An Efficient Best-First Derandomization Sampling Algorithm for Lattice Decoding

Yanhua Sun, Xishuang Chen, Ruizhe Yang, and Yanhua Zhang
College of Electronic Information and Control Engineering, Beijing University of Technology, Beijing, China
Email: {sunyanhua, yangruizhe, zhangyh}@bjut.edu.cn

***Abstract*—Although lattice reduction aided decoding improves the decoding performance, it has a performance gap to Maximum Likelihood (ML) decoding increasing with the lattice dimension. In view of this, efficient suboptimal decoding algorithms that can narrow the gap are desirable. In this paper, an efficient Best-First Derandomized Sampling (BFDS) decoding algorithm is proposed to achieve near optimal performance. The existing DS algorithm recently proposed adopts breadth-first search and a probability threshold pruning strategy to generate the candidate lattice point list, without making full use of sampling probabilities. Different from the existing DS algorithm, the cumulative sampling probability, which is the product of sampling probabilities of each sampled integer, is taken into account to generate a list by best-first search strategy, which brings complexity reduction compared to the existing DS algorithm without performance loss. Moreover, to enhance the performance, statistical properties of the cumulative sampling probability are considered to build candidate list instead, which yields better performance. It is shown that the proposed BFDS algorithm has much lower complexity compared to the known DS algorithm without performance loss. In addition, the further improved algorithm considering statistical information outperforms those without considering statistical information.**

***Index Terms*—lattice reduction, derandomization sampling, lattice decoding**

I. Introduction

Multi-Input Multi-Output (MIMO) system with multiple antennas at transmitter and receiver has been adopted in wireless communication systems, which can break through Shannon capacity limit of single-input single-output system and exponentially increase the capacity of communication system. Decoding for MIMO system can be considered as the Closest Vector Problem (CVP) [1]-[3] whose worst-case complexity increases exponentially with the lattice dimension, such as Maximum Likelihood Decoding (MLD) realized by Sphere Decoding (SD) [4]-[7] in MIMO system.

In order to reduce the complexity, suboptimal algorithms for solving CVP, such as Linear Decoding (LD) and Successive Interference Cancellation (SIC) algorithms have been widely adopted, which suffer from inferior performance compared to MLD. To bridge the performance gap, Lattice Reduction (LR) technique [8]-[11] has been proposed to combine with suboptimal decoding algorithms, referred to as Lattice-Reduction-Aided Decoding (LRAD), which achieves full diversity in uncoded MIMO system [12]-[14]. However, LRAD exhibits a widening gap to MLD performance as the dimension increases [15], [16].

To narrow the gap between LRAD and MLD, the Randomized Sampling (RS) decoding has been proposed in [17], [18] recently. It adopts Kleins sampling algorithm [19] to randomly sample lattice points, which applies randomized rounding according to Gaussian distribution instead of standard rounding and selects the best one among all the samples. However, due to the randomization, two defects exist in RS decoding, that is, inevitable sampling repetitions resulting in unnecessary complexity and performance loss resulting from some lattice points with small sampling probabilities in the early levels being neglected. To avoid these two defects, the Derandomized Sampling (DS) decoding has been proposed in [20] to remove the randomization and generate a deterministically sampling process. It admits a tree structure and the final candidate list is generated by traversing the tree with breadth-first search and a probability threshold pruning strategy. However, this method only considers sampling probability information as the threshold pruning rule and does not make full use of this information.

To decrease complexity and increase efficiency, it is necessary to prune the tree nodes that are less likely to lead to the optimal solution and find the list of the most likely to contain the optimal lattice point with the smallest possible complexity. The key to reduce the complexity is to design an efficient search strategy that can find the better candidate list as fast as possible, i.e. with the minimum number of nodes visited. In this paper, an efficient best-first Derandomized Sampling (DS) decoding algorithm referred to as BFDS is proposed to exploit the cumulative sampling probability, the product of sampling probabilities of each sampled integer, to generate a candidate list based on best-first search strategy, which brings complexity reduction compared to the existing DS algorithm without performance loss. Further, statistical properties of the cumulative sampling probability are considered as path metrics to build a candidate list instead, which yields better performance. Setting the initial sample size K and the length of candidate lattice point list L appropriately, the trade-off

Manuscript received February 11, 2015; revised November 30, 2015.

©2016 Int. J. Sig. Process. Syst.
doi: 10.18178/ijsps.4.5.354-361

between performance and complexity of the proposed algorithms can be realized flexibly.

The rest of the paper is organized as follows. Section II introduces the system model and briefly reviews the lattice decoding knowledge. In Section III, the BF search strategy is introduced and BFDS algorithm is proposed based on this search strategy. And then, the BFDS algorithm is further improved. Simulation results are presented and discussed in Section IV. Finally, the paper is concluded in Section V.

Notation: Upper and lower case boldface letters denote matrices and column vectors respectively. Superscript $\bullet^T$, $\bullet^{-1}$ denote transpose and inverse respectively. Superscript $\bullet^C$ denotes that the element is complex. I_{n_r} is the $n_r \times n_r$ identity matrix. $\boldsymbol{b}_{\mathrm{i}}$ indicates the *i*-th column of matrix $\boldsymbol{B}$, $b_{i,j}$ for the (*i*, *j*)-th entry of matrix $\boldsymbol{B}$, and b_{i} the *i*-th entry of vector $\boldsymbol{b}$. The $\Re$ and $\Im$ denote the real and imaginary parts of a complex number. $\| \cdot \|$ denotes the 2-norm. $\lceil x \rfloor$ indicates rounding to the closest integer to *x*, while $\lfloor x \rfloor$ to the closest integer smaller than or equal to *x*.

II. System Model and Lattice Decoding

A. System Model

Consider a MIMO system model with n_t transmit antennas and n_r receive antennas. The received signal vector is denoted by:

$$\boldsymbol{y}^c = \boldsymbol{B}\boldsymbol{x}^c + \boldsymbol{n}^c \tag{1}$$

where $\boldsymbol{B}$, $\boldsymbol{x}^c = \left[x_1^c, x_2^c, \cdots, x_{n_t}^c\right]^T$ and $\boldsymbol{n}^c$ denote the $n_r \times n_t, \left(n_r \geq n_t\right)$ channel matrix, the $n_t \times 1$ transmitted symbol vector, and the noise vector with zero mean and covariance $\sigma_N^2 I_{n_r}$. Let $x_i^c \in \mathbb{C}$, where $\mathbb{C}$ denotes the symbol alphabet of M-QAM modulation and the average transmission power of each antenna is normalized to one. The entries of $\boldsymbol{B}$ are i.i.d. complex Gaussian random variables with zero mean and unit variance. We assume a quasi-static channel environment, i.e., channel is invariant during a block and changes independently from block to block. Moreover, we assume that perfect Channel State Information (CSI) is available at the receiver.

Then, the equivalently $2n_r \times 2n_t$ real-value system is written as:

$$\begin{bmatrix} \Re \boldsymbol{y}^c \\ \Im \boldsymbol{y}^c \end{bmatrix} = \begin{bmatrix} \Re \boldsymbol{B} & -\Im \boldsymbol{B} \\ \Im \boldsymbol{B} & \Re \boldsymbol{B} \end{bmatrix} \begin{bmatrix} \Re \boldsymbol{x}^c \\ \Im \boldsymbol{x}^c \end{bmatrix} + \begin{bmatrix} \Re \boldsymbol{n}^c \\ \Im \boldsymbol{n}^c \end{bmatrix} \tag{2}$$

The real-valued QAM constellations $\mathcal{C}$ can be considered as the shift and scaled version of a consecutive integer set $\mathcal{A}$, i.e. $\mathcal{C} = a\left(\mathcal{A} + [1/2, ..., 1/2]^T\right)$, where the factor a depends on energy normalization. For instance, we have $\mathcal{A} = \left\{-\sqrt{\mathrm{M}}/2, \ldots, \sqrt{\mathrm{M}}/2 - 1\right\}$ for M-QAM modulation.

According to the lattice theory, an *n*-dimensional lattice in the *m*-dimensional Euclidean space $\mathbb{R}^m$ is the set of integer linear combinations of *n* linearly independent vectors $\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_n \in \mathbb{R}^m$ [8], [9]:

$$\mathcal{L} \triangleq \mathcal{L}(\boldsymbol{H}) = \left\{ \sum_{i=1}^{n} x_i \boldsymbol{h}_i \mid x_i \in \mathbb{Z}, i = 1, \ldots n \right\} \tag{3}$$

where $\mathbb{Z}$ is integer set, and $\boldsymbol{H} = \left[\boldsymbol{h}_1 \ldots \boldsymbol{h}_n\right]$ is referred to as a basis of the lattice $\mathcal{L}$. In the matrix form, we have $\mathcal{L} = \left\{\boldsymbol{H}\boldsymbol{x} : \boldsymbol{x} \in \mathbb{Z}^n\right\}$. The matrix $\boldsymbol{H}' = \boldsymbol{H}\boldsymbol{T}$ generates the same lattice as $\boldsymbol{H}$, if and only if the matrix $\boldsymbol{T}$ is unimodular, i.e. $\boldsymbol{T}$ contains only integers and the determinant of $\boldsymbol{T}$ is ± 1.

After scaling and shifting, (2) is simplified as a $n \times m$ real value system as follows:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{4}$$

where $n = 2n_t$, $m = 2n_r$ and $\boldsymbol{H} \in \mathbb{R}^{m \times n}$ can be interpreted as the basis matrix of the decoding lattice. The data vector $\boldsymbol{x}$ belongs to a finite subset $\mathcal{A}^n$.

B. Lattice Reduction Aided Decoding

Given the model in (4), the ML decoding is computed as follows:

$$\hat{\boldsymbol{x}} = \underset{\boldsymbol{x} \in \mathcal{A}^n}{\arg\min} \left\| \boldsymbol{y} - \boldsymbol{H}\boldsymbol{x} \right\|^2 \tag{5}$$

which corresponds to solving a Closest Vector Problem (CVP) in the lattice $\mathcal{L}(\boldsymbol{H})$. ML decoding can be realized by the sphere decoding, whose expected exponential complexity makes it difficult to be widely used. So, the LRAD is often preferred due to its acceptable complexity.

Lattice reduction technology can transform the basis $\boldsymbol{H}$ into a basis consisting of roughly orthogonal vectors $\boldsymbol{H}' = \boldsymbol{H}\boldsymbol{T}$, where $\boldsymbol{T}$ is an unimodular matrix. Therefore, we can get the equivalent system model:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} = \boldsymbol{H}'\boldsymbol{z} + \boldsymbol{n}, \boldsymbol{z} = \boldsymbol{T}^{-1}\boldsymbol{x} \tag{6}$$

Then, based on the reduced basis, conventional decoders (ZF or SIC) are adopted to obtain the estimate $\hat{\boldsymbol{z}}$, which is then transformed back into the original signal space by $\hat{\boldsymbol{x}} = \boldsymbol{T}\hat{\boldsymbol{z}}$. In the LR-aided SIC decoding, after QR-decomposition of the reduced channel matrix $\boldsymbol{H}' = \boldsymbol{Q}\boldsymbol{R}$, (6) is rewritten as:

$$\boldsymbol{y}' = \boldsymbol{Q}^{\mathrm{T}}\boldsymbol{y} = \boldsymbol{R}\boldsymbol{z} + \boldsymbol{n}' \tag{7}$$

At the each decoding level $i = n, n-1, \ldots, 1$, the pre-detection signal $\tilde{z}_i$ is calculated as:

$$\tilde{z}_i = \frac{y_i' - \sum_{j=i+1}^{n} r_{i,j}\hat{z}_j}{r_{i,i}} \tag{8}$$

where the decision $\hat{z}_i$ is obtained by rounding $\tilde{z}_i$ to the nearest integer as $\hat{z}_i = \lceil \tilde{z}_i \rfloor$. The decoding procedure begins at the level n and continues until the level 1 is detected.

C. Sampling Decoding

Different from the LR-aided SIC decoding, the pre-detection signal $\tilde{z}_i$ is not rounded to the closest integer but to the integers around $\tilde{z}_i$ randomly according to the following discrete Gaussian distribution [17], [19].

$$P\left(\hat{z}_i^j\right) = e^{-c_i\left(\tilde{z}_i - \hat{z}_i^j\right)^2} \Big/ s_i \,, \qquad s_i = \sum_{\hat{z}_i^j = -\infty}^{\infty} e^{-c_i\left(\tilde{z}_i - \hat{z}_i^j\right)^2} \tag{9}$$

where $c_i = A r_{i,i}^2$, $\hat{z}_i^j$ represents the sampled integer around $\tilde{z}_i$ and j is the index. Specifically, $\hat{z}_i^1$ denotes the closest integer to $\tilde{z}_i$ while $\hat{z}_i^2$ denotes the second closest integer and so on. The parameter A which affects the variance of sampling probabilities is given as $A = \log \rho \big/ \min_i r_{i,i}^2$, where the parameter $\rho > 1$ related with the sample size K set initially satisfies $K = \frac{1}{2}(e\rho)^{2n/\rho}$ [17].

Generally, we can approximate distribution (9) by $2N$-point discrete distribution [17].

$$P\left(\hat{z}_i^j\right) = e^{-c_i\left(\tilde{z}_i - \hat{z}_i^j\right)^2} \Big/ s_i' \,, \qquad s_i' = \sum_{\hat{z}_i^j \in \mathbb{N}} e^{-c_i\left(\tilde{z}_i - \hat{z}_i^j\right)^2} \tag{10}$$

$$N = \left\{ \lfloor \tilde{z}_i \rfloor - N + 1, \ldots, \lfloor \tilde{z}_i \rfloor, \ldots, \lfloor \tilde{z}_i \rfloor + N \right\}$$

In fact, 3-point approximation is sufficient as the probability in the central 3 points is almost one. In this paper, we choose N=2.

III. Best-First DS Decoding

Reference [15] demonstrates that LRAD exhibits a widening gap to MLD performance as the dimension increases. In order to narrow the gap, the DS decoding is proposed recently to generate a deterministically sampling process by breadth-first search and a probability threshold pruning strategy. However, this scheme only considers sampling probability information as the threshold pruning rule and does not make full use of this information. In this section, to reduce complexity and increase efficiency, a new Best-First Derandomized Sampling (BFDS) decoding algorithm is proposed to make full use of sampling probability information, which exploits the cumulative sampling probability, the product of sampling probabilities of each sampled integer, to generate a candidate list based on best-first search strategy. Moreover, to enhance the performance, the BFDS decoding algorithm is improved by considering statistical properties of the cumulative sampling probability.

A. BFDS Decoding Algorithm

As mentioned before, DS decoding admits a tree structure as shown in Fig. 1 that has n+1 layers of nodes with a virtual root node at layer n+1.

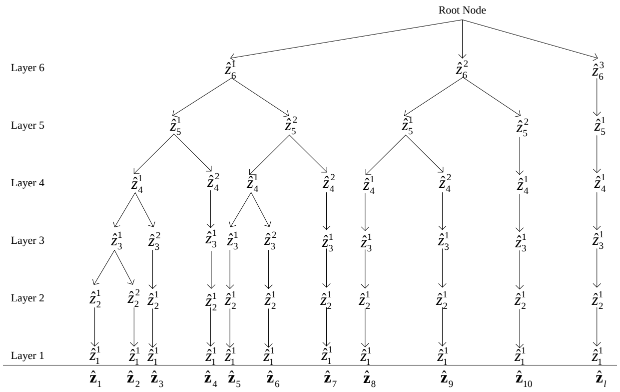

Figure 1. An illustrative example of the tree structure in the derandomized sampling decoding for a 3×3 system. $\hat{\boldsymbol{z}}_i, (i = 1, 2, \ldots, l)$ denote the candidate lattice points, where i is the index of the candidate lattice point.

The cumulative sampling probability of a node in layer $i\left(i = n, n-1, \ldots, 1\right)$, which is the product of the sampling probabilities in the path generated by traversing the tree from the root node to this node, is defined as its cost. In general, the cumulative sampling probability $CSP_i\left(\hat{z}_i\right)$ of a node $\hat{z}_i$ in layer i can be written as:

$$\begin{aligned}
CSP_i\left(\hat{z}_i\right) &= \prod_{k=i}^{n} P\left(\hat{z}_k\right) = \prod_{k=i}^{n} e^{-c_k\left(\tilde{z}_k - \hat{z}_k\right)^2} \Big/ s_k' \\
&= \prod_{k=i}^{n} e^{-A * r_{k,k}^2 \left[\left(y_k' - \sum_{j=k+1}^{n} r_{k,j}\hat{z}_j\right) \big/ r_{k,k} - \hat{z}_k\right]^2} \Big/ s_k' \\
&= \prod_{k=i}^{n} e^{-A\left(y_k' - \sum_{j=k+1}^{n} r_{k,j}\hat{z}_j - r_{k,k}\hat{z}_k\right)^2} \Big/ s_k' \\
&= \prod_{k=i}^{n} e^{-A\left(y_k' - \sum_{j=k}^{n} r_{k,j}\hat{z}_j\right)^2} \Big/ s_k' = e^{-A * \sum_{k=i}^{n}\left(y_k' - \sum_{j=k}^{n} r_{k,j}\hat{z}_j\right)^2} \Big/ \prod_{k=i}^{n} s_k'
\end{aligned} \tag{11}$$

Specially, the cumulative sampling probability of a leaf node in layer 1 is named as full path cumulative sampling probability, which is denoted as follows:

$$\begin{aligned}
CSP_1\left(\hat{z}_1\right) &= \prod_{k=1}^{n} P\left(\hat{z}_k\right) \\
&= e^{-A * \sum_{k=1}^{n}\left(y_k' - \sum_{j=k}^{n} r_{k,j}\hat{z}_j\right)^2} \Big/ \prod_{k=1}^{n} s_k' \\
&= e^{-A * \left\|\boldsymbol{y}' - \boldsymbol{R}\hat{\boldsymbol{z}}\right\|^2} \Big/ \prod_{k=1}^{n} s_k'
\end{aligned} \tag{12}$$

where $\left\|\boldsymbol{y}' - \boldsymbol{R}\hat{\boldsymbol{z}}\right\|^2$ represents the distance between the candidate lattice point and the received signal. And, the

©2016 Int. J. Sig. Process. Syst.

smaller the distance is, the larger the CSP_1 becomes. MLD can be considered as the CVP problem which is to find the lattice point that is the closest to the received signal. Therefore, finding the leaf node that has minimum distance is equivalent to finding the one with maximum cumulative sampling probability.

According to the above analysis, a new and efficient Best-First (BF) search strategy based on cumulative sampling probability is proposed, which adopts the cumulative sampling probability as path metric to explore the tree. Specific details are as follows: By traversing the tree from the root node to the leaf node, the algorithm always gives the expansion priority to the node with the largest cumulative sampling probability in the node list. Thus, the leaf node with the largest full path cumulative sampling probability can be obtained firstly. Then, add this lattice point into the candidate lattice point list. And, get another candidate lattice point by finding the leaf node with the second largest full path cumulative sampling probability, and so on. At last, some candidate lattice points with larger full path cumulative sampling probability are obtained. The number of the candidate lattice points is set initially according to the trade-off between performance and complexity.

The proposed BFDS algorithm can be described as Algorithm 1 in details (for level index $i = n, n-1, ..., 1$):

Algorithm 1 BFDS algorithm

Initialization: a node list Φ that contains the virtual root node only, an empty candidate lattice point list Ψ with the maximum list size *L*, and a sample size *K*.

Input: $\boldsymbol{y}'$, n, $\boldsymbol{R}$, $\boldsymbol{T}$, K, L

Output: $\hat{\boldsymbol{x}}$

$K = \frac{1}{2}(e\rho)^{2n/\rho}$;
$A = \log \rho / \min_i r_{i,i}^2$;
while (Ψ is not overflow)
    i = current decoding layer
    $\tilde{z}_i = \frac{y_i' - \sum_{j=i+1}^{n} r_{i,j}\hat{z}_j}{r_{i,i}}$;
    $c_i = A * r_{i,i}^2$;
    for j = 1:4
        $P\left(\hat{z}_i^j\right) = e^{-c_i\left(\tilde{z}_i - \hat{z}_i^j\right)^2} / s_i'$;
        $E\left(\hat{z}_i^j\right)$ = round ($kP\left(\hat{z}_i^j\right)$);
        if $E\left(\hat{z}_i^j\right) < 1$
            $\hat{z}_i^j$ is ignored;
        else
            let $\hat{z}_i = \hat{z}_i^j$ as a children node;
            update sample size as $k' = kP\left(\hat{z}_i^j\right)$
        end if
        put the children nodes into Φ and remove their parent node from Φ;
    end for
    Select the best with the largest k' from Φ;
    while (the best node is a leaf node)
        Take the best node as a candidate lattice point and add it into Ψ;
        If Ψ is overflow
            break;
        end if
        Remove the best node from Φ and select another best node from Φ
    end
    update the decoding layer;
end
for j = 1:*L*
    $\hat{\boldsymbol{x}}_j = \boldsymbol{T} * \hat{\boldsymbol{z}}_j$;
end for
$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}_j} \left\| \boldsymbol{y} - \boldsymbol{H}\hat{\boldsymbol{x}}_j \right\|^2$

In Algorithm 1, one thing to note is that the update sample size $kP\left(\hat{z}_i^j\right) = K * \prod_{j=i}^{n} P\left(\hat{z}_j\right)$ is just the product of cumulative sampling probability and constant *K*. That is, the cumulative sampling probability is a by-product in updating the sampled size *k*. So, we actually need not calculate the cumulative sampling probabilities of nodes, but only update sample size *k* and select the node with the largest *k* from Φ.

B. BFDS Complexity Analysis

As the illustrative example of the BFDS decoding shown in Fig. 2, BFDS decoding only generates $L < l(l \le K)$ candidate lattice points with relative larger full path cumulative sampling probabilities. Compared to Fig. 1, we can observe that some nodes in the BFDS decoding are not expanded to the leaf node, which results in fewer nodes visited. This is because the best-first search is adopted to find the candidate lattice points that are more likely to be the optimal lattice point in the list obtained in Fig. 1.

The computational complexity of the known DS algorithm in [20] is $K \cdot O(n^2)$, where $O(n^2)$ is the computational complexity generated by calculation of the sampling probability in case for $K = 1$. Therefore the computational complexity of BFDS decoding can be denoted by $L \cdot O(n^2)$, which is lower than the known DS algorithm [20].

By varying parameter *K* and *L*, the decoder enjoys a flexible trade-off between performance and complexity. Let *K* maintain a constant, with the increment of *L* the performance of BFDS decoding improves gradually and finally obtains the same performance with the known DS decoding. Of course, the complexity will also increases gradually with the increment of *L* since more candidate lattice points will be sampled. But, the BFDS algorithm can achieve the same performance with fewer number of nodes visited and a lower complexity compared to the known DS algorithm. As *K* increases, the performance of

©2016 Int. J. Sig. Process. Syst.

BFDS decoding becomes getting close to the MLD performance. And the near-optimum performance can be achieved by a moderate size *K*. Thus, the proposed decoder can enjoy a flexible performance between SIC and near-ML by adjusting *K* and *L*.

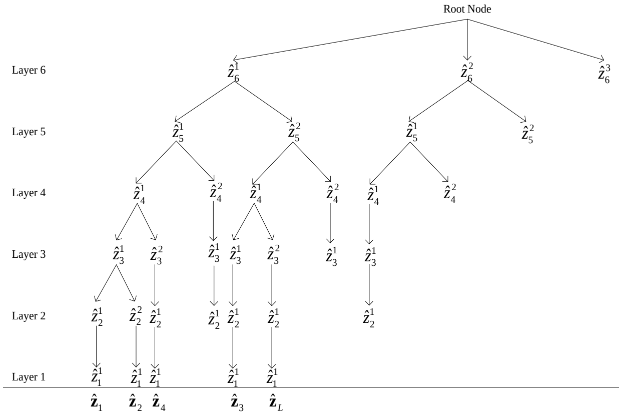

Figure 2. An illustrative example of the BFDS decoding for a 3×3 system. $\hat{\mathbf{z}}_i, (i=1,2,...,L)$ denote the candidate lattice points, where *i* is the index of the candidate lattice point with the *i*-th largest full path cumulative sampling probability.

C. Improved BFDS Decoding Algorithm

The BFDS algorithm proposed in the previous section is an efficient lattice decoding algorithm, which can bring complexity reduction compared to the known DS algorithm. However, path metric based on the cumulative sampling probability ignores the fact that nodes in the node list Φ may come from different layers and thus their cumulative sampling probabilities do not necessarily reflect the "goodness" of these nodes. For example, a node A in a lower layer whose cumulative sampling probability is smaller than that of node B in an upper layer may likely to be a better node. Based on this, the BFDS algorithm is improved by exploiting statistical properties of the cumulative sampling probability, which takes the layer information of the node into account to make nodes from different layers be explored on a fairer basis.

In the improved BFDS algorithm, natural logarithm of the cumulative sampling probability of each node at layer $i = n, n-1, \ldots, 1$ can be written as:

$$G_i = \ln\left(CSP_i\right) = -A * D_i - S_i,$$
$$D_i = \sum_{k=i}^{n}\left|y'_k - \sum_{j=k}^{n} r_{k,j}\hat{z}_j\right|^2, S_i = \sum_{k=i}^{n}\ln\left(s'_k\right) \tag{13}$$

And the cumulative distribution function (cdf) of cumulative sampling probability is considered as follows:

$$\begin{aligned}\mathrm{F}_{G_i}(g;i) &= P\{G_i \le g\} = P\{-A*D_i - S_i \le g\}\\ &= 1 - P\left\{D_i \le -\frac{g+S_i}{A}\right\}\\ &= 1 - P\left\{\frac{2D_i}{\sigma_N^2} \le -\frac{2(g+S_i)}{A*\sigma_N^2}\right\}\\ &= 1 - F_{D_i}(x;i)\end{aligned} \tag{14}$$

where $2D_i/\sigma_N^2$ follows the chi-square distribution with $(n-i+1)$ degrees of freedom, so its cdf is calculated as [21]:

$$\mathrm{F}_{D_i}(x;i) = \frac{\Upsilon\left(\frac{n-i+1}{2}, \frac{x}{2}\right)}{\Gamma\left(\frac{n-i+1}{2}\right)}, x \ge 0 \tag{15}$$

where $\Gamma(\cdot)$ is the Gamma function and $\Upsilon(\cdot)$ is the incomplete Gamma function. The improved BFDS algorithm expands the node which has the maximum cdf calculated by (14) instead of the maximum cumulative sampling probability denoted by (11).

According to the properties of cdf, in sorting two nodes from different layers where the node in a upper layer with a larger cumulative sampling probability, i.e., for $i<j$, $g_i<g_j$, sorting rule based on cdf may or may not give priority to this node, i.e., the inequality $F_{G_i}(g_i;i) < F_{G_j}(g_j;j)$ may or may not hold.

In fact, except the path metric adopted is different in these two algorithms, the other steps are similar. In the improved BFDS algorithm, the nodes are selected with the largest cumulative distribution function instead of the largest cumulative sampling probability and we need to calculate the natural logarithm of the cumulative sampling probability according to (13). Then input the calculated natural logarithm into (14) in place of g to obtain the cdf of the children node.

IV. Simulation Results

In this section, the performance and complexity of the two proposed algorithms are evaluated in a 10×10 MIMO system with 64-QAM by simulation. In this paper, we use the number of nodes visited, i.e., the number of nodes preserved in the tree, to denote the actual complexity involved. Let Eb denotes the average power per bit at the receiver, for the SNR, we have $E_b/N_0 = n_r/\left(\log_2 M * \sigma_N^2\right)$, where the modulation level *M*=64 for 64-QAM, σ_N^2 is the noise power.

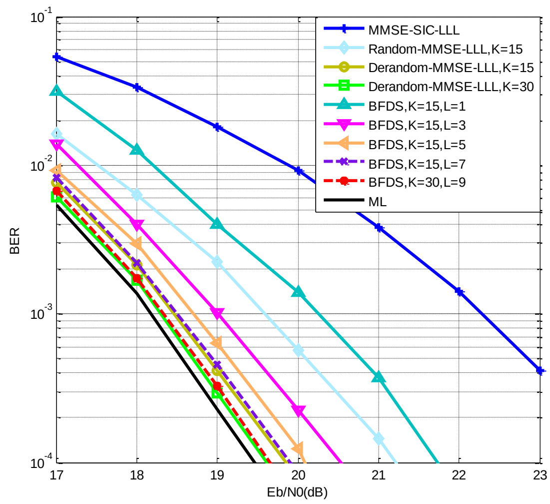

Figure 3. Comparison of bit error rate of different algorithms.

Fig. 3 shows the Bit Error Rate (BER) of the BFDS decoding algorithm compared with other decoding algorithms. Obviously, under the help of LLL reduction ($\delta = 0.99$), all the randomized and derandomized sampling decoding algorithms get considerable gains over the LR-aided SIC, even in case for the BFDS decoding algorithm with K=15, L=1. With the increment of L, the BFDS decoding performance improves gradually. Specially, the BFDS decoding algorithm with K=15, L=7 and K=30, L=9 achieve the same performance as the known DS algorithm with K=15 and K=30 respectively. With the increment of K, the BFDS decoding performance gradually approaches the ML performance.

Fig. 4 shows the complexity comparison of the BFDS decoding algorithm with the known DS decoding algorithm. From Fig. 4, we observe that with the increment of L the average number of nodes visited in BFDS algorithm increases gradually, but the BFDS algorithm can achieve the same performance as the known DS decoding with fewer number of nodes visited. For example, the BFDS algorithm with K=15, L=7 obtains the same performance as the DS algorithm with K=15, but having fewer average number of nodes visited. Moreover, this advantage will become more and more apparent as K increases. So, the BFDS algorithm can finally achieve the near-optimum performance with much lower complexity compared to the known DS decoding algorithm.

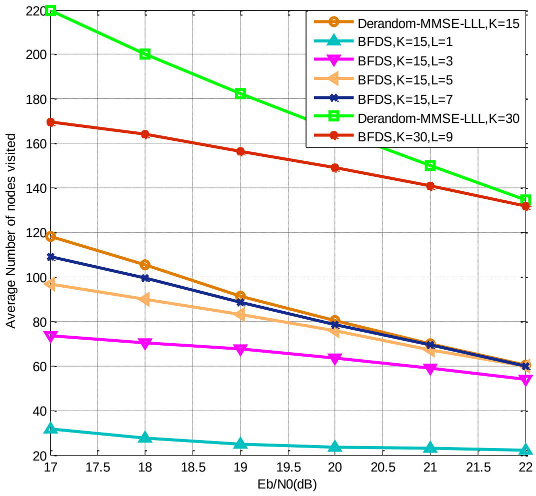

Figure 4. Average number of NOdes visited of different algorithms.

In Fig. 5 and Fig. 6, we respectively compare the error performance and the complexity of the improved BFDS decoding to the BFDS decoding algorithm with K=15, L=1, 3, 5. It is seen that with the same L, the improved BFDS decoding algorithm exhibits significantly better error performances than the BFDS decoding algorithm at the cost of a little increased complexity, such as L=3, but when L=1 the complexity is also reduced. However, when the improved BFDS decoding algorithm with L=3 and the BFDS decoding algorithm with L=5 are compared, the improved BFDS decoding algorithm can achieve a slightly better error performance than the BFDS decoding algorithm with even lower complexity.

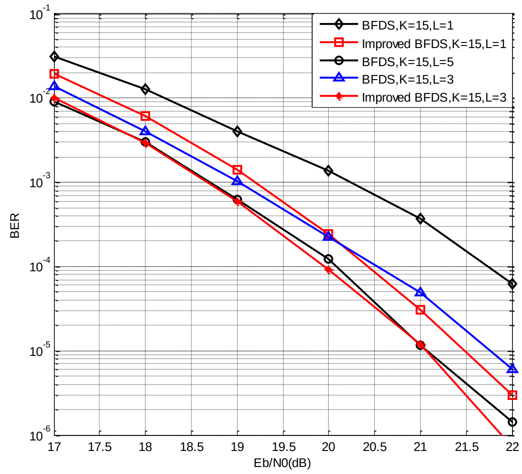

Figure 5. Comparison of BER of different algorithms with K=15.

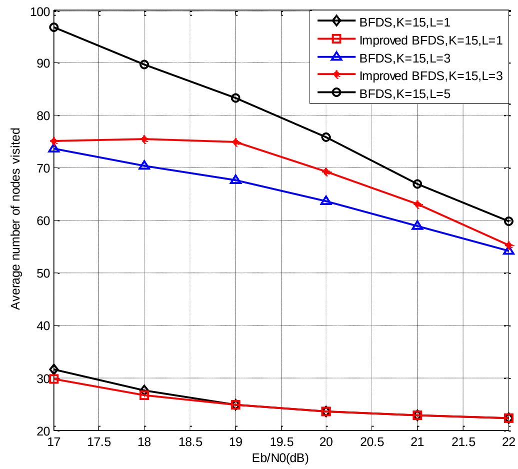

Figure 6. Average number of nodes visited of different algorithms with K=15.

Fig. 7 and Fig. 8 display the error performance and the complexity comparison of the improved BFDS decoding algorithm to the BFDS decoding algorithm with K=30 and L=1, respectively. Similar to the case for K=15 and L=1 shown in Fig. 5 and Fig. 6, both the error performance and the complexity are improved in the improved BFDS decoding algorithm.

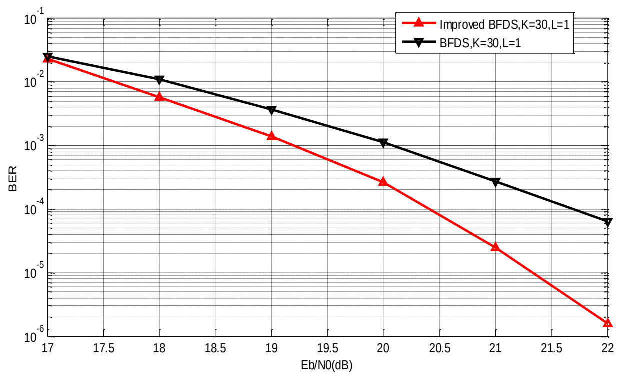

Figure 7. Comparison of bit error rate of different algorithms with K=30.

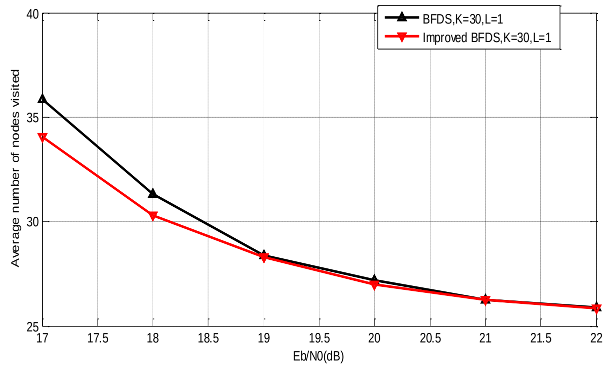

Figure 8. Average number of nodes visited of different algorithms with *K*=30.

V. Conclusions

In this paper, to reduce complexity, a new Best-First Derandomized Sampling (BFDS) decoding algorithm is proposed to make full use of sampling probability information to generate a candidate list by best-first search and a probability threshold pruning strategy. Moreover, to enhance the performance, the BFDS algorithm is improved by exploring the nodes in the node list based on the cumulative distribution function of cumulative sampling probability instead of cumulative sampling probability itself, which takes both the cumulative sampling probability and the layer information of the nodes into account. The simulations show that the BFDS algorithm has much lower complexity compared to the known DS algorithm without performance loss. By varying *K* and *L*, the decoder can enjoy a flexible trade-off between performance and complexity. And the further improved algorithm considering statistical information outperforms those without considering statistical information in a lower complexity.

Acknowledgment

This work is jointly supported by the National Natural Science Foundation of China under Grant 61201198 and 61101113, Beijing Natural Science Foundation of China under Grant 4132015, 4132007 and 4132019.

References

[1] L. Babai, “On Lovasz lattice reduction and the nearest lattice point problem,” *Combinatorica*, vol. 6, no. 1, pp. 1-13, 1986.

[2] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, “Closest point search in lattices,” *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201-2214, 2002.

[3] D. Micciancio and P. Voulgaris, “A deterministic single exponential time algorithm for most lattice problems based on Voronoi cell computations,” *SIAM Journal on Computing*, vol. 42, no. 3, pp. 1364-1391, 2013.

[4] B. Hochwald and S. T. Brink, “Achieving near-capacity on a multipleantenna channel,” *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389-399, 2003.

[5] J. Lee, B. Shim, and I. Kang, ”Soft-Input soft-output list sphere detection with a probabilistic radius tightening,” *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2848-2857, 2012.

[6] J. Choi, Y. Hong, and J. Yuan, “An approximate MAP based iterative receiver or MIMO channels using a modified sphere detection,” *IEEE Trans. Wireless Commun.*, vol. 5, no. 8, pp. 2199-2126, 2006.

[7] T. Cui, S. Han, and C. Tellambura, ”Probability-Distribution-Based node pruning for sphere decoding,” *IEEE Trans. Vechicular Technology*, vol. 62, no. 4, pp. 1586-1596, 2013.

[8] Y. H. Gan, C. Ling, and W. H. Mow, “Complex lattice reduction algorithm for low-complexity full-diversity MIMO detection,” *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2701-2710, 2009.

[9] L. Luzzi, G. B. Othman, and J. Belfiore, “Augmented lattice reduction for MIMO decoding,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 9, pp. 2853-2859, 2010.

[10] C. Ling, W. Mow, and N. Howgrave-Graham, “Reduced and fixed complexity variants of the LLL algorithm for communication,” *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 1040-1050, 2013.

[11] Z. Wang and X. Ma, “Low-Complexity soft-output decoding with latticereduction-aided detectors,” *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2621-2629, 2010.

[12] M. Taherzadeh, A. Mobasher, and A. K. Khandani, “LLL reduction achieves the receive diversity in MIMO decoding,” *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4801-4805, 2007.

[13] T. W. Tang, M. K. Chen, and H. F. Lu, “Improving the DMT performance for MIMO communication with linear receivers,” *IEEE Trans. Vechicular. Technology*, vol. 62, no. 3, pp. 1189-1200, 2013.

[14] J. Jalden and P. Elia, “DMT optimality of LR-aided linear decoders for a general class of channels, lattice designs, and system models,” *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4765-4780, 2010.

[15] C. Ling, “On the proximity factors of lattice reduction-aided decoding,” *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2795-2808, 2011.

[16] A. H. Mehana and A. Nosratinia, “Performance of linear receiver in frequency-selective MIMO channels,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2697-2705, 2013.

[17] S. Liu, C. Ling, and D. Stehle, ”Decoding by sampling: A randomized lattice algorithm for bounded distance decoding,” *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 5933-5945, 2011.

[18] L. Bai and J. Choi, “Lattice reduction-based MIMO iterative receiver using randomized sampling,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 1-11, 2013.

[19] P. Klein, “Finding the closest lattice vector when its unusually close,” in *Proc. Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, 2013, pp. 937-941.

[20] Z. Wang, S. Liu, and C. Ling, “Decoding by sampling II: Derandomization and soft-output decoding,” *IEEE Trans. Commun.*, vol. 61, no. 11, pp. 1-10, 2013.

[21] B. Shim and I. Kang, “Sphere decoding with a probabilistic tree pruning,” *IEEE Trans. Signal Process*, vol. 56, no. 10, pp. 4867-4878, 2008.

Yanhua Sun received a M.S. degree in signal processing at China University of Petroleum, Beijing, in 2003 and a Ph.D. degree in Communication at the University of Posts and Telecommunications in Beijing in 2006. From 2006, she was a lecturer in Beijing University of Technology. At present, she is in charge of the National Natural Science Foundation of China under Grant 61201198 and Beijing Natural Science Foundation of China under Grant 4132015. Her research interests focus on signal detection, network coding and multiple antenna technology.

Xishuang Chen received the B.E. degree in communications engineering from Beijing University of Technology, Beijing, China, in 2012. He is currently a M.S. candidate at Beijing University of Technology. His research interests include network coding and MIMO system.

Ruizhe Yang received a M.S. degree in Communication in 2006 and a Ph.D. degree in 2009 at the University of Posts and Telecommunications in Beijing. From 2009, she was a lecturer in Beijing University of Technology. Her research interests focus on massive MIMO, channel estimation and resource management.

Yanhua Zhang received the B.E. degree from Xi'an University of Technology, Xi'an, China, in 1982, and the M.S. degree from Lanzhou University, Lanzhou, China, in 1988. From 1982 to 1990, he was in Jiuquan Satellite Launch Center (JSLC), Jiuquan, China. During the 1990s, he is a visiting professor at Concordia University, Montreal, Canada. He joined Beijing University of Technology, Beijing, China, in 1997, where he is currently a professor. His research interests include QoS-aware networking, channel estimation, radio resource management in wireless communications systems.

He served as the Technical Program Committee (TPC) Co-Chair of the IEEE ICCC-GMCN'2013, and TPC member of numerous conferences. He was also the principle investigator of projects of the National High-Tech R&D Program of China (863 Program), National Natural Science Foundation of China, *et al*.