Derandomized Sampling Algorithm for Lattice Decoding

Zheng Wang and Cong Ling
Department of Electrical and Electronic Engineering
Imperial College London
London, SW7 2AZ, United Kingdom
Email: z.wang10, c.ling@imperial.ac.uk

***Abstract*—The sampling decoding algorithm randomly samples lattice points and selects the closest one from the candidate list. Although it achieves a remarkable performance gain with polynomial complexity, there are two inherent issues due to random sampling, namely, repetition and missing of certain lattice points. To address these issues, a derandomized algorithm of sampling decoding is proposed with further performance improvement and complexity reduction. Given the sample size K, candidates are deterministically sampled if their probabilities P satisfy the threshold $PK \geq \frac{1}{2}$. By varying K, the decoder with low complexity enjoys a flexible performance between successive interference cancelation (SIC) and maximum-likelihood (ML) decoding.**

***Index Terms*—Lattice decoding, lattice reduction, derandomized algorithms, near-ML detection.**

I. Introduction

As one of the core problems of lattices, the closest vector problem (CVP) has wide applications in number theory, cryptography, and communications. In [1], lattice reduction was introduced to solve CVP approximately. It greatly improves the performance of suboptimal decoding schemes such as successive interference cancelation (SIC). Since then, a number of improved decoding schemes based on lattice reduction have been proposed [2]–[5]. In multiple-input multiple-output (MIMO) communications, it has been shown in [6] that the minimum mean-square error (MMSE) decoding based on lattice reduction achieves the optimal diversity and multiplexing trade-off. However, the performance gap between maximum-likelihood (ML) and lattice-reduction-aided decoding is still substantial especially in high dimensions [7].

Very recently, randomized sampling decoding has been proposed to narrow the gap between lattice-reduction-aided decoding and sphere decoding [8]. As a randomized version of SIC, it applies Klein's sampling technique [9] to randomly sample lattice points from a Gaussian-like distribution and chooses the closest one among all the samples. Unfortunately, because of randomization, there are two inherent issues in it. One is the inevitable repetitions in the sampling process leading to unnecessary complexity, while the other one is the performance loss since some lattice points with small probabilities in early decoding levels can be missed. Although Klein mentioned a deterministic algorithm very briefly in [9], it does not seem to allow for an efficient implementation.

In this paper, to overcome the two problems caused by randomization, we propose a new kind of decoding algorithm referred to as derandomized sampling decoding. With a sample size K set initially, candidate values are sampled deterministically based on a threshold we define. As randomization is removed, derandomized sampling decoding shows considerable improvement in both performance and complexity. Meanwhile, a lower bound on the total probability of the lattice points sampled by the proposed algorithm is derived. By increasing K to improve that lower bound, the gap to ML performance can be decreased.

This paper is organized as follows: Section II introduces the system model and briefly reviews the randomized sampling decoding. In Section III, the proposed derandomized sampling decoding algorithm is introduced and analysis of its performance is given. Section IV evaluates its performance and complexity by simulations. Finally, conclusions are presented in Section V.

II. Sampling Decoding

Consider the decoding of an $n \times n$ real-valued system. The extension to the complex-valued system is straightforward [8]. Let $\mathbf{x}$ denote the transmitted signal taken from a constellation $\mathcal{X}^n \subseteq \mathbb{Z}^n$. The corresponding received signal $\mathbf{y}$ is given by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{1}$$

where $\mathbf{H}$ is an $n \times n$ full column-rank matrix of channel coefficients and $\mathbf{n}$ is the noise vector with zero mean and variance σ^2.

Given the model in (1), the ML decoding is as follows:

$$\widehat{\mathbf{x}} = \arg\min_{\mathbf{x} \in \mathcal{X}^n} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \tag{2}$$

where $\|\cdot\|$ denotes the Euclidean norm. Vector $\mathbf{Hx}$ can be viewed as a lattice point and the (possibly relaxed) ML decoding corresponds to solving the CVP in a lattice. Due to the exponential complexity of sphere decoding, lattice-reduction-aided decoding is often preferred for its acceptable complexity.

In SIC, after QR-decomposition of the channel matrix $\mathbf{H} = \mathbf{QR}$, the system model in (1) becomes

$$\mathbf{y}' = \mathbf{Q}^T\mathbf{y} = \mathbf{R}\mathbf{x} + \mathbf{n}' \tag{3}$$

978-1-4673-0223-4/12/$31.00 ©2012 IEEE

sample 1	sample 2	sample 3	$\cdots$	sample K
$R(\widetilde{x}_n)$	$R(\widetilde{x}_n)$	$R(\widetilde{x}_n)$	$\cdots$	$R(\widetilde{x}_n)$
$\downarrow$	$\downarrow$	$\downarrow$	$\vdots$	$\downarrow$
$R(\widetilde{x}_{n-1})$	$R(\widetilde{x}_{n-1})$	$R(\widetilde{x}_{n-1})$	$\cdots$	$R(\widetilde{x}_{n-1})$
$\downarrow$	$\downarrow$	$\downarrow$	$\vdots$	$\downarrow$
$R(\widetilde{x}_{n-2})$	$R(\widetilde{x}_{n-2})$	$R(\widetilde{x}_{n-2})$	$\cdots$	$R(\widetilde{x}_{n-2})$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$R(\widetilde{x}_1)$	$R(\widetilde{x}_1)$	$R(\widetilde{x}_1)$	$\cdots$	$R(\widetilde{x}_1)$
$\widehat{\mathbf{x}}_1$	$\widehat{\mathbf{x}}_2$	$\widehat{\mathbf{x}}_3$	$\cdots$	$\widehat{\mathbf{x}}_K$

Fig. 1. Sampling procedure in randomized sampling decoding.

where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{R}$ is an upper triangular matrix. At each decoding level, the pre-detection signal $\widetilde{x}_i$ $(1 \le i \le n)$ is calculated as

$$\widetilde{x}_i = \frac{y_i' - \sum_{j=i+1}^{n} r_{i,j}\widehat{x}_j}{r_{i,i}} \tag{4}$$

where the decision $\widehat{x}_i$ $(1 \le i \le n)$ is obtained by rounding $\widetilde{x}_i$ to the nearest integer as

$$\widehat{x}_i = \lceil \widetilde{x}_i \rfloor. \tag{5}$$

Different from SIC, in randomized sampling decoding [8], $\widehat{x}_i$ is generated randomly:

$$\widehat{x}_i = R(\widetilde{x}_i) \tag{6}$$

where function $R(\widetilde{x}_i)$ denotes the random rounding. According to Klein's sampling algorithm [9], probabilities of $R(\widetilde{x}_i)$ returning integer $\widehat{x}_i^j$ are calculated from the following discrete Gaussian distribution

$$\begin{aligned} P(\widehat{x}_i = \widehat{x}_i^j) &= e^{-c_i(\widetilde{x}_i - \widehat{x}_i^j)^2}/s, \\ s = s(c_i) &= \sum_{\widehat{x}_i^j = -\infty}^{\infty} e^{-c_i(\widetilde{x}_i - \widehat{x}_i^j)^2}, \end{aligned} \tag{7}$$

where j is the index of integers around $\widetilde{x}_i$, $c_i = Ar_{i,i}^2$, $A = \log\rho / \min_i r_{i,i}^2$ for a parameter $\rho > 1$ related with K as [8]

$$K = (e\rho)^{2n/\rho}. \tag{8}$$

It has been demonstrated in [8] that given $\mathbf{y}$, the probability of a vector $\hat{\mathbf{x}}$ being sampled is lower bounded by

$$P(\hat{\mathbf{x}}) \ge \frac{1}{\prod_{i=1}^{n} s(Ar_{i,i}^2)} e^{-A\|\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}\|^2}. \tag{9}$$

The decisions $\widehat{x}_i$'s are generated level by level, and a candidate lattice point $\widehat{\mathbf{x}}$ is obtained if all the entries are generated. By repeating this sampling procedure for K times, a candidate list of K lattice points is obtained as shown in Figure 1 and the closest one in Euclidean norm is chosen as the decoding output.

However, because sampling is random and because the K samples are independent of each other, there are two inherent problems. On one hand, the existence of repetitions means unnecessary complexity is incurred to sample the same lattice point. On the other hand, some lattice points, especially those with small probabilities on early decoding levels are not guaranteed to be sampled, leading to inevitable performance loss.

III. Derandomized Sampling Decoding

A. Algorithm Description

In this section, we propose a derandomized sampling algorithm to solve the afore-mentioned problems originating from randomization. The sampling procedure can be described as three steps (for level index $i = n, n-1, \cdots, 1$):

1) Calculate the probabilities $P = P(\widehat{x}_i^j)$ of integers around $\widetilde{x}_i$ from the discrete Gaussian distribution (7).
2) Calculate the value $E = E(\widehat{x}_i^j) = \lceil KP \rfloor$ where K is the (nominal) sample size.
3) Sample the integers according to the threshold $E \ge 1$:
   - If $E < 1$, then $\widehat{x}_i^j$ is omitted.
   - If $E = 1$, then let $\widehat{x}_i = \widehat{x}_i^j$ and run the standard SIC for $\widehat{x}_{i-1}, \cdots, \widehat{x}_1$ on the remaining levels to generate a candidate lattice point.
   - If $E > 1$, then let $\widehat{x}_i = \widehat{x}_i^j$ and sample $\widetilde{x}_{i-1}$ on the next level with the updated sample size $K' = E$.

We note that the boundary of a finite constellation can be easily controlled in this procedure.

At each decoding level, size K is allocated to candidate integers according to $E = \lceil KP \rfloor$ and all integers with $E \ge 1$ are deterministically sampled. Therefore, lattice points with small probabilities in early stages avoid to be ignored from the beginning, which means the probability that the closest lattice point being sampled is improved. For integers with $E > 1$, after updating the size K', sampling continues from the next level in the same way. Note that when $K = 1$, derandomized sampling decoding performs SIC by always selecting the integer with the largest probability. Hence, for integers with $E = 1$, SIC is applied directly to obtain a candidate lattice point. Finally, among all the candidate lattice points, the closest one is selected as the solution.

As shown in Figure 2, derandomized sampling decoding admits a tree structure. The final candidate list is generated by traversing the tree from level n to level 1 rather than by K independent samples. From this perspective, derandomized sampling functions like a pruning algorithm in sphere decoding [10]–[12] by pruning the branches with $E < 1$. Meanwhile, calculations in branches with $E > 1$ on each decoding level are performed only once without repetitions, thus saving a lot of complexity. Therefore, different from randomized decoding and other decoding schemes establishing a candidate list with constant size around the SIC output [13], [14], the size of the final candidate list $K_{\text{final}} \le K$ is variable, which means the size K set initially is actually a *nominal sample size* of the candidate list.

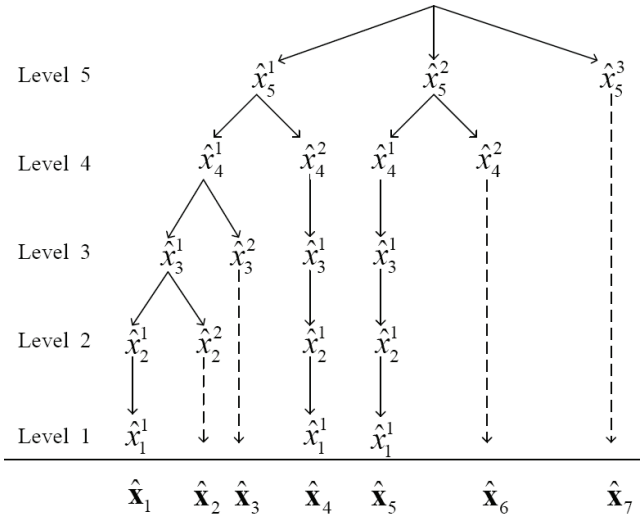

Fig. 2. Illustration of the tree structure in derandomized sampling decoding for case $n = 5$. The dashed line denotes using standard SIC to detect the values in the rest of levels if $E = 1$.

As a nominal sample size, K is essentially a parameter in the threshold $\lceil KP \rfloor \geq 1$ to evaluate the sampling performance. With the increment of K, complexity improves gradually since more lattice points will be sampled. Note that K is not the real size of the final samples, the complexity of derandomized sampling decoding in fact grows slowly with the increment. Because the problems caused by randomization are overcome, derandomized algorithm achieves considerable improvement in both performance and complexity.

B. Performance Analysis

Now we show that the total probability of the samples obtained by the derandomized algorithm is high. To do this, we use a truncation of the discrete Gaussian distribution (7).

As shown in [8], the probability that the integer $\widehat{x}_i$ generated by random rounding $R(\widetilde{x}_i)$ is located within the 2*N* integers around $\widetilde{x}_i$ is bounded by

$$P_{2N} \geq 1 - O(\rho^{-N^2}). \tag{10}$$

Because $\rho > 1$, the term $O(\rho^{-N^2})$ decays exponentially, meaning a finite truncation with moderate N achieves an accurate approximation. Normally, 3-integer approximation is sufficient:

$$P(\widehat{x}_i^1) + P(\widehat{x}_i^2) + P(\widehat{x}_i^3) \approx 1. \tag{11}$$

Since these probabilities follow the discrete Gaussian distribution, they decrease monotonically with the distance from $\widetilde{x}_i$. Let us order them as follows

$$P(\widehat{x}_i^1) \geq P(\widehat{x}_i^2) \geq P(\widehat{x}_i^3). \tag{12}$$

As shown in (4), $\widetilde{x}_i$ is subject to the effect of noise. Intuitively, $\widetilde{x}_i$ tends to be close to an integer for small noise while it tends to be halfway between two integers for large noise. Since $\widetilde{x}_i$ is the peak of the continuous Gaussian distribution associated with the discrete one (7), we define the *worst case* in sampling as the one where $\widetilde{x}_i$ is centered between two integers.

Because random noise makes it hard for an exact analysis, we only consider the worst-case scenario in sampling. Then, under the 3-integer approximation in (11), the following holds in the worst case:

$$P(\widehat{x}_i^1) = P(\widehat{x}_i^2) \gg P(\widehat{x}_i^3) \tag{13}$$

where $P(\widehat{x}_i^3)$ is much smaller due to exponential decay of the probability with the distance.

For an integer $\widehat{x}_i^j$ to be sampled in the derandomized algorithm, $E(\widehat{x}_i^j)$ must satisfy

$$E(\widehat{x}_i^j) = \lceil KP(\widehat{x}_i^j) \rfloor \geq 1 \tag{14}$$

and therefore, $\widehat{x}_i^j$ is sampled if and only if

$$P(\widehat{x}_i^j) \geq \frac{1}{2K}. \tag{15}$$

Now, let us calculate the total probability of lattice points sampled by the derandomized algorithm, in the worst case. Consider level n first. Obviously, the first two integers will be sampled if $K > 2$. If $P(\widehat{x}_n^3) \geq \frac{1}{2K}$, all the 3 integers around $\widetilde{x}_n$ are deterministically sampled. On the other hand, if $P(\widehat{x}_n^3) < \frac{1}{2K}$, integer $\widehat{x}_n^3$ will be discarded while the summation of the probabilities of the other two integers will be larger than $1 - \frac{1}{2K}$ according to (11). Therefore, given the nominal sample size K, the sum probability of samples on level n is bounded by

$$P(\text{level } n) \geq 1 - \frac{1}{2K}. \tag{16}$$

To further derive the lower bound of the total probabilities of samples, we assume the third sample $\widehat{x}_i^3$ at each sampling level is always discarded. Then, still in the worst case, the total probability of the samples on level $n-1$ is given by

$$\begin{aligned} P(\text{level } n-1) &\geq (1-\frac{1}{2K})\left[\frac{1}{2}\left(1-\frac{1}{K}\right)+\frac{1}{2}\left(1-\frac{1}{K}\right)\right] \\ &= (1-\frac{1}{2K})(1-\frac{1}{K}). \end{aligned}$$

Similarly, on level $n-p$, the total probability of the samples in the worst case can be lower bounded by

$$P(\text{level } n-p) > \prod_{i=1}^{p}(1-\frac{2^{i-2}}{K}). \tag{17}$$

Therefore, the total probability of the sampled lattice points in the derandomized algorithm is lower bounded by a function of K. We define a parameter η to evaluate the decoding performance as

$$P(\text{level } 1) > \eta \triangleq \prod_{i=1}^{n}(1-\frac{2^{i-2}}{K}). \tag{18}$$

Obviously, the lower bound η increases with K and a larger η means a higher probability of the closest lattice point being sampled. Thus, derandomized sampling decoding can be used

to approximate the ML decoding as η approaches 1.

The lower bound (18) may be loose because it quantifies the probability in the worst case. For η close to 1, K can be very huge (in fact exponential). A lower bound in the average case is an open question. Because noise is random, the average-case probability may be more useful.

In order to obtain a better estimate, the idea of fixed-complexity sphere decoding (FSD), which also follows a tree structure in decoding, is exploited. Different from standard sphere decoding, it only performs the full search in upper p levels known as the full-expansion stage while standard SIC is applied on the rest of levels. It has been proved in [15] that by applying the channel matrix ordering to make sure signals with the largest postprocessing noise amplification are detected in the full-expansion stage, FSD algorithm yields near-ML performance in high SNR if it satisfies:

$$(p+1)^2 \geq n \tag{19}$$

where p is the number of levels in the full-expansion stage.

We propose to use sampling in the full-expansion stage of FSD. With suitable channel matrix ordering, the modified sampling decoder also consists of two stages. Candidate values on the upper p levels are sampled based on the lower bound η while decodings on the remaining levels are processed by SIC.

According to (17) and (19), if we set η to a value near 1 on the upper p decoding levels, then the decoder will achieve near-ML performance:

$$P(\text{level } n-p) > \prod_{i=1}^{p}(1-\frac{2^{i-2}}{K}) = \eta. \tag{20}$$

Compared with (18), the lower bound (20) is better because p is much smaller than n, which means the value of K achieving the same η is reduced significantly. When $K=1$, derandomized sampling decoding achieves the same performance as SIC. Thus, the decoder enjoys a flexible performance between SIC and ML by adjusting K. To achieve a near-ML performance, η is recommended to be no less than 0.9.

It is worth pointing out that the complexity analysis of derandomized lattice decoding is hard to perform as the final sample size is uncertainty. Nonetheless, by solving the problem of sampling repetitions, the proposed algorithm has much lower complexity than randomized lattice decoding, which is demonstrated by simulations.

IV. Simulation Results

In this section, performance and complexity of the derandomized sampling decoding in MIMO systems are studied. Here, the i-th entry of the transmitted signal $\mathbf{x}$, denoted as x_i, is the modulation symbol taken independently from a Q^2-QAM constellation $\mathcal{X}$ with $E\ [|x_i|^2]=\frac{1}{n}$. Let E_b represents the average power per bit at the receiver, then $E_b/N_0 = n/(\log_2(M)\sigma^2)$ holds where M is the modulation level and σ^2 is the noise power.

Figure 3 shows the bit error rate (BER) of the derandomized sampling decoding compared with other decoding schemes in a 20×20 uncoded system with 64-QAM. Clearly, under the help of LLL reduction ($\delta = 0.75$) all the decoding schemes attain the full receive diversity and sampling decoding schemes have considerable gains over lattice-reduction-aid SIC. Compared to fixed candidates algorithm (FCA) in [13] and iterative list decoding in [14] with 30 samples, sampling decodings offer not only the improved BER performance but also the promise of smaller list size. As expected, derandomized sampling decoding achieves a better BER performance than randomized sampling decoding with the same K. Specifically, the gain in MMSE schemes with $K=15$ is approximately 1 dB for a BER of 10^{-4}.

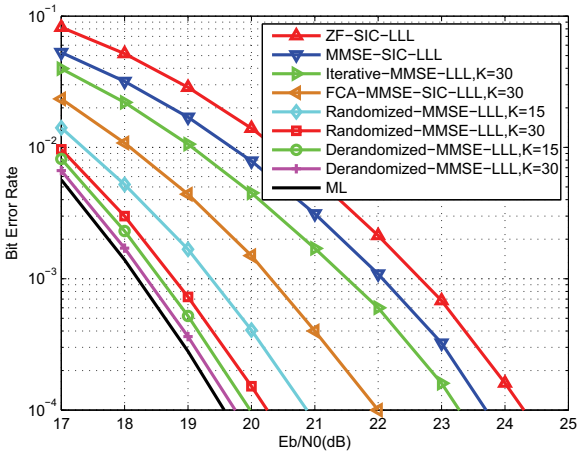

Fig. 3. Performance comparison of derandomized sampling decoding with randomized sampling decoding for a 20×20 MIMO system using 64-QAM.

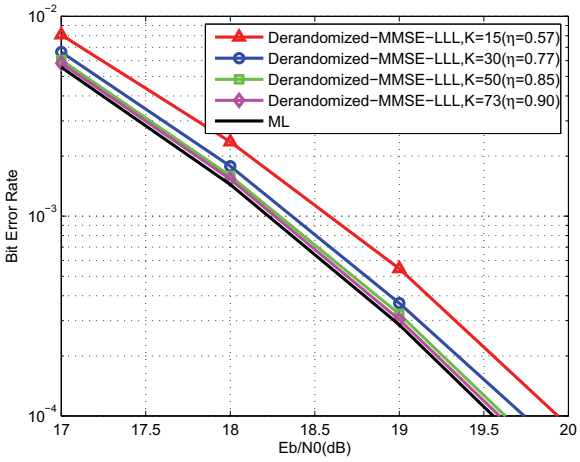

Fig. 4. Performance comparison of derandomized sampling decoding with different sample sizes K for a 20×20 MIMO system using 64-QAM.

Figure 4 compares the BER performance of the LLL reduced derandomized sampling decoding with different values of K in a 20×20 uncoded system with 64-QAM. According

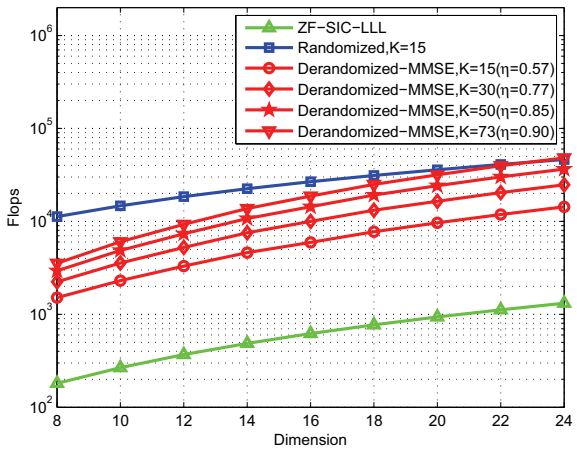

Fig. 5. Complexity comparison in flops for a MIMO system using 64-QAM at SNR per bit = 17dB.

to (20), a larger K leads to a larger lower bound η. Therefore, with the increment of K, the BER performance improves gradually. Observe that with $\eta = 0.9$, the performance of derandomized sampling decoding suffers negligible loss compared with ML. Therefore, with a moderate K, derandomized sampling decoding achieves a near-ML performance.

Figure 5 shows the complexity comparison of the derandomized sampling decoding with other decodings in different dimensions. It is clearly seen that in a 64-QAM MIMO system for the fixed SNR ($E_b/N_0 = 17$ dB), the derandomized sampling decoding needs much lower average flops than randomized sampling decoding with the same size K. Even for a large K for $\eta = 0.9$, the complexity is still lower than that of randomized sampling decoding with $K = 15$. Therefore, better BER performance and less complexity make derandomized sampling decoding a very promising algorithm for lattices.

V. Conclusions

In this paper, we proposed a derandomized algorithm to address the issues in sampling decoding caused by randomization. By setting a probability threshold to sample the candidates, the whole sampling procedure becomes deterministic, which brings further performance improvement and complexity reduction. Then we derived a lower bound on the total probability of samples collected by the derandomized decoding algorithm and found an effective way to bridge the gap between SIC and ML by varying K. Simulations showed that the proposed algorithm outperforms randomized sampling decoding with much lower complexity. Other questions such as the lower bound in the average case and selection of the parameter ρ will be studied in future work.

References

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

[2] L. Luzzi, G. Othman, and J. Belfiore, "Augmented lattice reduction for low-complexity MIMO decoding," *IEEE Trans. Wireless Commun.*, vol. 9, pp. 2853–2859, Sep. 2010.

[3] D. Wubben, R. Bohnke, V. Kuhn, and K. D. Kammeyer, "Near-maximum-likelihood detection of MIMO systems using MMSE-based lattice reduction," in *Proc. IEEE Int. Conf. Commun.(ICC'04)*, Paris, France, Jun. 2004, pp. 798–802.

[4] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[5] 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, Jul. 2009.

[6] 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. Inform. Theory*, vol. 56, no. 10, pp. 4765–4780, Oct. 2010.

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

[8] S. Liu, C. Ling, and D. Stehle, "Decoding by sampling: a randomized lattice algorithm for bounded distance decoding," *IEEE Trans. Inform. Theory*, vol. 57, pp. 5933–5945, Sep. 2011.

[9] P. Klein, "Finding the closest lattice vector when it is unusually close," *SIAM Symposium on Discrete Algorithms*, pp. 937–941, ACM, 2000.

[10] R. Gowaikar and B. Hassibi, "Statistical pruning for near-maximum likelihood decoding," *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2661–2675, Jun. 2007.

[11] W. Zhao and G. Giannakis, "Sphere decoding algorithms with improved radius search," *IEEE Trans. Commun.*, vol. 53, no. 7, pp. 1104–1109, Jul. 2005.

[12] B. Shim and I. Kang, "Sphere decoding with a probabilistic tree pruning," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4867–4878, Oct. 2008.

[13] W. Zhang and X. Ma, "Low-complexity soft-output decoding with lattice-reduction-aided detectors," *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2621–2629, Sep. 2010.

[14] T. Shimokawa and T. Fujino, "Iterative lattice reduction aided MMSE list detection in MIMO system," in *Proc. IEEE International Conference on Advanced Technologies for Communications*, Oct. 2008, pp. 50–54.

[15] J. Jalden, L. Barbero, B. Ottersten, and J. Thompson, "The error probability of the fixed-complexity sphere decoder," *IEEE Trans. Signal Process.*, vol. 57, pp. 2711–2720, Jul. 2009.