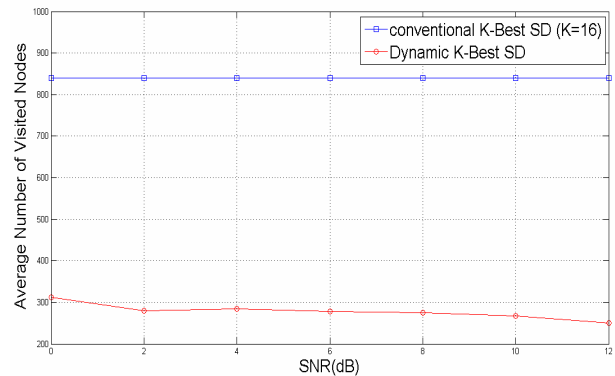


Figure 4. Computation Complexity of conventional KSD and Dynamic KSD in 4x4 MIMO System with 64QAM.

are 6.47%, 15.44%, 21.70%, 17.75%, 8.13%, 2.82% and 2.71% when SNR (dB) equals 0, 2, 4, 6, 8, 10, 12. Meanwhile the average number of visited nodes is reduced to 312.6, 280.3, 284.4, 278.9, 274.8, 267.3 and 250.7. This result proves that our algorithm still works well under 64QAM. A considerable reduction of computational complexity, around 64% on average, is accomplished.

6. Conclusions

In this paper, we have introduced a dynamic K-best algorithm, where the value of K is adjusted based on the difference between the PEDs of the optimal solution and the suboptimal solution. The simulation results above show that by using this Dynamic K-best SD algorithm, computational complexity is significantly reduced at a cost of negligible performance degradation.

7. Acknowledgements

The corresponding author is Xinyu Mao.

REFERENCES

- [1] K. Yu and B. E. Ottersten, "Models for MIMO Propagation Channels: A Review," *Wireless Communications and Mobile Computing*, Vol. 2, No. 7, 2002, pp. 653-666. [doi:10.1002/wcm.78](https://doi.org/10.1002/wcm.78)
- [2] E. Agrell, T. Eriksson, A. Vardy and K. Zeger, "Closest Point Search in Lattices," *IEEE Trans. Inf. Theory*, Vol. 48, No. 8, 2002, pp. 2201-2214. [doi:10.1109/TIT.2002.800499](https://doi.org/10.1109/TIT.2002.800499)
- [3] M. O. Damen, H. El Gamal and G. Caire, "On Maximum-likelihood Detection and the Search for the Closest Lattice Point," *IEEE Trans. Inf. Theory*, Vol. 49, No. 10, 2003, pp. 2389-2402. [doi:10.1109/TIT.2003.817444](https://doi.org/10.1109/TIT.2003.817444)
- [4] K. Wong, C. Tsui, R. S. Cheng and W. Mow, "A VLSI Architecture of a K-Best Lattice Decoding Algorithm for MIMO Channels," in *Proc. IEEE International Symposium on Circuits and Systems*, Vol. 3, May 26-29, 2002,

pp. III-273-III-276.

- [5] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner and H. Bolcklei, "VLSI Implementation of MIMO Detection using the Sphere Decoding Algorithm," *IEEE Journal of Solid State Circuit*, Nov. 2004.
- [6] A J. Paulraj, D. A. Gore, R. U. Nabar, *et al.*, "An Overview of MIMO Communications—A Key to Gigabit Wireless," *Proceedings of IEEE*, 2004, Vol. 92, No. 2, pp. 198-216.
- [7] Pammer, V. Y. Delignon, W. Sawaya and D. Boulinguez, "A Low Complexity Suboptimal MIMO Receiver: The Combined ZF-MLD Algorithm," *IEEE PIMRC'03*, Vol. 3, 2003, 2003, pp. 2271 -2275.
[doi:10.1109/JPROC.2003.821915](https://doi.org/10.1109/JPROC.2003.821915)
- [8] Q. Li and Z. Wang, "Improved K-Best Sphere Decoding Algorithms for MIMO Systems," *International Symposium on Circuits and Systems (ISCAS 2006)*, Island of Kos, Greece. 21-24 May 2006.L.