Practical HKZ and Minkowski Lattice Reduction Algorithms

Wen Zhang, Sanzheng Qiao, and Yimin Wei

August 17, 2011

Abstract

Recently, lattice reduction has been widely used for signal detection in multiinput multioutput (MIMO) communications. In this paper, we present three novel lattice reduction algorithms. First, using a unimodular transformation, a significant improvement on an existing Hermite-Korkine-Zolotareff-reduction algorithm is proposed. Then, we present two practical algorithms for constructing Minkowski-reduced (M-reduced) bases. To assess the output quality, we compare the orthogonality defect of the reduced bases produced by LLL algorithm and our new algorithms, and find that in practice M-reduced basis vectors are the closest to being orthogonal. An error-rate analysis of suboptimal decoding algorithms aided by different reduction notions is also presented. To this aim, the proximity factor is employed as a measurement. We improve some existing results and derive upper bounds for the proximity factors of Minkowski-reduction-aided decoding (MRAD) to show that MRAD can achieve the same diversity order with infinite lattice decoding (ILD).

Keywords Lattice reduction, LLL, HKZ, Minkowski, MIMO detection, proximity factors.

1 Introduction

In this paper, we shall concern with the problem of *lattice basis reduction* and its application in MIMO detection. Suppose that $\mathbf{B}$ is an m-by-n, $m \geq n$, real matrix of full column rank, then a *lattice* generated by $\mathbf{B}$ is defined by the set:

$$L(\mathbf{B}) = \{\mathbf{Bz} \quad | \quad \mathbf{z} \in \mathbb{Z}^n\},$$

where $\mathbb{Z}^n$ denotes the set of integer n-vectors. The columns of $\mathbf{B}$ form a *basis* for the lattice $L(\mathbf{B})$, and the value of n is called the *dimension* of $L(\mathbf{B})$. When $n \geq 2$, the lattice $L(\mathbf{B})$ can have infinitely many different bases other than $\mathbf{B}$.

Since a lattice can have more than one basis, it is desirable to find one that is reasonably short and nearly orthogonal. A lattice basis consisting of relatively short vectors is called *reduced*. An ideally reduced basis consists of shortest possible vectors. The problem of finding good reduced bases is known as *lattice reduction*, which plays an important role in many fields of mathematics and computer science [1–4], particularly in communications [5–7] and cryptology [8, 9].

In wireless communications, many problems arised from the linear MIMO model, such as code-division multiple access (CDMA) [10] and linearly precoded orthogonal frequency-division multiplexing (OFDM) [11], can be solved optimally by maximum-likelihood (ML) decoding of MIMO systems. For lattice-type modulation, ML decoding can be modeled as the problem of

finding a lattice point closest to a given received signal vector, or equivalently, the *closest vector problem* (CVP) [5, 12–14]. To solve CVP exactly, several solvers, such as the *sphere decoding* algorithms [5, 12, 15, 16], are developed, and lattice reduction has become a powerful tool for reducing the decoding complexity. However, the complexity of the sphere decoding algorithms increase exponentially with the number of transmit antennas [5, 13, 14]. Thus, for large problem sizes or real-time fast fading situations where the received signal vectors change rapidly, ML detection can not meet the time requirement. This motivates the presentations of several low-complexity suboptimal decoding algorithms such as zero-forceing (ZF) decoding and successive interference cancellation (SIC) decoding [17–19]. To improve the performance loss of these approximate detectors, lattice reduction is believed to be an efficient preprocessor [6, 19–21].

There are various definitions of reduced bases. They differ in the degree of reduction. In 1850, Hermite introduced the first notion of reduction for lattices of arbitrary dimension. Algorithms for achieving such reduction can be found in [15, 22]. In 1873, Korkine and Zolotareff [23] strengthened the definition of Hermite-reduction, and their proposed notion is referred to as *HKZ-reduction* [2], named after Hermite, Korkine and Zolotareff. In 1983, using induction, Kannan [24] presented the first algorithm for constructing HKZ-reduced bases. Helfrich [25], Kannan [26], and Banihashemi and Khandani [27] further refined Kannan's algorithm and improved the complexity analysis. However, due to the super exponential complexity of Kannan's algorithm [24], most algorithms based on Kannan's strategy are intended as theoretical tools, and related papers [25–27] focus on asymptotic complexity.

In 1891, Minkowski [28] defined another reduction notion, which is now known as *Minkowski-reduction* (M-reduction). The concept of M-reduction is of fundamental importance in many fields of mathematics. For example, M-reduced bases are used in assessing the quality of random number generators [3] and used in the reduction of quadratic forms in number theory [1].

The construction of M-reduced bases is a classical problem which attracts much attentions. In 1773, Lagrange [29] presented the first algorithm for producing an M-reduced basis for lattices of dimension two. Recently, this algorithm was extended to dimensions three and four by Semaev [30] and Nguyen and Stehlé [31], respectively. More generally, Helfrich [25] and Afflerbach and Grothe [32] presented algorithms for constructing M-reduced bases for lattices of arbitrary dimension. However, these algorithms are believed to be of theoretical values for high dimensional lattices since their complexity are also super exponential with respect to the lattice dimension.

The construction of HKZ-reduced bases or M-reduced bases consists of a sequence of *shortest vector problems* (SVPs). The SVP, which is actually a special CVP, is to find a shortest nonzero lattice point with respect to the L_2-norm in a given lattice. This problem has been proven to be NP-hard [33]. Even finding an approximate solution with which the ratio between the computed distance and the shortest distance is upper-bounded by a constant, is also NP-hard [34]. That is why the construction of an HKZ-reduced or an M-reduced basis requires intensive computation. In 1982, Lenstra, Lenstra and Lovász relaxed the definition of Hermite-reduction [22] to obtain a new reduction notion, known as *LLL-reduction*, named after the three authors [35]. The associated LLL algorithm is the first polynomial-time lattice reduction algorithm and has been widely used in public-key cryptanalysis [2, 36] and MIMO detection/precoding [7, 19]. Further improvements of LLL algorithm have been developed. While some improve the output quality [37–39], others improve the efficiency [6, 40–44].

Our results. This paper presents three practical algorithms: one for constructing HKZ-

reduced bases and two for constructing M-reduced bases. The first algorithm uses the same SVP solver during the recursive process as the algorithm in [5]. However, it uses a different method for the expansion of a shortest vector into a new lattice basis. In [5], the basis expansion strategy introduced by Kannan [24] is used, while in our new algorithm, the unimodular transformation technique presented in [45] is used. Note that Kannan's basis expansion method only works for rational lattices, while our unimodular transformation technique works for any real lattice and is much more efficient than Kannan's method.

The other two algorithms are focused on the construction of M-reduced bases for lattices of arbitrary dimension. In general, both algorithms are based on Schnorr-Euchner search strategy [15] and the unimodular transformation technique [45]. Specifically, the first algorithm uses a simple variation of Schnorr-Euchner enumeration to compute each M-reduced basis vector, while the second algorithm dynamically monitors the basis expansion condition during the search process. Thus, the second algorithm is more efficient than the first one. However, the first algorithm can be preconditioned by using LLL algorithm, while the second one can not. To accelerate the second algorithm, we propose a partial lattice reduction method as a preprocessor. Numerical results show that the second algorithm is much faster than the first one, and both of them can significantly outperform the existing algorithms [25, 32].

Note that in most communication applications, the lattice needed for decoding remains unchanged, while the observed received vectors change frequently. That is, the preprocessing of the lattice generator matrix needs to be performed only once, while the processed basis is typically used many times. So it is worthy to invoke a good preprocessing procedure, even it requires a relatively high complexity. Since the vectors of HKZ and M-reduced bases are shorter and more orthogonal than those of LLL-reduced bases, the bit-error-rate (BER) performance of approximate MIMO detectors is expected to be further improved by applying our new algorithms. Theoretically, we discuss the BER related proximity factor as defined in [20], and prove that M-reduction-aided decoding, such as ZF decoding and SIC decoding, can achieve the same receive diversity with ILD. Numerical results show that M-reduction-aided ZF decoding performs slightly better than that based on LLL or HKZ-reduced bases, while HKZ-reduction-aided SIC decoding performs better than that aided by LLL or M-reduced bases.

All algorithms presented in this paper are described in matrix form. The rest of the paper is organized as follows. In Section 2, we introduce the MIMO system model and review several concepts in lattice theory. In Sections 3, we briefly review and compare different SVP solvers. The new algorithm for constructing HKZ-reduced bases is given in Section 4. Section 5 presents the first algorithm for constructing M-reduced bases. The partial lattice reduction preprocessor as well as the second algorithm for constructing M-reduced bases are presented in Section 6. In Section 7, we discuss the performance of ZF decoding and SIC decoding aided by different reduction notions. In Section 8, we present our experimental results. Finally, the paper is concluded in Section 9.

2 Preliminaries

In this section, we shall briefly introduce the model of MIMO systems as well as several detection methods. Some basic concepts in the field of lattice theory are also presented.

2.1 System Model

Consider an $n_R \times n_T$ MIMO system consisting of n_T transmit antennas and n_R receive antennas. The relationship between the $n_T \times 1$ transmitted signal vector $\mathbf{x}$ and the $n_R \times 1$ received signal vector $\mathbf{y}$ is given by

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

where $\mathbf{H}$ is an $n_R \times n_T$ matrix representing the channel matrix, and $\mathbf{n}$ is an $n_R \times 1$ vector representing the additive noise vector. In a full-rank flat-fading MIMO system, $\mathbf{H}$ is a complex matrix with all of its elements being independent Gaussian random variables $CN(0,1)$, and the noise $\mathbf{n}$ is a complex vector with all of its elements being independent Gaussian random variables $CN(0, 2\sigma^2)$. Treating the real and imaginary parts of (1) separately, an equivalent real-valued system of doubled dimension can be obtained, with the transformed channel matrix

$$\mathbf{B} = \begin{bmatrix} \Re(\mathbf{H}) & -\Im(\mathbf{H}) \\ \Im(\mathbf{H}) & \Re(\mathbf{H}) \end{bmatrix} \tag{2}$$

where $\Re(\mathbf{H})$ and $\Im(\mathbf{H})$ denote the real and imaginary parts of $\mathbf{H}$, respectively. We shall adopt such real-valued model throughout this paper.

2.2 Detection Methods and Lattice Viewpoint

Given a MIMO system modeled as (1), and let $n = 2n_T$. Then the optimum ML decoding selects $\mathbf{x}_{ML}$ that is a solution for the following minimization problem as the transmit signal:

$$\mathbf{x}_{ML} = \arg\min_{\mathbf{x}\in\mathcal{A}} \|\mathbf{y} - \mathbf{B}\mathbf{x}\|_2, \tag{3}$$

where $\mathcal{A}$ denotes the finite set of real-valued modulation alphabet being used. Assume that the constellation $\mathcal{A}$ is of lattice type, such as PAM or QAM, then upon scaling and shifting the problem (3) can be transformed into an integer least squares problem. For solving such problem exactly, several algorithms such as Kannan's method [24] as well as the sphere decoding algorithms [5, 15, 16] are proposed. However, the complexity of these algorithms increase exponentially with the number of transmit antennas [5, 13, 14]. So ML decoding is not feasible for large number of transmit antennas or fast fading situation where the received signal changes rapidly.

To reduce the detection cost, many approximate algorithms with low-complexity have been proposed, such as ZF decoding and SIC decoding [17–19]. For ZF decoding, the interference is completely suppressed by multiplying the received vector $\mathbf{y}$ with the Moore-Penrose inverse $\mathbf{B}^\dagger = (\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}\mathbf{B}^{\mathrm{T}}$ of the channel matrix where $\mathbf{B}^{\mathrm{T}}$ represents the transpose of $\mathbf{B}$. The signal vector $\mathbf{x}_{ZF}$ is then decided by mapping each element of the output vector $\mathbf{B}^\dagger\mathbf{y}$ onto an element of the symbol alphabet by a minimum distance quantization. As shown in [42], SIC decoding can be described in terms of QR decomposition of the channel matrix. Specifically, we first apply QR decomposition $\mathbf{B} = \mathbf{Q}\mathbf{R}$ such that $\mathbf{Q}$ consists of orthonormal columns and $\mathbf{R}$ is upper triangular. Then by multiplying $\mathbf{Q}^{\mathrm{T}}$ to (1), we can obtain

$$\mathbf{y}' = \mathbf{R}\mathbf{x} + \mathbf{n}', \tag{4}$$

where $\mathbf{y}' = \mathbf{Q}^{\mathrm{T}}\mathbf{y}$ and $\mathbf{n}' = \mathbf{Q}^{\mathrm{T}}\mathbf{n}$. Due to the upper triangular structure of $\mathbf{R}$, the n-th element of $\mathbf{x}$ is free of interference, and by assuming that all previous decisions are correct, the interference

can be perfectly cancelled in each step thus $\mathbf{x}_{n-1}, \cdots, \mathbf{x}_1$ can be detected successively. Based on SIC decoding, the so-called V-BLAST decoding [18] can further improve the detection performance by adopting a detection order in accordance with the descending order of signal-to-noise ratios (SNR) of different elements in a received vector.

The performance of approximate detectors is highly related to the structure of $\mathbf{B}$. It is well known that the closer to being orthogonal the column vectors of $\mathbf{B}$ are, the lower BER the approximate detectors have [19, 21, 46]. Especially, ZF decoding and SIC decoding are identical to ML decoding if $\mathbf{B}$ is orthogonal.

An integer matrix $\mathbf{Z} \in \mathbb{Z}^{n\times n}$ is called *unimodular* if $|\det(\mathbf{Z})| = 1$, where $\det(\mathbf{Z})$ denotes the determinant of $\mathbf{Z}$. In general, the columns vectors of any matrix $\mathbf{B}'$ can form a basis for $L(\mathbf{B})$ if and only if $\mathbf{B}'$ can be factorized as $\mathbf{B}' = \mathbf{BZ}$, where $\mathbf{Z}$ is a unimodular matrix. A lattice reduction algorithm is an algorithm that, given $\mathbf{B}$, finds a proper unimodular matrix $\mathbf{Z}$ such that $\mathbf{BZ}$ is reduced. Suppose that $\mathbf{Z}$ is a unimodular matrix produced by a lattice reduction algorithm, then the MIMO system can be transformed into

$$\mathbf{y} = \mathbf{Bx} + \mathbf{n} = \mathbf{BZZ}^{-1}\mathbf{x} + \mathbf{n} = \mathbf{B}'\mathbf{x}' + \mathbf{n}, \tag{5}$$

where $\mathbf{B}' = \mathbf{BZ}$ and $\mathbf{x}' = \mathbf{Z}^{-1}\mathbf{x}$. Applying the MIMO detection algorithms aforementioned on (5), $\mathbf{x}'$ can be obtained and thus the transmit vector $\mathbf{x}$ is given by $\mathbf{x} = \mathbf{Zx}'$.

2.3 Some Basic Definitions

In this subsection, we shall introduce several classical concepts in lattice theory that will be used throughout this paper.

2.3.1 Lattice volume and orthogonality defect

Let L be the lattice generated by a matrix $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n] \in \mathbb{R}^{m\times n}$, $m \geq n$. Then the *volume* of L is defined as $\mathrm{vol}(L) = \sqrt{\det(\mathbf{B}^{\mathrm{T}}\mathbf{B})}$. From the definition of unimodular, we have $\det(\mathbf{B}^{\mathrm{T}}\mathbf{B}) = \det((\mathbf{BZ})^{\mathrm{T}}\mathbf{BZ})$ for any unimodular $\mathbf{Z} \in \mathbb{Z}^{n\times n}$. Hence the volume of a lattice is independent of the choice of basis. The *orthogonality defect* of the basis $\{\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_n\}$ for L is defined as $\delta(\mathbf{B}) = (\prod_{i=1}^{n} \|\mathbf{b}_i\|_2)/\mathrm{vol}(L)$. The concept of orthogonality defect is used to measure the degree of orthogonality for a given matrix. From Hardama's Inequality, $\delta(\mathbf{B})$ is always larger than or equal to 1, with equality if and only if $\mathbf{B}$ is orthogonal.

2.3.2 Gram-Schmidt orthogonalization and QR decomposition

Let $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_n]$ be of full column rank. Then *Gram-Schmidt orthogonalization* (GSO) $\mathbf{q}_1^*, \cdots, \mathbf{q}_n^*$ are defined as follows: for any $1 \leq j \leq n$, $\mathbf{q}_j^*$ is the component of $\mathbf{b}_j$ that is orthogonal to the subspace spanned by the vectors $\mathbf{b}_1, \cdots, \mathbf{b}_{j-1}$. Initializing with $\mathbf{q}_1^* = \mathbf{b}_1$, the vectors $\mathbf{q}_2^*, \cdots, \mathbf{q}_n^*$ can be calculated successively by

$$\mathbf{q}_j^* = \mathbf{b}_j - \sum_{i<j} u_{i,j}\mathbf{q}_i^*, \quad 1 < j \leq n, \tag{6}$$

where $u_{i,j} = \frac{\langle \mathbf{b}_j, \mathbf{q}_i^* \rangle}{\|\mathbf{q}_i^*\|_2^2}$ ($\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors). Another orthogonalization approach is QR decomposition, obtained by applying a sequence of orthogonal transformations

such as Householder transformations [47]:

$$\mathbf{B} = \mathbf{QR}, \tag{7}$$

where $\mathbf{Q}$ consists of orthonormal columns, and $\mathbf{R} = [r_{i,j}]$ is an upper triangular matrix with positive diagonal. Instead of GSO, many recent lattice reduction algorithms [7, 38, 46, 48–50] adopt the QR decomposition approach, since the QR decomposition can be performed efficiently and numerically more stable than GSO.

2.3.3 Minkowski's successive minima and Hermite's constant

Let L be an n-dim lattice in $\mathbb{R}^m$. For $1 \leq i \leq n$, the i-th *Minkowski's successive minima* $\lambda_i(L)$ is the radius of the smallest closed ball centered at the origin containing at least i linearly independent lattice vectors. In particular, $\lambda_1(L)$ is the Euclidean length of a shortest nonzero lattice vector of L. There always exist independent lattice vectors $\mathbf{v}_i$'s such that $\|\mathbf{v}_i\|_2 = \lambda_i(L)$ for all i. Note that for $n > 4$, such vectors do not necessarily form a basis for L. It is a classical fact that $\lambda_1(L)/\mathrm{vol}(L)^{1/n}$ can be upper bounded over all n-dim lattices L, and Hermite's constant γ_n is defined as the supremum of $\lambda_1(L)^2/\mathrm{vol}(L)^{2/n}$ over all n-dim lattices. Finding the exact value of γ_n is a very difficult problem, which plays a central role in the theory of geometry of numbers. The exact value of γ_n is only known for $1 \leq n \leq 8$ and $n = 24$ [2, Page 33]. An upper bound of Hermite's constant is given in [2, Page 35]:

$$\gamma_n \leq 1 + \frac{n}{4}, \quad \text{for all } n \geq 1. \tag{8}$$

2.3.4 Size-reduction and HKZ-reduction

A lattice generator matrix $\mathbf{B} \in \mathbb{R}^{m \times n}$ is called *size-reduced* if the upper triangular factor $\mathbf{R} = [r_{i,j}]$ of its QR decomposition satisfies:

$$|r_{i,j}| \leq \frac{1}{2}|r_{i,i}|, \quad \text{for } 1 \leq i < j \leq n. \tag{9}$$

By calling the procedure as shown in Fig. 1, the condition (9) can be enforced. A generator matrix $\mathbf{B}$ is called HKZ-reduced if it is size-reduced and its R-factor $\mathbf{R}$ satisfies: for all $1 \leq i \leq n$, $r_{i,i} = \lambda_1(L(\mathbf{R}(i:n, i:n)))$, where $L(\mathbf{R}(i:n, i:n))$ is the lattice generated by $\mathbf{R}(i:n, i:n)$. It is proved in [51] that the length of each HKZ-reduced basis vector can approximate Minkowski's successive minima within a polynomial factor:

$$\frac{4}{i+1} \leq \frac{\|\mathbf{b}_i\|_2^2}{\lambda_i^2(L)} \leq \frac{i+3}{4}, \quad 1 \leq i \leq n; \tag{10}$$

$$\prod_{i=1}^{n} \|\mathbf{b}_i\|_2 \leq \left(\gamma_n^n \prod_{i=1}^{n} \frac{i+3}{4}\right)^{\frac{1}{2}} \cdot \mathrm{vol}(L). \tag{11}$$

Procedure SIZE-REDUCE($\mathbf{R}$, $\mathbf{Z}$, i, j)

Input: $\mathbf{R} \in \mathbb{R}^{n\times n}$, $\mathbf{Z} \in \mathbb{Z}^{n\times n}$, and indices i, j
Output: $\mathbf{R}$ with $|r_{i,j}| \le |r_{i,i}|/2$, and updated $\mathbf{Z}$.
1: **if** $|r_{i,j}| > |r_{i,i}|/2$ **then**
2: $\quad t \leftarrow \lfloor r_{i,j}/r_{i,i} \rceil$
3: $\quad \mathbf{R}(1:i,j) \leftarrow \mathbf{R}(1:i,j) - t \cdot \mathbf{R}(1:i,i)$
4: $\quad \mathbf{Z}(:,j) \leftarrow \mathbf{Z}(:,j) - t \cdot \mathbf{Z}(:,i)$
5: **end if**

Figure 1: The size-reduction algorithm

2.3.5 Minkowski-reduction

A lattice generator matrix $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_n]$ is called M-reduced if for all $1 \le i \le n$, the vector $\mathbf{b}_i$ has the minimum norm among all lattice vectors $\mathbf{b}_i$ such that $\{\mathbf{b}_1, \cdots, \mathbf{b}_i\}$ can be extended to a basis for $L(\mathbf{B})$ [28]. Intuitively, M-reduction requires each basis vector as short as possible. From [52], the length of each M-reduced basis vector can be bounded by

$$\lambda_i^2(L) \le \|\mathbf{b}_i\|_2^2 \le \max\{1, (5/4)^{(n-4)}\}\lambda_i^2(L), \quad 1 \le i \le n; \tag{12}$$

$$\prod_{i=1}^{n} \|\mathbf{b}_i\|_2 \le \gamma_n^{\frac{n}{2}} \cdot \mathrm{vol}(L), \quad \text{for} \quad n \le 4; \tag{13}$$

$$\prod_{i=1}^{n} \|\mathbf{b}_i\|_2 \le \gamma_n^{\frac{n}{2}} \cdot (5/4)^{\frac{(n-3)(n-4)}{4}} \cdot \mathrm{vol}(L), \quad \text{for} \quad n > 4. \tag{14}$$

We can obtain from (12) that for lattices of dimension $n \le 4$, the norms of M-reduced basis vectors can simultaneously achieve Minkowski's successive minima. In high dimensions, however, there need not exist an M-reduced basis whose vector norms simultaneously reach Minkowski's successive minima.

2.3.6 LLL-reduction

A lattice generator matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$ is called LLL-reduced if it is size-reduced and its R-factor $\mathbf{R} = [r_{i,j}]$ satisfies:

$$r_{i,i}^2 + r_{i-1,i}^2 \ge \omega\, r_{i-1,i-1}^2, \quad 1 < i \le n, \tag{15}$$

where $\omega \in (1/4, 1)$ is a parameter which influences the quality of the reduced basis. Obviously, an HKZ-reduced basis is LLL-reduced for any $1/4 < \omega < 1$. To justify that an LLL-reduced basis consists of vectors reasonably short, it is shown in [35] that

$$\beta^{1-i}\lambda_i^2 \le \|\mathbf{b}_i\|_2^2 \le \beta^{n-1}\lambda_i^2, \tag{16}$$

$$\prod_{i=1}^{n} \|\mathbf{b}_i\|_2 \le \beta^{\frac{n(n-1)}{4}} \cdot \mathrm{vol}(L). \tag{17}$$

where $\beta = (\omega - 1/4)^{-1}$. Like M-reduction, the upper bound in the right hand side of (16) grows exponentially with the dimension of lattice. However, M-reduction is stronger than LLL-reduction, since the exponential factor in (12) is smaller than that of LLL-reduction for any $1/4 < \omega < 1$.

The LLL algorithm [35] is the first lattice reduction method which can practically produce a reduced basis of high quality in polynomial time. It is shown in [44, 53] that LLL algorithm has an average complexity of $O(mn^3 \log n)$ flops over all Gaussian random matrices whose entries are i.i.d. $\mathcal{N}(0, 1)$ distributed. Therefore LLL algorithm has become one of the most practical tools for MIMO detection [6, 12, 19, 54].

3 Algorithms for Solving SVP

As pointed out previously, the calculation of HKZ or M-reduced bases involves solving a sequence of SVP, which is known as a fundamental problem in lattice theory. Let $\mathbf{B} \in \mathbb{R}^{m \times n}$ be a lattice generator matrix of full column rank, then SVP can be modeled as:

$$\min_{\mathbf{z} \neq \mathbf{0}} \{\|\mathbf{Bz}\|_2 : \ \mathbf{z} = [z_i] \in \mathbb{Z}^n\} \tag{18}$$

From (18), SVP is actually a special CVP with $\mathbf{0}$ being the observed vector to decode.

There are many algorithms for solving SVP exactly, and the choice of methods depends on the structure of the lattice generator matrix. For many classical lattices, efficient search algorithms exploiting the special structure of the lattice generator matrix are known [55, 56]. For general SVP, that is, the lattice generator matrix has no exploitable structure, related algorithms can be classified in three categories: algorithms based on Kannan's strategy [24–27], the sphere decoding algorithms [5, 15, 16, 57, 58] and the sieve algorithms [59–61], of which the first two are deterministic enumeration algorithms and are unified in the same framework in [5], while the last one are randomized algorithms.

In general, the common feature of most deterministic algorithms is to first identify a region in which a shortest lattice point must lie, and then exhaustively search the lattice points lying in this region for the shortest nonzero lattice vector, while possibly reducing the size of the region dynamically.

3.1 Algorithms Based on Kannan's Strategy

Kannan's algorithm was first presented in [24] and further improved in [25–27]. Let $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$ be a basis for an n-dim lattice L and let $\pi_2(\cdot)$ be an orthogonal projection operator which projects "$\cdot$" onto $\mathbf{b}_1^{\perp}$, where $\mathbf{b}_1^{\perp}$ denotes the orthogonal complement of the subspace spanned by $\mathbf{b}_1$. The basic idea of Kannan enumeration is to first find an HKZ-reduced basis for the lattice $\pi_2(L)$ by calling itself recursively, and then lift it to a size-reduced basis $(\mathbf{b}_1, \mathbf{b}_2', \cdots, \mathbf{b}_n')$ for L such that $(\pi_2(\mathbf{b}_2'), \cdots, \pi_2(\mathbf{b}_n'))$ is an HKZ-reduced basis of $\pi_2(L)$ and $\|\mathbf{b}_1\|_2^2 \leq \frac{4}{3}\|\mathbf{b}_2'\|_2^2$. Then the shortest lattice point must lie in a parallelepiped of cardinality no more than $n^{0.5n+o(1)}$ and can thus be found by enumerating this finite set. It is proved in [25, 26] that algorithms based on Kannan's strategy require a complexity of $n^{0.5n+o(n)}$ polynomial-time operations. Variants of Kannan' strategy [24], [25], [26], [27] differ mainly in how the size of the search region for each iteration level are chosen. Note that Kannan enumeration not only finds a shortest lattice point, but also constructs an HKZ-reduced basis simultaneously.

3.2 The Sphere Decoding Algorithms

Let ρ be the radius of the initial search sphere in which at least one shortest lattice point must lie. Then the basic idea of the sphere decoding algorithms is to enumerate lattice points lying in the hypersphere defined by:

$$\|\mathbf{Bz}\|_2^2 \leq \rho^2. \tag{19}$$

Such hypersphere search strategy was firstly presented in [16] and further improved in [5, 15, 57, 58]. To learn the principle of the hypersphere enumeration more directly, we shall present a recursive version of the sphere decoding algorithms in this section. Let $\mathbf{B} = \mathbf{QR}$ be the QR decomposition of $\mathbf{B}$, then (19) can be transformed into

$$\|\mathbf{Rz}\|_2^2 \leq \rho^2 \tag{20}$$

Due to the upper triangular structure of $\mathbf{R}$, the last entry of $\mathbf{Rz}$ is a function of z_n only. Thus, the upper and lower bounds of z_n can be obtained from (20),

$$\left\lceil -\frac{\rho}{r_{n,n}} \right\rceil \leq z_n \leq \left\lfloor \frac{\rho}{r_{n,n}} \right\rfloor. \tag{21}$$

Partition $\mathbf{R}$ into

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_{n-1} & \mathbf{h} \\ \mathbf{0}^{\mathrm{T}} & r_{n,n} \end{bmatrix}, \tag{22}$$

where $\mathbf{R}_{n-1} \in \mathbb{R}^{(n-1)\times(n-1)}$, $\mathbf{h} \in \mathbb{R}^{n-1}$. Then for each integer value of z_n satisfying (21), the n-dim SVP (18) is reduced to an $(n-1)$-dim CVP

$$\min\{\|\mathbf{R}_{n-1}\mathbf{z}' + z_n\mathbf{h}\|_2 : \ \mathbf{z}' \in \mathbb{Z}^{n-1}\}, \tag{23}$$

with a solution lying in the $(n-1)$-dim hypersphere

$$\|\mathbf{R}_{n-1}\mathbf{z}' + z_n\mathbf{h}\|_2^2 \leq \rho^2 - z_n^2 r_{n,n}^2. \tag{24}$$

Therefore, an n-dim SVP can be reduced to a finite number (at most $\lfloor 2\rho/r_{n,n} \rfloor + 1$) of $(n-1)$-dim CVPs, leading to a recursive algorithm. In summary, we unite Phost's strategy [16, 57, 58] and Schnorr-Euchner strategy [5, 15] in the same framework, and present a recursive implementation in Fig. 2.

Obviously, a shortest nonzero lattice vector can be found by calling Algorithm Sph-Dec($\mathbf{R}$, $\mathbf{0}$, ϕ, r, 0). From line 1 and the for-loop from line 4 to line 18, one n-dim problem can be solved recursively by reducing it to at most $\lfloor 2\sqrt{r}/r_{n,n} \rfloor + 1$ $(n-1)$-dim subproblems as described in (23). From lines 11–13, the size l of the search region is reduced dynamically. That is, when any lattice point $\mathbf{Rz}'$ inside the search region is found, the squared radius l can be reduced to $\|\mathbf{Rz}'\|_2^2$, since $\|\mathbf{Rz}'\|_2^2 < l$. Therefore, not all of the $\lfloor 2\sqrt{r}/r_{n,n} \rfloor + 1$ $(n-1)$-dim subproblems are necessarily to be solved in practice.

Note that the condition in line 14 is to make sure that the lattice points being searched are nonzero. The above algorithm can be applied to solve general CVP, by deleting "**if** *newdist* $\neq 0$" from line 14. The efficiency of the sphere decoding algorithms lies in the following aspects:

Algorithm SPH-DEC($\mathbf{R}$, $\mathbf{x}$, $\mathbf{z}_{in}$, r, $dist$)

Input: $\mathbf{R} \in \mathbb{R}^{n \times n}$, a vector $\mathbf{x} = [x_i] \in \mathbb{R}^n$ to decode, an integer partial solution $\mathbf{z}_{in}$, the current distance record r and the distance to examined layer $dist$.
Output: a solution $\mathbf{z} \in \mathbb{Z}^n$ and $l = \|\mathbf{Rz} - \mathbf{x}\|_2^2$

$LB \leftarrow \left\lceil \frac{-\sqrt{r-dist}+x_n}{r_{n,n}} \right\rceil$, $UB \leftarrow \left\lfloor \frac{\sqrt{r-dist}+x_n}{r_{n,n}} \right\rfloor$
$l \leftarrow r$, $\mathbf{z} \leftarrow \phi$ /* ϕ represents an empty vector */
if $LB \leq UB$ **then**
  for each integer s lying in $[LB, UB]$ **do**
    $newdist \leftarrow dist + (x_n - s \cdot r_{n,n})^2$
    if $newdist < l$ **then**
      $\hat{\mathbf{z}}_{in} \leftarrow [s; \mathbf{z}_{in}]$
      if $n > 1$ **then**
        $\hat{\mathbf{x}} \leftarrow \mathbf{x}(1 : n-1) - s \times \mathbf{R}(1 : n-1, n)$
        $[\mathbf{z}', l'] \leftarrow$SPH-DEC($\mathbf{R}_{n-1}$, $\hat{\mathbf{x}}$, $\hat{\mathbf{z}}_{in}$, l, $newdist$)
        if $l' < l$ **then**
          $l \leftarrow l'$, $\mathbf{z} \leftarrow \mathbf{z}'$
        end if
      else if $newdist \neq 0$ **then**
        $\mathbf{z} \leftarrow \hat{\mathbf{z}}_{in}$, $l \leftarrow newdist$
      end if
    end if
  end for
end if

Figure 2: The sphere decoding algorithm

- Choice of the initial size r. If r is too small, there is no nonzero lattice point inside the hypersphere, whereas if r is too large, there are too many lattice points to enumerate and the complexity would become very high. So we have to find a sufficient small r such that at least one lattice point lies in this hypersphere. Let $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$ be an LLL-reduced basis of the given lattice. It follows from (16) that $\|\mathbf{b}_1\|_2$ is a reasonable approximation to λ_1, thus a natural candidate for r is $\|\mathbf{b}_1\|_2^2$.

- The order in which the subproblems are solved. Once the interval $[LB, UB]$ in line 1 is obtained, the algorithms based on Phost's strategy [16, 57, 58] search through this interval from the lower bound to the upper bound. To further improve the performance, algorithms based on Schnorr-Euchner strategy [5, 15] search through $[LB, UB]$ in the order of increasing distance from the center. In this way, the chance of finding short vectors early is increased. So algorithms based on Schnorr-Euchner strategy are more efficient than those based on Phost's strategy.

Another aspect that is closely related to the efficiency of the sphere decoding algorithms is the structure of lattice basis. It is believed that the closer to being orthogonal the basis vectors are, the more efficient Algorithm Sph-Dec is. Thus, an appropriate preprocessor, such as LLL algorithm, is necessary.

The complexity of the sphere decoding algorithms was discussed in [13, 14, 60]. In [13, 14], the search process was modeled as a search tree, and the search space can be measured by the number of nodes visited in practice. Since the n-dim problem can be reduced to at most $\lfloor 2\sqrt{r}/r_{n,n} \rfloor + 1$ $(n-1)$-dim subproblems, we can obtain the upper bound for the cardinality of the search space by induction,

$$\prod_{j=1}^{n} \left(\left\lfloor \frac{2\sqrt{r}}{r_{j,j}} \right\rfloor + 1 \right). \tag{25}$$

Moreover, let the basis be LLL-reduced with the parameter $\omega \approx 1$, and $r = \|\mathbf{b}_1\|_2^2$, then it is easy to verify that (25) is bounded above by $\sqrt{4/3}^{n^2/2+O(n)}$. It is further shown in [60] that for random bases as defined in [62], the asymptotic complexity of the sphere decoding algorithms is expected to be of $1.02^{n^2+O(n)}$ polynomial-time operations. For Gaussian random matrices appeared in communications, it is proved in [13] that the expected cardinality of the search space is bounded above by $e^{\pi r}$. Since the expectation $\mathrm{E}(\|\mathbf{b}_1\|_2^2) = n$, the expected asymptotic complexity of the sphere decoding algorithms is of $e^{\pi n}$ polynomial-time operations.

3.3 The Sieve Algorithms

Rather than the deterministic algorithms as introduced above, the sieve algorithm [59] proposed by Ajtai, Kumar and Sivakumar (AKS) is a randomized method for solving SVP. It is shown in [59] that the time and space complexity of AKS are both $2^{O(n)}$. Note that the sphere decoding algorithms as well as Kannan enumeration only require a polynomial space. Thus, a big drawback of AKS is its exponential space requirement. Besides, the original AKS was widely believed to be impractical since the constant hidden in the $2^{O(n)}$ complexity was thought to be large. To further improve the efficiency of the sieve algorithm, two practical heuristic variants of AKS were proposed in [60] and [61], respectively. The first variant [60] runs in $(4/3+\varepsilon)^n$ polynomial-time using $(4/3+\varepsilon)^{n/2}$ space. For the second variant, the complexity is still unknown currently.

However, numerical results in [61] show that in practice it runs in $2^{0.48n}$ polynomial-time using $2^{0.18n}$ space.

The efficiency of the three strategies introduced in this section were compared in [60]. In general, for lattices of small dimensions, the sphere decoding using Schnorr-Euchner enumeration is the fastest. On one hand, for lattices of rather small dimensions, the exponential factor of the complexity of the sphere decoding algorithms may not be larger than that of Kannan's algorithm or that of the sieve algorithms. On the other hand, the polynomial factors of the complexity of these algorithms are not negligible for low dimensions. The polynomial-time unit of the sphere decoding algorithms is extremely small in practice, whereas the polynomial-time units of the other two methods are much larger (see [60] for more details). Simulation results in [60, 61] illustrate that for lattices of dimension $n \leq 40$, Schnorr-Euchner enumeration is indeed the most efficient algorithm.

4 A New Algorithm for Constructing HKZ-reduced Bases

As pointed out previously, algorithms based on Kannan's strategy [24–27] not only find a shortest nonzero lattice point, but also construct an HKZ-reduced basis simultaneously. However, they are intended as theoretical results rather than practical tools, since the induction conditions imposed by Kannan's strategy are crucial and the complexity becomes prohibitive quickly as the dimension of lattices increases.

From the definition of HKZ-reduction, the key to the construction of an HKZ-reduced basis is to recursively find a shortest nonzero lattice vector and then to extend this vector to a basis for the lattice. From Section 3, the sphere decoding algorithm using Schnorr-Euchner enumeration is currently the most efficient method for solving general SVP with small dimensions. Therefore, to calculate an HKZ-reduced basis efficiently, it is natural to combine Schnorr-Euchner enumeration and Kannan's basis expansion method [24]. Indeed, this is the method presented in [5].

In this section, we present a new algorithm for constructing HKZ-reduced bases for general lattices. Like the algorithm in [5], we also adopt Schnorr-Euchner enumeration to solve SVP. However, instead of Kannan's basis expansion method, we use a novel unimodular transformation basis expansion strategy.

Firstly, we state Kannan's basis expansion method [26] in Fig. 3. A brief complexity analysis of Procedure Select-Basis is given as follows. From line 14, this algorithm is performed in a recursive way and each time the problem size is reduced by one. In each recursion k, $1 \leq k \leq n$, the computations in lines 4 and 7 involve solving k-dim systems of linear equations which require $O(k^3)$ operations. The computations in lines 13 and 15 require $O(k^2)$ operations. Moreover, the computations in lines 8–10 also lead to some additional operations. In summary, the complexity of Procedure Select-Basis is at least $O(n^4)$ ($O(\sum_{k=1}^{n} k^3)$). Although the sphere decoding algorithm has an exponential complexity, numerical results in [13] show that for small dimensions, the expected complexity can be approximated by a polynomial function (often roughly cubic). Hence, from a practical point of view, the computational cost required by Kannan's basis expansion method is not negligible. Moreover, note that Procedure Select-Basis only works for rational lattices, not general real-valued lattices.

Secondly, based on the unimodular transformation presented in [45], we propose a new basis expansion method, which is applicable for lattices of any type, as long as the coordinates of one shortest nonzero lattice point is available. Specifically, let $\mathbf{B} \in \mathbb{R}^{m \times n}$ be a generator matrix for

Procedure SELECT-BASIS(n; $\mathbf{b}_1, \cdots, \mathbf{b}_{n+1}$)

Input: vectors $\mathbf{b}_i \in \mathbb{Q}^m$, $1 \le i \le n+1$, that can span an n-dim lattice L
Output: a basis $\{\mathbf{a}_1, \cdots, \mathbf{a}_n\}$ of L, where $\mathbf{a}_1$ is a shortest lattice vector in the direction of $\mathbf{b}_1$

if $n = 0$ or $\mathbf{b}_1 = 0$ **then**
  do the obvious
else
  if $\mathbf{b}_1$ is independent of $\mathbf{b}_2, \cdots, \mathbf{b}_{n+1}$ **then**
    $\mathbf{a}_1 \leftarrow \mathbf{b}_1$
  else
    find $\alpha_2, \cdots, \alpha_{n+1}$ (rationals) such that $\sum_{j=2}^{n+1} \alpha_j \mathbf{b}_j = \mathbf{b}_1$
    $M \leftarrow$ least common multiples of the denominators of $\alpha_2, \cdots, \alpha_{n+1}$
    $\gamma \leftarrow \gcd(M\alpha_2, \cdots, M\alpha_{n+1})$
    let $M/\gamma = p/q$, where p, q are relatively prime integers
    $\mathbf{a}_1 \leftarrow (1/q) \cdot \mathbf{b}_1$
  end if
  $\overline{\mathbf{b}}_i \leftarrow \mathbf{b}_i - \frac{\langle \mathbf{b}_i, \mathbf{a}_1 \rangle}{\langle \mathbf{a}_1, \mathbf{a}_1 \rangle} \mathbf{a}_1, \quad i = 2, \cdots, n+1$
  $(\mathbf{c}_2, \cdots, \mathbf{c}_n) \leftarrow$ SELECT-BASIS($n-1; \overline{\mathbf{b}}_2, \cdots, \overline{\mathbf{b}}_{n+1}$)
  lift $\mathbf{c}_i$ to $\mathbf{a}_i$ in L for $i = 2, \cdots, n$
  return $\mathbf{a}_1, \cdots, \mathbf{a}_n$
end if

Figure 3: Kannan's basis expansion method [26]

an n-dim lattice L. Suppose that $\mathbf{Bz}$ is a shortest nonzero point in L, where $\mathbf{z} = [z_i] \in \mathbb{Z}^n$. Then the problem of expanding $\mathbf{Bz}$ to a basis for L is equivalent to the problem of constructing an n-by-n unimodular matrix $\mathbf{Z}$ whose first column is $\mathbf{z}$. In other words, $\mathbf{Z}^{-1}\mathbf{z} = \mathbf{e}_1$, which says that $\mathbf{Z}^{-1}$, also unimodular, transforms $\mathbf{z}$ into the first unit vector $\mathbf{e}_1$.

For the special case when $n = 2$, such a unimodular matrix is easy to construct. Suppose that $\mathbf{z} = [p, q]^{\mathrm{T}} \in \mathbb{Z}^2$, and let $\gcd(p, q) = d$. Using the extended Euclidean algorithm, one can find integers a and b such that $ap + bq = d$. Construct

$$\mathbf{M} = \begin{bmatrix} p/d & -b \\ q/d & a \end{bmatrix}. \tag{26}$$

It is obvious that $\mathbf{M}$ is a unimodular matrix with

$$\mathbf{M}^{-1} = \begin{bmatrix} a & b \\ -q/d & p/d \end{bmatrix}, \quad \mathbf{M}^{-1} \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix}. \tag{27}$$

Thus, $\mathbf{M}^{-1}$ can be applied to $\mathbf{z}$ to annihilate its second entry. In particular, if $\gcd(p, q) = \pm 1$, then $\mathbf{z}$ can be transformed into the first unit vector.

Now we consider the general case when $n > 2$. Since $\mathbf{Bz}$ is a shortest nonzero lattice point, we have $\gcd(z_i) = \pm 1$, implying that a sequence of the plane unimodular transformations $\mathbf{M}$ of the form (26) can be applied to transform $\mathbf{z}$ into the first unit vector.

Putting all things together, we present our new algorithm for constructing an HKZ reduced basis in Fig. 4. During the process of Algorithm HKZ-RED, Procedure TRANSFORM called

Algorithm HKZ-RED($\mathbf{B}$, ω)

Input: $\mathbf{B} \in \mathbb{R}^{m\times n}$, and the LLL parameter ω, $1/4 < \omega < 1$
Output: a unimodular matrix $\mathbf{Z} \in \mathbb{Z}^{n\times n}$ such that the columns of $\mathbf{BZ}$ form an HKZ-reduced basis

1: QR decomposition: $\mathbf{B} = \mathbf{QR}$
2: $\mathbf{Z} \leftarrow \mathbf{I}_n$
3: **for** $k = 1$ to $n-1$ **do**
4: use LLL-aided Schnorr-Euchner enumeration to find a vector $\mathbf{z} \in \mathbb{Z}^{n-k+1}$ such that $\mathbf{R}(k:n, k:n)\mathbf{z}$ is a shortest nonzero point in the lattice generated by $\mathbf{R}(k:n, k:n)$
5: $[\mathbf{R}, \mathbf{Z}] \leftarrow$ TRANSFORM($\mathbf{R}$, $\mathbf{Z}$, $\mathbf{z}$, k)
6: **end for**
7: **for** $j = 2$ to n **do**
8: **for** $i = j-1$ down to 1 **do**
9: $[\mathbf{R}, \mathbf{Z}] \leftarrow$ SIZE-REDUCE ($\mathbf{R}$, $\mathbf{Z}$, i, j)
10: **end for**
11: **end for**

Figure 4: The new HKZ-reduction algorithm

in line 5 expands the shortest lattice vector found in line 4 to a basis for the $(n-k+1)$-dim lattice generated by the trailing submatrix $\mathbf{R}(k:n, k:n)$. Moreover, this procedure should keep the upper triangular structure of $\mathbf{R}$ and update the unimodular matrix $\mathbf{Z}$. Fig. 5 is an implementation of the procedure.

Now we analyze the complexity of Procedure TRANSFORM. For each iteration of the for-loop, the computations from line 5 to line 8 require $O(n)$ fp operations. Then we consider the cost of line 2. Given two integers p and q, it is well known that the complexity of Euclidean algorithm is $O(\log g)$, where $g = \min\{|p|, |q|\}$. So the cost of line 2 can be obtained if an upper bound of $|z_{n-k+1}|$ can be found. Suppose that $\mathbf{R}$ is LLL-reduced with a parameter $\omega = 3/4$. It follows from [35] that $r_{k,k}^2 \le 2^{n-k} r_{n,n}^2$, which implies that in the k-th iteration, the initial radius of the sphere decoding algorithm is bounded by $2^{(n-k)/2} r_{n,n}$. Consequently, setting ρ in (21) to $2^{(n-k)/2} r_{n,n}$, we have $|z_{n-k+1}| \le 2^{(n-k)/2}$. Thus, the complexity of line 2 is $O(n-k)$, and therefore the total cost of Procedure TRANSFORM is $O(n(n-k))$. In particular, when the size of $\mathbf{z}$ is n, the complexity is $O(n^2)$. Hence, Procedure TRANSFORM is much more efficient than Procedure SELECT-BASIS, whose complexity is at least $O(n^4)$.

5 New Algorithm for Computing Minkowski Reduced Bases: I

Among all reduction notions, M-reduction is perhaps the most intuitive and strongest one, and up to dimension four, M-reduction is better than any other known reduction, because it can exactly reach Minkowski's successive minima. In 1773, Lagrange [29] presented the first algorithm for constructing M-reduced bases for lattices of dimension two. Recently, this algorithm was extended to dimensions three and four by Semaev [30] and Nguyen and Stehlé [31], respectively. More generally, Helfrich [25] and Afflerbach and Grothe [32] presented algorithms for constructing M-reduced bases for lattices of arbitrary dimension.

Procedure TRANSFORM($\mathbf{R}$, $\mathbf{Z}$, $\mathbf{z}$, k)

Input: an upper triangular $\mathbf{R} \in \mathbb{R}^{n\times n}$, a unimodular $\mathbf{Z} \in \mathbb{Z}^{n\times n}$, $\mathbf{z} \in \mathbb{Z}^{n-k+1}$, and the index k
Output: the updated $\mathbf{R} = [r_{i,j}]$ with $r_{k,k} = \|\mathbf{R}(k:n, k:n)\mathbf{z}\|_2$, and the updated $\mathbf{Z} \in \mathbb{Z}^{n\times n}$

1: **for** $j = n - k + 1$ down to 2 **do**
2: $d \leftarrow \gcd(z_{j-1}, z_j)$, and find integers a and b such that $az_{j-1} + bz_j = d$
3: $\mathbf{M} \leftarrow \begin{bmatrix} z_{j-1}/d & -b \\ z_j/d & a \end{bmatrix}$
4: $z_{j-1} \leftarrow d$
5: $\mathbf{R}(1 : j+k-1, j+k-2 : j+k-1) \leftarrow \mathbf{R}(1 : j+k-1, j+k-2 : j+k-1) \cdot \mathbf{M}$
6: find a 2×2 Givens matrix $\mathbf{G}$ such that element $\mathbf{R}(j+k-1, j+k-2)$ can be annihilated by $\mathbf{G}$
7: $\mathbf{R}(j+k-2 : j+k-1, j+k-2 : n) \leftarrow \mathbf{G} \cdot \mathbf{R}(j+k-2 : j+k-1, j+k-2 : n)$
8: $\mathbf{Z}(:, j+k-2 : j+k-1) \leftarrow \mathbf{Z}(:, j+k-2 : j+k-1) \cdot \mathbf{M}$
9: **end for**

Figure 5: The new basis expansion method

Before the discussion of the algorithms in [25, 32] and our new algorithm, we state a result which plays a central role in the construction of M-reduced bases.

Lemma 1 ([52]) *Let $\mathbf{B} = [\mathbf{b}_1, \cdots, \mathbf{b}_n] \in \mathbb{R}^{m\times n}$ and let L be the lattice generated by $\mathbf{B}$. For a vector $\mathbf{v} = \sum_{i=1}^{n} t_i\mathbf{b}_i$ and any index p, $1 \le p \le n$, there exists a basis for L containing $\{\mathbf{b}_1, \cdots, \mathbf{b}_{p-1}, \mathbf{v}\}$ if and only if $\gcd(t_p, \cdots, t_n) = 1$.*

Given an n-dim lattice L, suppose that $\mathbf{B}_p$ is a generator matrix of L such that the first $p-1$ columns of $\mathbf{B}_p$ can be extended to an M-reduced basis for L. Then it follows from Lemma 1 that the p-th M-reduced basis vector $\mathbf{m}_p$, which can be extended to an M-reduced basis with the first $p-1$ columns of $\mathbf{B}_p$, must satisfy

$$\|\mathbf{m}_p\|_2 = \min\{\|\mathbf{B}_p\mathbf{z}\|_2 : \ \mathbf{z} \in \mathbb{Z}^n, \ \gcd(z_p, \cdots, z_n) = 1\}. \tag{28}$$

It is shown in [32] that for lattices of dimension less than 8, the condition (28) can be reduced to

$$\|\mathbf{m}_p\|_2 = \min\{\|\mathbf{B}_p\mathbf{z}\|_2 : \ \mathbf{z} \in \mathbb{Z}^n, \ z_p = 1\}. \tag{29}$$

Obviously, the minimization problem (28) can be viewed as an SVP with the constraint $\gcd(z_p, \cdots, z_n) = 1$. Therefore, (28) or (29) can be solved by incorporating such gcd constraint into the SVP solvers introduced in Section 3. Effort in this direction was firstly taken by Helfrich [25]. Briefly speaking, a variant of Kannan's strategy [24] was proposed in [25] to solve (28). Unfortunately, this variant is more complicated and time-consuming than the original Kannan's strategy, since it associates with solving roughly $(5/4)^{n^3/(4-o(1))}$ $(p-1)$-dim CVPs. Hence, like Kannan's algorithm [24, 26], Helfrich's algorithm is also intended as a theoretical result rather than a practical tool.

The algorithm presented in [32] constructs an M-reduced basis in a quite different way. Starting from $p = 1$, this algorithm first performs Phost enumeration [16, 57], and during the search process, whenever an intermediate lattice point $\mathbf{B}_p\mathbf{z}$ inside the search region satisfying

$\gcd(z_p, \cdots, z_n) = 1$ ($n > 7$) or $z_p = 1$ ($n \leq 7$) is found, the p-th column of $\mathbf{B}_p$ is then replaced by $\mathbf{B}_p\mathbf{z}$ and the algorithm is restarted from $p = 1$. On the other hand, if the p-th column of $\mathbf{B}_p$ is already the shortest lattice point satisfying the corresponding gcd constraint, we set $p = p+1$ and repeat the above process. The algorithm terminates when $p = n + 1$.

Note that the number of lattice points enumerated by Phost's strategy grows exponentially with the dimension n. Therefore, in practice the algorithm in [32] is restarted many times, and the complexity becomes prohibitive quickly as the dimension increases. Furthermore, we have found that for lattices of dimension $n > 7$, this algorithm may fail. For instance, during Phost enumeration, if a lattice point $\mathbf{B}_p\mathbf{z}$ inside the search region with $\gcd(z_p, \cdots, z_n) = 1$ is found, the p-th column of $\mathbf{B}_p$ is then replaced by $\mathbf{B}_p\mathbf{z}$ to obtain a new basis $\mathbf{B}'$. Note that if $z_p \neq \pm 1$, then $\mathbf{B}_p$ and $\mathbf{B}'$ do not generate the same lattice. Consequently, the algorithm fails.

In this section, we shall present a practical algorithm for constructing M-reduced bases for general lattices. Differing from the algorithm in [32], the proposed new algorithm is based on Schnorr-Euchner enumeration [15] and is also valid for lattices of dimensions higher than 7.

For clarity, as Algorithm HKZ-Red in Section 4, the new algorithm is presented in a iterative way. Apparently, the first M-reduced basis vector $\mathbf{m}_1$ is a shortest nonzero lattice vector in L, which can be obtained by applying Schnorr-Euchner enumeration [5, 15]. We can extend $\mathbf{m}_1$ to a basis for L by calling Procedure Transform. Now, suppose that a basis $\{\mathbf{m}_1, \cdots, \mathbf{m}_{p-1}, \mathbf{b}_p, \cdots, \mathbf{b}_n\}$, $1 < p \leq n$, has been obtained, to extend $\{\mathbf{m}_1, \cdots, \mathbf{m}_{p-1}\}$ to an M-reduced basis for L, we have to solve the following two problems:

- Constructing the p-th M-reduced basis vector $\mathbf{m}_p$.
- Extending $\{\mathbf{m}_1, \cdots, \mathbf{m}_p\}$ to a basis for L.

From (28), $\mathbf{m}_p$ can be obtained by incorporating the constraint $\gcd(z_p, \cdots, z_n) = 1$ into Schnorr-Euchner enumeration. Instead of the length of the first column of the basis matrix, we use the length of the p-th column as the initial size of search region, so that at least one lattice point satisfying such gcd constraint lies inside the search region. To further accelerate the search process, LLL algorithm can be applied as a preprocessor. Putting all things together, we present the algorithm for calculating $\mathbf{m}_p$ in Fig. 6.

As shown in Fig. 6, Procedure M-Decode-1 is a wrapper function. It calls Procedure M-Search-1, which finds a solution for a more general problem: a CVP with the constraint $\gcd(z_p, ..., z_n) = 1$. Like Algorithm Sph-Dec, we present a recursive version of this procedure in Fig. 7.

As shown in Fig. 7, Procedure M-Search-1 is based on Schnorr-Euchner enumeration. The main difference between it and the original Schnorr-Euchner enumeration is the way of updating the search radius. Specifically, from lines 16–18, Procedure M-Search-1 updates the search radius when a shorter lattice vector satisfying the gcd constraint is found, whereas Schnorr-Euchner enumeration updates the search radius whenever a shorter lattice vector is found. Due to the additional gcd constraint, the search space of Procedure M-Search-1 is expected to be larger than that of the original Schnorr-Euchner enumeration. Moreover, the computations in lines 15 and 16 lead to some additional complexity. Thus, in practice, Procedure M-Search-1 costs more than Schnorr-Euchner enumeration. For the complexity of Procedure M-Search-1, one can obtain from Section 3 that for Gaussian random matrices appeared in communications, the expected cardinality of the search space is bounded above by $e^{\pi r}$. Since the expectation

Procedure M-DECODE-1($\mathbf{R}$, ω, p)

Input: $\mathbf{R} \in \mathbb{R}^{n\times n}$, the LLL parameter ω, and an index p, $1 \le p \le n$
Output: a vector $\mathbf{z} \in \mathbb{Z}^n$ such that $\mathbf{Rz}$ is a shortest lattice point with $\gcd(z_p, \cdots, z_n) = 1$

if $n = 1$ **then**
  return $\mathbf{z} = 1$
else
  set the initial size $r \leftarrow \|\mathbf{R}(:, p)\|_2^2$
  utilize LLL algorithm to find a unimodular $\mathbf{Z}$ and an upper triangular matrix $\mathbf{R}_{new}$ such that $\mathbf{RZ}$ is LLL-reduced and $\mathbf{R}_{new}$ is the R-factor of $\mathbf{RZ}$
  $[\mathbf{z}, l] \leftarrow$M-SEARCH-1($\mathbf{R}_{new}$, $\mathbf{Z}$, $\mathbf{0}$, ϕ, r, 0, p)
end if

Figure 6: The first algorithm for calculating each M-reduced basis vector

Procedure M-SEARCH-1($\mathbf{R}$, $\mathbf{Z}$, $\mathbf{x}$, $\mathbf{z}_{in}$, r, $dist$, p)

Input: $\mathbf{R} \in \mathbb{R}^{n\times n}$, $\mathbf{Z} \in \mathbb{Z}^{n\times n}$, a vector $\mathbf{x} \in \mathbb{R}^n$ to decode, an integral partial solution $\mathbf{z}_{in}$, the current distance record r, the distance to examined layer $dist$, and an index p, $1 \le p \le n$
Output: a vector $\mathbf{z} \in \mathbb{Z}^n$ such that $\mathbf{RZ}^{-1}\mathbf{z}$ is a closest lattice point to $\mathbf{x}$ satisfying $\gcd(z_p, ..., z_n) = 1$, and $l = \|\mathbf{RZ}^{-1}\mathbf{z} - \mathbf{x}\|_2^2$

$LB \leftarrow \left\lceil \frac{-\sqrt{r-dist}+x_n}{r_{n,n}} \right\rceil$, $UB \leftarrow \left\lfloor \frac{\sqrt{r-dist}+x_n}{r_{n,n}} \right\rfloor$
$l \leftarrow r$, $\mathbf{z} \leftarrow \phi$
if $LB \le UB$ **then**
  for each integer s in the order of increasing distance from the center of $[LB, UB]$ **do**
    $newdist \leftarrow dist + (x_n - s \cdot r_{n,n})^2$
    if $newdist < l$ **then**
      $\hat{\mathbf{z}}_{in} \leftarrow [s; \mathbf{z}_{in}]$
      if $n > 1$ **then**
        $\hat{\mathbf{x}} \leftarrow \mathbf{x}(1 : n-1) - s \times R(1 : n-1, n)$
        $[\mathbf{z}', l'] \leftarrow$M-SEARCH-1($\mathbf{R}_{n-1}$, $\mathbf{Z}$, $\hat{\mathbf{x}}$, $\hat{\mathbf{z}}_{in}$, l, $newdist$, p)
        if $l' < l$ **then**
          set $l \leftarrow l'$, $\mathbf{z} \leftarrow \mathbf{z}'$
        end if
      else
        $\mathbf{z} \leftarrow \mathbf{Z} \cdot \hat{\mathbf{z}}_{in}$
        if $\gcd(z_p, ..., z_n) = 1$ **then**
          set $l \leftarrow newdist$
        end if
      end if
    else
      return $\mathbf{z}$ and l
    end if
  end for
end if

Figure 7: The first algorithm for solving CVP with the gcd constraint

Algorithm M-RED-1($\mathbf{B}$, ω)

Input: $\mathbf{B} \in \mathbb{R}^{m\times n}$, and the LLL parameter ω, $1/4 < \omega < 1$
Output: a unimodular $\mathbf{Z} \in \mathbb{Z}^{n\times n}$ such that the columns of $\mathbf{BZ}$ form an M-reduced basis
1: QR decomposition: $\mathbf{B} = \mathbf{QR}$
2: $\mathbf{Z} \leftarrow \mathbf{I}_n$
3: **for** $k = 1$ to n **do**
4: $\quad \mathbf{z} \leftarrow$M-DECODE-1($\mathbf{R}$, ω, k)
5: $\quad [\mathbf{R}, \mathbf{Z}] \leftarrow$ TRANSFORM($\mathbf{R}$, $\mathbf{Z}$, $\mathbf{z}$, k)
6: $\quad \mathbf{R}(1:k-1,k) \leftarrow \mathbf{R}(1:k-1,k) + \mathbf{R}(1:k-1,1:k-1)\cdot[z_1,\cdots,z_{k-1}]^{\mathrm{T}}$
7: $\quad \mathbf{Z}(:,k) \leftarrow \mathbf{Z}(:,k) + \mathbf{Z}(:,1:k-1)\cdot[z_1,\cdots,z_{k-1}]^{\mathrm{T}}$
8: **end for**

Figure 8: The first M-reduction algorithm

$\mathrm{E}(\|\mathbf{b}_p\|_2^2) = n$, for any $1 \le p \le n$, the expected asymptotic complexity of Procedure M-SEARCH-1 is of $e^{\pi n}$ polynomial-time operations.

Once the p-th M-reduced basis vector $\mathbf{m}_p = \mathbf{B}_p\mathbf{z}$ is found, the second problem is to extend $\{\mathbf{m}_1,\cdots,\mathbf{m}_p\}$ to a basis for L. In terms of matrices, it is to find a unimodular matrix $\mathbf{Z}$ such that

$$\mathbf{B}_{p+1} = \mathbf{B}_p\mathbf{Z}, \tag{30}$$

which implies that the first $p-1$ columns of $\mathbf{Z}$ are the first $p-1$ unit vectors $\mathbf{e}_i$, $i = 1, ..., p-1$, and the p-th column of $\mathbf{Z}$ is the integer vector $\mathbf{z}$ found by Procedure M-DECODE-1, so that the first $p-1$ columns of $\mathbf{B}_{p+1}$ equal the first $p-1$ columns $\mathbf{m}_1, ..., \mathbf{m}_{p-1}$ of $\mathbf{B}_p$ and the p-th column of $\mathbf{B}_{p+1}$ is $\mathbf{m}_p = \mathbf{B}_p\mathbf{z}$ as desired. Since $\gcd(z_p, ..., z_n) = 1$, from the discussion in Section 4, one can construct a unimodular matrix $\mathbf{M}_p$ whose first column is $[z_p, ..., z_n]^{\mathrm{T}}$. Now consider the two $n \times n$ unimodular matrices

$$\mathbf{Z}_1 = \begin{bmatrix} \mathbf{I}_{p-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_p \end{bmatrix}, \quad \mathbf{Z}_2 = \begin{bmatrix} & z_1 & & \\ \mathbf{I}_{p-1} & \vdots & \mathbf{0} & \\ & z_{p-1} & & \\ & 1 & & \\ & & \ddots & \\ \mathbf{0} & & & 1 \end{bmatrix} \tag{31}$$

We claim that the product $\mathbf{Z}_1\mathbf{Z}_2$ is a unimodular matrix satisfying (30). Indeed, $\mathbf{Z}_1\mathbf{Z}_2$ is unimodular since both $\mathbf{Z}_1$ and $\mathbf{Z}_2$ are unimodular. From (31), the first $p-1$ columns of $\mathbf{Z}_1\mathbf{Z}_2$ are the first $p-1$ unit vectors and the p-th column of $\mathbf{Z}_1\mathbf{Z}_2$ is $\mathbf{z} = [z_1, ..., z_n]^{\mathrm{T}}$.

The application of $\mathbf{Z}_1$ can be performed by Procedure TRANSFORM and the application of $\mathbf{Z}_2$ is the calculation of a linear combination of the first p columns. Putting all things together, the new algorithm for constructing M-reduced bases for general lattices is presented in Fig. 8.

Procedure M-SEARCH-2($\mathbf{R}$, $\mathbf{x}$, $\mathbf{z}_{in}$, r, $dist$, p)

Input: $\mathbf{R} \in \mathbb{R}^{n\times n}$, a vector $\mathbf{x} \in \mathbb{R}^n$ to decode, an integral partial solution $\mathbf{z}_{in}$, the current distance record r, the distance to examined layer $dist$, and an index p, $1 \le p \le n$

Output: a vector $\mathbf{z} \in \mathbf{Z}^n$ such that $\mathbf{Rz}$ is a closest lattice point to $\mathbf{x}$ satisfying $\gcd(z_p, ..., z_n) = 1$, and $l = \|\mathbf{Rz} - \mathbf{x}\|_2^2$

1: $LB \leftarrow \left\lceil \frac{-\sqrt{r-dist}+x_n}{r_{n,n}} \right\rceil$, $UB \leftarrow \left\lfloor \frac{\sqrt{r-dist}+x_n}{r_{n,n}} \right\rfloor$
2: $l \leftarrow r$, $\mathbf{z} \leftarrow \phi$
3: **if** $LB \le UB$ **then**
4: **for** each integer s in the order of increasing distance from the center of $[LB, UB]$ **do**
5: $newdist \leftarrow dist + (x_n - s \cdot r_{n,n})^2$
6: **if** $newdist < l$ **then**
7: $\hat{\mathbf{z}}_{in} \leftarrow [s; \mathbf{z}_{in}]$
8: **if** $n \neq p$ **or** ($n = p$ **and** $\gcd(\hat{\mathbf{z}}_{in}) = 1$) **then**
9: **if** $n > 1$ **then**
10: $\hat{\mathbf{x}} \leftarrow \mathbf{x}(1 : n-1) - s \times R(1 : n-1, n)$
11: $[\mathbf{z}', l'] \leftarrow$M-SEARCH-2($\mathbf{R}_{n-1}$, $\hat{\mathbf{x}}$, $\hat{\mathbf{z}}_{in}$, l, $newdist$, p)
12: **if** $l' < l$ **then**
13: set $l \leftarrow l'$, $\mathbf{z} \leftarrow \mathbf{z}'$
14: **end if**
15: **else**
16: $\mathbf{z} \leftarrow \hat{\mathbf{z}}_{in}$, $l \leftarrow newdist$
17: **end if**
18: **end if**
19: **else**
20: **return** $\mathbf{z}$ and l
21: **end if**
22: **end for**
23: **end if**

Figure 9: The second algorithm for solving CVP with the gcd constraint

6 New Algorithm for Computing Minkowski Reduced Bases: II

From the discussion in Section 5, the search space of Procedure M-SEARCH-1 is larger than that of the original Schnorr-Euchner enumeration. This motivates us to design a more efficient way to calculate each M-reduced basis vector. Our idea is to impose the constraint as early as possible to reduce the number of points to be searched. Clearly, $\gcd(z_p, ..., z_n)$ can be calculated as soon as $z_p, ..., z_n$ are available. Note that during the process of Schnorr-Euchner enumeration, a solution is built bottom-up, from z_n to z_1, thus the gcd condition can be checked at level p, instead of level 1 as in Procedure M-SEARCH-1. Fig. 9 shows an implementation of the above idea.

Like Schnorr-Euchner enumeration, one can obtain from line 16 that the above procedure updates the search radius whenever a shorter lattice vector is found. Moreover, as shown in line 8, all $(p-1)$-dim subproblems indexed by those $\mathbf{z}$ not satisfying $\gcd(z_p, ..., z_n) = 1$ are excluded from the search process. Consequently, the search space of Procedure M-SEARCH-2 is expected

to be drastically reduced from the original Schnorr-Euchner enumeration.

However, one drawback of Procedure M-SEARCH-2 is: LLL algorithm cannot be used as its preprocessor to accelerate the search process. Specifically, from line 15 of Procedure M-SEARCH-1, to check the gcd condition, the unimodular matrix obtained from LLL algorithm must be applied to the solution vector firstly. Unfortunately, the application of the unimodular matrix requires a complete n-vector, whereas at level p, where $\gcd(z_p, ..., z_n)$ is calculated in Procedure M-SEARCH-2, only a subvector $\mathbf{z}(p:n)$ is available. To alleviate the problem, we propose a new lattice reduction technique to accelerate Procedure M-SEARCH-2.

Consider an $n \times n$ unimodular matrix $\mathbf{Z}$ with the following structure:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{D} & \mathbf{E} \\ \mathbf{0}^{(n-p+1)\times(p-1)} & \mathbf{F} \end{bmatrix}, \tag{32}$$

where $\mathbf{D}$, $\mathbf{E}$, and $\mathbf{F}$ have proper dimensions. Then both $\mathbf{D}$ and $\mathbf{F}$ are unimodular. If an integer vector $\hat{\mathbf{z}}$ satisfies $\gcd(\hat{\mathbf{z}}(p), ..., \hat{\mathbf{z}}(n)) = 1$, then the integer vector $\mathbf{z} = \mathbf{Z}\hat{\mathbf{z}}$ also satisfies the condition $\gcd(\mathbf{z}(p), ..., \mathbf{z}(n)) = 1$, since $\mathbf{z}(p:n) = \mathbf{F}\hat{\mathbf{z}}(p:n)$ and $\mathbf{F}$ is unimodular. Thus, if a appropriate unimodular matrix $\mathbf{Z}$ with the form (32) is chosen as a preprocessor for Procedure M-SEARCH-2, the information of the subvector $\hat{\mathbf{z}}(p:n)$ obtained at level p is sufficient to check the gcd condition of the solution $\mathbf{z} = \mathbf{Z}\hat{\mathbf{z}}$.

Suppose now that the first $p-1$ columns $\mathbf{m}_1, ..., \mathbf{m}_{p-1}$ of the current basis matrix $\mathbf{B}_p$ can be extended to an M-reduced basis. Let $\mathbf{R}$ be the R-factor of $\mathbf{B}_p$. Then it is obvious that the first $p-1$ columns of $\mathbf{R}$ is M-reduced. Thus the submatrix $\mathbf{D}$ in (32) can be chosen as $\mathbf{I}_{p-1}$. In other words, we only need to reduce the submatrix of $\mathbf{R}$ consisting of its last $n-p+1$ columns. A natural approach is to select the submatrices $\mathbf{E}$ and $\mathbf{F}$ appropriately such that after preprocessing, $\mathbf{R}(p:n, p:n)$ is LLL-reduced and all off-diagonal entries of $\mathbf{R}$ belonging to the last $n-p+1$ columns are size-reduced. Fig. 10 shows an implementation of this idea.

Since Procedure PARTIAL-LR only involves the last $n-p+1$ columns of $\mathbf{R}$, it always costs less than LLL algorithm for any $1 < p \leq n$, and the same as LLL algorithm when $p = 1$. Combining Procedure PARTIAL-LR and Procedure M-SEARCH-2 together, we present the algorithm for calculating $\mathbf{m}_p$ in Fig. 11.

Finally, the second algorithm M-RED-2 for constructing an M-reduced basis can be obtained by simply replacing Procedure M-DECODE-1 called in line 4 of Algorithm M-RED-1 with Procedure M-DECODE-2.

7 Performance Analysis

In this section, we firstly compare the theoretical upper bounds on the orthogonality defect of LLL, HKZ, and M-reduced bases. Then after a brief review of existing results on the proximity factors of approximate lattice decoding [6, 20, 21], we give new improved upper bounds for the proximity factors of LLL-reduction-aided SIC decoding and LLL-reduction-aided ZF decoding. Also, we derive upper bounds for the proximity factors of both M-reduction-aided SIC decoding and M-reduction-aided ZF decoding. Thus, like LLL-reduction and HKZ-reduction, approximate decoding algorithms aided by M-reduction can also achieve the same diversity order with ILD.

Procedure PARTIAL-LR($\mathbf{R}$, ω, p)

Input: $\mathbf{R} \in \mathbb{R}^{n \times n}$, the LLL parameter ω, and an index p, $1 \leq p \leq n$
Output: the updated $\mathbf{R}$ and a unimodular matrix $\mathbf{Z}$ such that the last $n - p + 1$ columns of $\mathbf{R}$ are reduced

1: $\mathbf{Z} \leftarrow \mathbf{I}_n$
2: $k \leftarrow p + 1$
3: **while** $k \leq n$ **do**
4: $[\mathbf{R}, \mathbf{Z}] \leftarrow$ SIZE-REDUCE ($\mathbf{R}$, $\mathbf{Z}$, $k-1$, k)
5: **if** $r_{k-1,k}^2 + r_{k,k}^2 \leq \omega \cdot r_{k-1,k-1}^2$ **then**
6: find a Givens matrix $\mathbf{G}$ such that $G \cdot \begin{bmatrix} r_{k-1,k} \\ r_{k,k} \end{bmatrix} = \begin{bmatrix} \times \\ 0 \end{bmatrix}$
7: swap columns $k-1$ and k in $\mathbf{R}$ and $\mathbf{Z}$
8: $\mathbf{R}(k-1:k, k-1:n) \leftarrow \mathbf{G} \cdot \mathbf{R}(k-1:k, k-1:n)$
9: $k \leftarrow \max\{k-1, p+1\}$
10: **else**
11: $k \leftarrow k + 1$
12: **end if**
13: **end while**
14: **for** $j = p$ to n **do**
15: **for** $i = j - 1$ down to 1 **do**
16: $[\mathbf{R}, \mathbf{Z}] \leftarrow$ SIZE-REDUCE ($\mathbf{R}$, $\mathbf{Z}$, i, j)
17: **end for**
18: **end for**

Figure 10: A partial lattice reduction algorithm

Procedure M-DECODE-2($\mathbf{R}$, ω, p)

Input: $\mathbf{R} \in \mathbb{R}^{n \times n}$, the LLL parameter ω, and an index p, $1 \leq p \leq n$
Output: a vector $\mathbf{z} \in \mathbb{Z}^n$ such that $\mathbf{Rz}$ is a shortest lattice point with $\gcd(z_p, \cdots, z_n) = 1$

1: **if** $n = 1$ **then**
2: **return** $\mathbf{z} = 1$
3: **else**
4: $[\mathbf{R}_{new}, \mathbf{Z}] \leftarrow$PARTIAL-LR($\mathbf{R}$, ω, p)
5: set the initial size $r \leftarrow \|\mathbf{R}_{new}(:, p)\|_2^2$
6: $[\mathbf{z}', l] \leftarrow$M-SEARCH-2($\mathbf{R}_{new}$, $\mathbf{0}$, ϕ, r, 0, p)
7: $\mathbf{z} \leftarrow \mathbf{Z}\mathbf{z}'$
8: **end if**

Figure 11: The second algorithm for calculating each M-reduced basis vector

Table 1: Upper Bounds of Orthogonality Defect of HKZ, LLL ($\omega = 3/4$), and M-reduced Bases

n	2	3	4	5	6	7	8	24
γ_n	$2/\sqrt{3}$	$2^{1/3}$	$\sqrt{2}$	$8^{1/5}$	$(64/3)^{1/6}$	$64^{1/7}$	2	4
$\delta_{H,n}$	1.291	1.937	3.623	7.246	17.75	48.61	161.2	4.26×10^{13}
$\delta_{L,n}$	1.414	2.828	8	32	181.0	1.45×10^3	1.64×10^4	3.48×10^{41}
$\delta_{M,n}$	1.155	1.414	2	3.162	6.455	15.63	48.83	2.51×10^{17}

7.1 Orthogonality Defect

As pointed out previously, the orthogonality defect is a commonly used indicator to reveal the degree of orthogonality for a given lattice basis. Denote $\delta_{H,n}$, $\delta_{L,n}$, and $\delta_{M,n}$ the upper bounds of the orthogonality defect over all $n \times n$ HKZ, LLL (with $w = 3/4$) and M-reduced bases, respectively. Then from (11), (17), (14) and (8), one can immediately obtain

$$\delta_{H,n} \leq \gamma_n^{n/2} \left(\prod_{i=1}^{n} \frac{i+3}{4} \right)^{\frac{1}{2}} = 2^{O(n \log n)}; \tag{33}$$

$$\delta_{L,n} \leq 2^{\frac{n(n-1)}{4}}; \tag{34}$$

$$\delta_{M,n} \leq \gamma_n^{n/2} \left(\frac{5}{4} \right)^{\frac{(n-3)(n-4)}{4}} = \left(\frac{5}{4} \right)^{\frac{n^2}{4} + O(n \log n)}. \tag{35}$$

Thus for lattices of high dimension, an HKZ-reduced basis is expected to be more orthogonal than an LLL-reduced basis or an M-reduced basis. Note that the values of γ_n are known for $1 \leq n \leq 8$ and $n = 24$ [2, Page 33]. Thus for lattices of these dimensions, tight upper bounds on the orthogonality defect can be calculated. From Table 1, one can see that for lattices of dimension $n \leq 8$, the upper bound of the orthogonality defect of M-reduced bases is slightly smaller than that of HKZ-reduced bases, and both M-reduced and HKZ-reduced bases vectors are expected to be more orthogonal than LLL-reduced bases vectors, especially for $n = 7$ and $n = 8$. However, for lattices of a little higher dimensions such as $n = 24$, the degree of orthogonality of HKZ-reduced bases is expected to be higher than M-reduced bases, and the gap between HKZ-reduced bases and LLL-reduced bases gets larger quickly as dimension increases.

It is well known that the performance of MIMO detectors is highly related to the structure of the given basis. Since HKZ and M-reduced bases are expected to be more orthogonal than the conventional LLL-reduced bases, the error probability of approximate decoding algorithms can be further improved when HKZ or M-reduced bases are employed. Of course, the data presented in Table 1 only represent theoretical upper bounds. The average orthogonality defect of these reduction notions in practice shall be shown in Section 8.

7.2 Proximity Factors and Error Probability

The commonly used SIC decoding and ZF decoding were firstly proposed by Babai [17] in 1986. It is also proved in [17] that SIC and ZF aided by LLL-reduction (for $\omega = 3/4$) can find the closest vector within a factor $2^{n/2}$ and $1 + 2n(9/2)^{n/2}$, respectively. However, such results only

represent upper bounds on normwised gaps between approximate decoding and ILD, but can not reveal the bit performance loss of decoding algorithms.

From computer simulation, LLL-reduction-aided decoding can always achieve the full receive diversity of a MIMO fading channel [19, 46, 48]. The achievability was proved theoretically in [54, 63]. However, knowing the diversity order is insufficient to assess the performance gap between approximate decoding and ILD. To characterize the performance gap in a more precise way, a novel proximity factor was defined in [20] and further discussed in [6, 21].

We first consider the decision regions of different decoding algorithms for a fixed but arbitrary lattice. Without loss of generality, let L be an n-dimensional lattice and let the transmitted lattice vector be $\mathbf{x} = \mathbf{0}$, then the decision region of ILD is a *Voronoi region* defined by

$$\mathcal{R}_{ILD} = \{\mathbf{y} : \|\mathbf{y} - \mathbf{v}\|_2 \geq \|\mathbf{y}\|_2, \quad \forall \mathbf{v} \in L\}. \tag{36}$$

An error occurs when the noise falls outside of $\mathcal{R}_{ILD}$. It is known that $\mathcal{R}_{ILD}$ is an n-dimensional convex polytope and is symmetrical with respect to the origin. Each *facet* of $\mathcal{R}_{ILD}$ is an $(n-1)$-dimensional face of the polytope. Let $d_{i,ILD}$ be the Euclidean distance from $\mathbf{0}$ to the i-th facet of $\mathcal{R}_{ILD}$ and $\mathbf{v}_i$ be the corresponding Voronoi neighbor. Then $d_{i,ILD} = \|\mathbf{v}_i\|_2/2 \geq \lambda_1(L)/2$, and the minimum decoding distance $d_{ILD} \triangleq \min_i\{d_{i,ILD}\} = \lambda_1(L)/2$.

The decision regions of both SIC and ZF are polyhedra with $2n$ facets and are symmetrical with respect to the origin. As shown in [20], the i-th distance of SIC is $d_{i,SIC} = (\|\mathbf{b}_i\|_2 \sin\phi_i)/2 = r_{i,i}/2$, for $i = 1, \cdots, n$, where $r_{i,i}$ is the i-th diagonal element of the R-factor of the lattice generator matrix $\mathbf{B}$, and ϕ_i denotes the acute angle between $\mathbf{b}_i$ and the linear space spanned by the previous $i-1$ basis vectors. The i-th distance of ZF is $d_{i,ZF} = (\|\mathbf{b}_i\|_2 \sin\theta_i)/2$, where θ_i denotes the acute angle between $\mathbf{b}_i$ and the rest $n-1$ basis vectors $\mathbf{b}_1, \cdots, \mathbf{b}_{i-1}, \mathbf{b}_{i+1}, \cdots, \mathbf{b}_n$. It is easy to see $\theta_i \leq \phi_i$ and hence $d_{i,ZF} \leq d_{i,SIC}$.

Secondly, to measure the performance gap between approximate decoding and ILD, the proximity factors [20] are defined as:

$$\rho_{i,SIC} \triangleq \sup_{\mathbf{B}\in\mathcal{B}_{\text{Red}}} \frac{d_{ILD}^2}{d_{i,SIC}^2} = \sup_{\mathbf{B}\in\mathcal{B}_{\text{Red}}} \frac{\lambda_1^2(L)}{r_{i,i}^2}, \tag{37}$$

$$\rho_{i,ZF} \triangleq \sup_{\mathbf{B}\in\mathcal{B}_{\text{Red}}} \frac{d_{ILD}^2}{d_{i,ZF}^2} = \sup_{\mathbf{B}\in\mathcal{B}_{\text{Red}}} \frac{\lambda_1^2(L)}{\|\mathbf{b}_i\|_2^2 \sin\theta_i^2}, \tag{38}$$

where the supremum is taken over the set $\mathcal{B}_{\text{Red}}$ of bases satisfying a certain reduction notion for any n-dim lattice L. We further define $\rho_{SIC} \triangleq \max_i\{\rho_{i,SIC}\}$ and $\rho_{ZF} \triangleq \max_i\{\rho_{i,ZF}\}$. Using a union-bound argument [20, 21], the average error probability of ZF decoding can be bounded as

$$P_{e,ZF}(SNR) \leq \sum_{i=1}^{n} P_{e,ILD}\left(\frac{SNR}{\rho_{i,ZF}}\right) \leq nP_{e,ILD}\left(\frac{SNR}{\rho_{ZF}}\right) \tag{39}$$

for arbitrary SNR. A similar bound exists for SIC decoding. From (39), the error rate of approximate decoding with finite proximity factors can approximate that of ILD within a factor n. Thus, lattice-reduction-aided decoding (LRAD) has the same diversity order with ILD, and existing results on the diversity order of ILD can be extended to LRAD [54].

7.3 Proximity Factors of SIC Decoding

The upper bounds of ρ_{SIC} for LLL and HKZ-reduction were given in [20,21]. In this subsection, we shall improve existing result on ρ_{SIC} for LLL-reduction, and derive an upper bound of ρ_{SIC} for M-reduction.

7.3.1 LLL-reduction

Let $\mathbf{B} \in \mathbb{R}^{m\times n}$, $m \geq n$, be an LLL-reduced matrix and let $\mathbf{R}$ be the R-factor of $\mathbf{B}$. From (15), we have

$$r_{i,i}^2 \geq \omega r_{i-1,i-1}^2 - r_{i-1,i}^2 \geq (\omega - 1/4) r_{i-1,i-1}^2, \tag{40}$$

for $1 < i \leq n$. By induction, we have

$$r_{j,j}^2 \leq \beta^{i-j} r_{i,i}^2, \quad \text{for} \quad 1 \leq j < i \leq n, \tag{41}$$

where $\beta = 1/(\omega - 1/4) \geq 4/3$. Based on (41), an upper bound of $\rho_{i,SIC}$ was presented in [20]:

$$\rho_{i,SIC} = \sup \frac{\lambda_1^2(L)}{r_{i,i}^2} \leq \sup \frac{r_{1,1}^2}{r_{i,i}^2} \leq \beta^{i-1}. \tag{42}$$

In [21], the upper bound (42) was improved by a constant factor as follows:

$$\rho_{i,SIC} \leq 1 + \frac{\beta}{4(\beta-1)}(\beta^{i-1} - 1). \tag{43}$$

Now we present an improvement of the above bound. From the definition of Hermite's constant, we have

$$\lambda_1^2(L) \leq \gamma_i \cdot (r_{1,1} r_{2,2} \cdots r_{i,i})^{2/i}, \quad \text{for} \quad 1 < i \leq n. \tag{44}$$

Substituting (8) and (41) into (44), we obtain

$$\rho_{i,SIC} = \sup \frac{\lambda_1^2(L)}{r_{i,i}^2} \leq \gamma_i \cdot \beta^{\frac{i-1}{2}} \leq \left(1 + \frac{i}{4}\right) \beta^{\frac{i-1}{2}}. \tag{45}$$

It follows from (45) that

$$\rho_{SIC} = \rho_{n,SIC} \leq \gamma_n \cdot \beta^{\frac{n-1}{2}} \leq \left(1 + \frac{n}{4}\right) \beta^{\frac{n-1}{2}}. \tag{46}$$

Although the new upper bound (45) is still exponential with respect to the dimension n, it significantly improves the currently best known estimation (43), and the gap between (45) and (43) becomes larger quickly as i increases.

7.3.2 HKZ-reduction

The proximity factor for HKZ-reduction was discussed in [21]. Let $\mathbf{B} \in \mathbb{R}^{m\times n}$ be an HKZ-reduced matrix and let $\mathbf{R}$ be the R-factor of $\mathbf{B}$. It follows from [21,37] that

$$r_{j,j}^2 \leq \xi_{j-i+1} \cdot r_{i,i}^2, \quad \text{for} \quad 1 \leq j < i \leq n, \tag{47}$$

where ξ_k, $1 < k \leq n$, is the *KZ constant* defined in [37] as

$$\xi_k \triangleq \sup \frac{r_{1,1}^2}{r_{k,k}^2} \leq \gamma_k \prod_{t=2}^{k} \gamma_t^{1/(t-1)} \leq k^{1+\ln k}. \tag{48}$$

It is easy to see $\rho_{i,SIC} = \xi_i$, for $1 < i \leq n$, and thus

$$\rho_{SIC} = \rho_{n,SIC} \leq n^{1+\ln n}. \tag{49}$$

Comparing (46) and (49), the proximity factor of HKZ-reduction is better than that of LLL-reduction, since (49) grows sub-exponentially with the dimension n.

7.3.3 M-reduction

Let $\mathbf{B} \in \mathbb{R}^{m\times n}$ be an M-reduced matrix and let $\mathbf{R}$ be the R-factor of $\mathbf{B}$. From the definition of M-reduction, the submatrix consists of the first i $(1 < i \leq n)$ columns of $\mathbf{B}$ is also M-reduced. It follows from (13) and (14) that

$$\rho_{i,SIC} = \sup \frac{\lambda_1^2(L)}{r_{i,i}^2} \leq \frac{\|\mathbf{b}_i\|_2^2}{r_{i,i}^2} \leq \gamma_i^i, \quad \text{for} \quad i \leq 4, \tag{50}$$

$$\rho_{i,SIC} \leq \frac{\|\mathbf{b}_i\|_2^2}{r_{i,i}^2} \leq \gamma_i^i \cdot \left(\frac{5}{4}\right)^{\frac{(i-3)(i-4)}{2}}, \quad \text{for} \quad i > 4. \tag{51}$$

Hence, if $n \leq 4$, we have

$$\rho_{SIC} \leq \gamma_n^n, \tag{52}$$

else

$$\rho_{SIC} = \rho_{n,SIC} \leq \gamma_n^n \cdot (5/4)^{\frac{(n-3)(n-4)}{2}}. \tag{53}$$

In particular, when $n = 2$, we have $\rho_{SIC} = \gamma_2^2 = 4/3$, which agrees with Gaussian reduction. For large value of n, the proximity factor for M-reduction is worse than that for both LLL and HKZ-reduction, since (53) grows super-exponentially with the dimension n. Of course, this upper bound may not be tight and is only used to prove the diversity order of M-reduction-aided SIC decoding.

7.4 Proximity Factors of ZF Decoding

The upper bounds of ρ_{ZF} for LLL and HKZ-reduction were given in [20, 21]. In this subsection, we shall improve existing result on ρ_{ZF} for LLL-reduction, and derive an upper bound of ρ_{ZF} for M-reduction.

The derivation for ρ_{ZF} needs a lower bound of $\sin\theta_i^2$. Let $\mathbf{R}$ be the R-factor of $\mathbf{B}$, and set $\mathbf{A}_i = \mathbf{R}(i:n, i:n)^{\mathrm{T}}\mathbf{R}(i:n, i:n)$, $1 \leq i \leq n$. Then it is proved in [21] that

$$\sin^2\theta_i = \frac{1}{\|\mathbf{b}_i\|_2^2 \cdot (\mathbf{A}_i^{-1})_{1,1}} \tag{54}$$

Substituting (54) into (38), we obtain

$$\rho_{i,ZF} = \sup_{\mathbf{B}\in\mathcal{B}_{\mathrm{Red}}} \lambda_1^2(L) \cdot (\mathbf{A}_i^{-1})_{1,1}. \tag{55}$$

So an upper bound of $\rho_{i,ZF}$ can be immediately determined if the upper bound of $(\mathbf{A}_i^{-1})_{1,1}$ is found.

7.4.1 LLL-reduction

Following Babai's method [17] for the estimation of the lower bound of $\sin\theta_i^2$, an upper bound of ρ_{ZF} for LLL-reduction was presented in [20] as

$$\rho_{ZF} = \rho_{1,ZF} \leq \left(\frac{9\beta}{4}\right)^{n-1}. \tag{56}$$

In [21], using an estimation for $(\mathbf{A}_i^{-1})_{1,1}$, the lower bound of $\sin\theta_i^2$ was further refined and thus an improved upper bound of ρ_{ZF} was given as

$$\rho_{ZF} \leq \frac{\beta}{9\beta-4}\left(\frac{9\beta}{4}\right)^{n-1} + \frac{8\beta-4}{9\beta-4}. \tag{57}$$

Now we present an improvement of the above bound. To this aim, we first recall the following result on the upper bound of $(\mathbf{A}_i^{-1})_{1,1}$.

Lemma 2 ([21]) *Let $\mathbf{R} = [r_{i,j}]$ be the R-factor of a size-reduced lattice generator matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$. Then*

$$(\mathbf{A}_i^{-1})_{1,1} \leq r_{i,i}^{-2} + \frac{1}{9}\sum_{j=1}^{n-i}\left(\frac{9}{4}\right)^j r_{i+j,i+j}^{-2}. \tag{58}$$

From (55) and (58),

$$\rho_{i,ZF} \leq \frac{\lambda_1^2(L)}{r_{i,i}^2} + \frac{1}{9}\sum_{j=1}^{n-i}\left(\frac{9}{4}\right)^j \frac{\lambda_1^2(L)}{r_{i+j,i+j}^2}. \tag{59}$$

Substituting (45) into (59), we obtain

$$\rho_{i,ZF} \leq \gamma_i\beta^{\frac{i-1}{2}} + \frac{1}{9}\sum_{j=1}^{n-i}\left(\frac{9}{4}\right)^j \gamma_{i+j}\beta^{\frac{i+j-1}{2}}. \tag{60}$$

Thus,

$$\rho_{ZF} = \rho_{1,ZF} \leq 1 + \frac{1}{9}\sum_{j=1}^{n-1}\left(\frac{9}{4}\right)^j \gamma_{j+1}\beta^{\frac{j}{2}}. \tag{61}$$

It is easy to verify that the new bound (61) is better than the previous bound (57).

7.4.2 HKZ-reduction

In [21], using (47) and (58), an upper bound of ρ_{ZF} for HKZ-reduction is given:

$$\rho_{ZF} \leq 1 + \frac{1}{9}\sum_{j=1}^{n-1}\left(\frac{9}{4}\right)^j \xi_{j+1} \leq \left(\frac{9}{4}\right)^{n-1} n^{1+\ln n}. \tag{62}$$

Comparing (61) and (62), the proximity factor of HKZ-reduction is smaller than that of LLL-reduction. This is in accordance with the fact that HKZ-reduction is a stronger notion than LLL-reduction.

7.4.3 M-reduction

To derive the upper bound of $\rho_{i,ZF}$ for M-reduction, we give a technical lemma. The proof is given in Appendix 10.

Lemma 3 *Given an M-reduced basis* $\mathbf{B} \in \mathbb{R}^{m\times n}$ *and its R-factor* $\mathbf{R} \in \mathbb{R}^{n\times n}$*. If* $n \leq 4$*, then*

$$(\mathbf{A}_i^{-1})_{1,1} \leq r_{i,i}^{-2} \cdot \prod_{k=i+1}^{n} \gamma_k^k, \tag{63}$$

else

$$(\mathbf{A}_i^{-1})_{1,1} \leq r_{i,i}^{-2} \cdot \prod_{k=i+1}^{n} \gamma_k^k \cdot \prod_{k=\max\{5,i+1\}}^{n} \left(\frac{5}{4}\right)^{\frac{(k-3)(k-4)}{2}}. \tag{64}$$

It follows from (55), (63) and (13) that if $n \leq 4$,

$$\rho_{i,ZF} \leq \frac{\lambda_1^2(L)}{r_{i,i}^2} \prod_{k=i+1}^{n} \gamma_k^k \leq \prod_{k=i}^{n} \gamma_k^k. \tag{65}$$

Similarly, for the case $n > 4$, we can deduce

$$\rho_{i,ZF} \leq \prod_{k=i}^{n} \gamma_k^k \cdot \prod_{k=\max\{5,i\}}^{n} \left(\frac{5}{4}\right)^{\frac{(k-3)(k-4)}{2}}. \tag{66}$$

Thus, if $n \leq 4$, we have

$$\rho_{ZF} = \rho_{1,ZF} \leq \prod_{k=1}^{n} \gamma_k^k, \tag{67}$$

while for $n > 4$,

$$\rho_{ZF} = \rho_{1,ZF} \leq \prod_{k=1}^{n} \gamma_k^k \cdot \prod_{k=5}^{n} \left(\frac{5}{4}\right)^{\frac{(k-3)(k-4)}{2}} \tag{68}$$

In particular, when $n = 2$, we have $\rho_{ZF} = \gamma_2^2 = 4/3$, which agrees with Gaussian reduction. Comparing (52) and (53) with (67) and (68), the proximity factor of SIC decoding is much smaller than ZF decoding for M-reduction. Interestingly, when $i = n$, the ZF decoder has the proximity factor $\rho_{n,ZF} \leq \gamma_n^n$ $(n \leq 4)$ or $\rho_{n,ZF} \leq \gamma_n^n (5/4)^{\frac{(n-3)(n-4)}{2}}$ $(n > 4)$, which is equal to $\rho_{n,SIC}$. This is in accordance with the fact that for the first component $\mathbf{x}_n$ to detect, SIC reduces to ZF as it can not benefit from interference cancellation. As i goes from $n-1$ to 1, SIC decoding gets better, while ZF decoding gets worse. That is, $\rho_{i,ZF}$ gets larger while $\rho_{i,SIC}$ gets smaller as i decreases.

8 Simulation Results

In this section, we present our simulation results to support the theoretical analysis in Section 7. We compare the efficiency of the proposed new algorithms by means of computer simulation. The

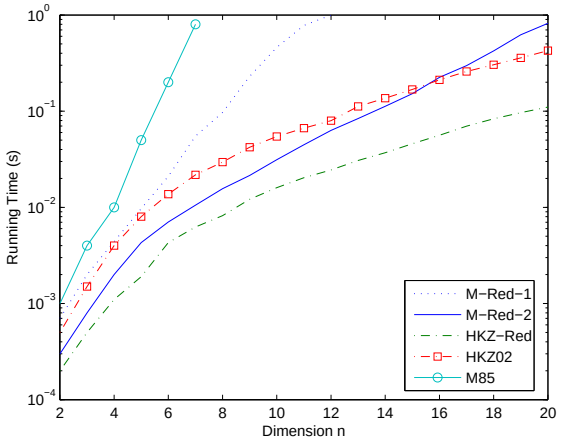

Figure 12: Comparison of the average running times among algorithms HKZ-Red, M-Red-1, M-Red-2, the HKZ-reduction algorithm in [5] (HKZ02), and the M-reduction algorithm in [32] (M85) for random generator matrices

proximity factors as well as the BER performance of approximate lattice decoding algorithms aided by LLL, HKZ and M-reduced bases are also compared. All experiments were performed on matrices with random entries, drawn from i.i.d. zero-mean, unit variance Gaussian distributions.

Firstly, we compare the running times of Algorithm HKZ-Red, Algorithm M-Red-1 and Algorithm M-Red-2 with the HKZ-reduction algorithm presented in [5] and the M-reduction algorithm presented in [32]. To assess the efficiency of these algorithms, the median of the average running times for 1000 random matrices are computed. Occasionally, a random matrix with very long running time is drawn. Using the median rather than the mean guarantees that these rare matrices do not dominate the average running times. Fig. 12 depicts our results, where each point is given in average time (in seconds) of dimension n, using a DELL computer with a 2.0-GHz Pentium Dual processor, with MATLAB running under Windows XP. Note that for each dimension, the running times for all the algorithms are averaged using the same matrices. Fig. 12 shows that Algorithm HKZ-Red is more efficient, about one magnitude order, than the HKZ-reduction algorithm presented in [5] (with the legend HKZ02). This illustrates that the new basis expansion strategy Procedure Transform is more efficient than the conventional strategy Procedure Select-Basis. Also, our second improved M-reduction Algorithm M-Red-2 is more efficient than our first M-reduction Algorithm M-Red-1, and the gap between them becomes larger quickly as the dimension increases. Apparently, both Algorithm M-Red-1 and Algorithm M-Red-2 are much more efficient than the algorithm presented in [32] (with the legend M85). Besides, as discussed in Section 5, the algorithm in [32] produces M-reduced bases only for matrices of dimensions up to seven, while the two new algorithms are valid for matrices of arbitrary dimension and are practical for dimensions much larger than seven.

Secondly, during the process of Algorithm HKZ-Red, Algorithm M-Red-1 and Algorithm M-Red-2, the computational cost in each iteration is dominated by Schnorr-Euchner enumeration, Procedure M-Decode-1 and Procedure M-Decode-2, respectively. Then, to further investigate the efficiency of the three reduction algorithms, we compare the average complexity

Table 2: The Average Cardinality of The Search Space and The Number of Gcd Operations Costed in Each Iteration of Algorithms HKZ-Red, M-Red-1 and M-Red-2, Over Random Matrices of Order 20

k	search space			# of gcd operations	
	Schnorr-Euchner	M-Decode-1	M-Decode-2	M-Decode-1	M-Decode-2
1	7.16×10^2	4.10×10^3	7.16×10^2	3.66×10^2	1.97×10^1
5	4.01×10^2	1.15×10^4	2.68×10^3	1.69×10^3	1.53×10^2
10	1.76×10^2	1.64×10^4	3.14×10^3	2.62×10^3	2.72×10^2
15	3.60×10^1	4.74×10^4	2.86×10^3	8.10×10^3	6.34×10^1
19	6.48×10^0	1.03×10^5	2.98×10^3	8.91×10^3	5.26×10^0

of the three procedures called in each iteration, by using the cardinality of the search space as a measurement. Moreover, note that in Procedure M-Decode-1 and Procedure M-Decode-2, the gcd computations (Euclidean algorithm) provide some additional complexity. We show our results in Table 2. Again, each entry in the table is the average of 1000 random matrices of order 20, and the index of iterations is denoted by k. Table 2 shows that as the iteration continues, more basis vectors are produced, the search space of Schnorr-Euchner enumeration (called in HKZ-Red) decreases, the search space of Procedure M-Decode-1 (called in M-Red-1) increases, while the search space of Procedure M-Decode-2 (called in M-Red-2) stays about the same. This can be explained as follows. In HKZ-reduction, after each iteration, the dimension of the sublattice to be searched is reduced by one, thus the search space of Schnorr-Euchner enumeration decreases rapidly as k increases. However, for M-reduction, the dimension of the sublattice to be searched stays the same as the iteration continues. Note that in Procedure M-Decode-2, the constraint $\gcd(z_k, ..., z_n) = 1$ is imposed as soon as $z_k, ..., z_n$ are available. Thus the complexity of Procedure M-Decode-2 do not vary much for different k. But for Procedure M-Decode-1, the gcd constraint can not be checked until the whole integer vector $\mathbf{z}$ is available. Therefore Procedure M-Decode-1 always costs more than Procedure M-Decode-2. Moreover, as the iteration continues, the search space of Procedure M-Decode-1 increases rapidly, since the constraint $\gcd(z_k, ..., z_n) = 1$ gets more severe as k increases. We can also obtain from Table 2 that for each iteration, the average numbers of gcd operations performed in both of the two procedures are roughly 1/10 of the cardinality of the search space. Thus, checking the gcd constraint does not costs much when compared with the total complexity.

Thirdly, we compare the average orthogonality defect of LLL, HKZ and M-reduced bases produced by our new algorithms. Fig. 13 shows both the theoretical upper bounds (33), (34), and (35) and our experimental results. As shown in the figure, for low dimensions, the upper bound in M-reduction is the best and LLL-reduction is the worst, as discussed in Section 7.1. For higher dimensions, the theoretical upper bound (33) for HKZ-reduction is better than M-reduction. However, our experiments showed that the orthogonality defect of M-reduction is always the best. This suggests that the upper bound (35) for M-reduction may be too conservative and there may be a room for improvement.

Forthly, we compare the proximity factors of ZF decoding and SIC decoding with LLL, HKZ, and M-reduced bases. Fig. 14 shows the theoretical upper bounds. As described in Section 7, for each reduction, the proximity factor of SIC decoding is much smaller than that of ZF decoding. For both SIC decoding and ZF decoding, the proximity factor of LLL-reduction is larger than

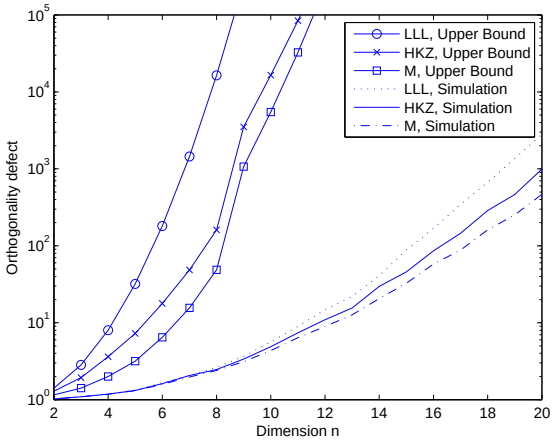

Figure 13: Comparison of the theoretical upper bounds and numerical results of the orthogonality defect for LLL ($\omega = 3/4$), HKZ, and M-reduction for random matrices.

that of HKZ-reduction, and M-reduction is the largest. Note that the upper bounds on the proximity factors may not be tight. Especially, the bound (53) for M-reduction is unlikely to be tight, because we have applied the trivial bound $\lambda_1^2(L) \leq \|\mathbf{b}_i\|_2^2$ in (50). Since we know from (12) that $(5/4)^{-(n-4)} \leq \lambda_1^2(L)/\|\mathbf{b}_i\|_2^2 \leq 1$, this is likely to loosen the bound by a factor of $(5/4)^{(n-4)}$ at the worst. Besides, for all upper bounds, we have applied the bounds on Hermite's constants when the exact values are unknown.

To obtain a practical view of the proximity factors, we simulated them by means of numerical experimentation. For each value of n, we generate 1000 random matrices and apply LLL algorithm and our new algorithms to obtain LLL, HKZ, and M-reduced bases. Then the proximity factors can be taken as the maximum over these reduced bases. Although the maximum may not reach the bounds in the worst case, they should be reasonable approximations of the theoretical proximity factors. Fig. 15 shows the numerical results. We can learn from this figure that for ZF decoding, the proximity factor of M-reduction is the smallest, while for SIC decoding, the proximity factor of HKZ-reduction is the smallest. For both ZF decoding and SIC decoding, the proximity factor of LLL-reduction is the largest.

Finally, we investigate the BER performance of both ZF decoding and SIC decoding with different reduction notions. In Fig. 16, we simulated the BER of different decoding algorithms for an 8×8 MIMO system with a 64-QAM constellation. The SNR at each receive antenna is defined as $\text{SNR} = n_T E_{x \in \text{64-QAM}}[x^2]/\sigma^2$. Note that ILD is quite different from other algorithms shown in Fig. 16, since ILD ignores the signal boundary, while SIC decoding, ZF decoding and ML decoding map each entry of the decoded lattice point onto the 64-QAM alphabet by a minimum distance quantization. This figure shows that for ZF decoding, M-reduction has the lowest BER, while for SIC decoding, HKZ-reduction has the lowest BER, which is consistent with the simulation results on the proximity factors depicted in Fig. 15.

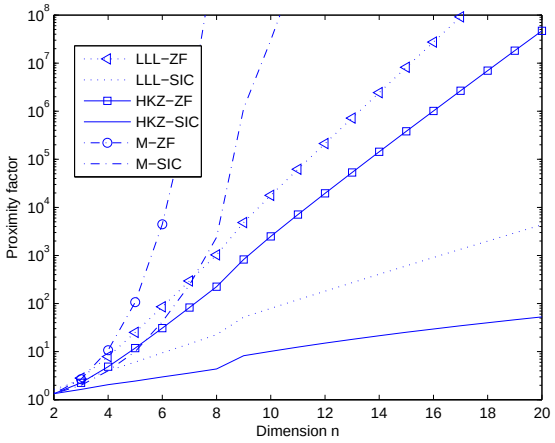

Figure 14: Comparison of the theoretical upper bounds on the proximity factors for ZF decoding and SIC decoding with LLL ($\omega = 3/4$), HKZ, and M-reduced bases.

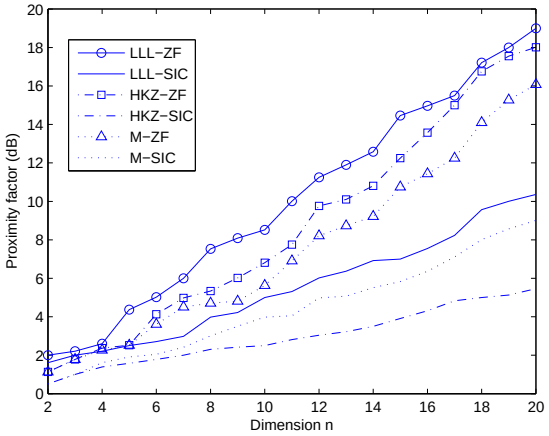

Figure 15: Comparison of the simulated results on the proximity factors for ZF decoding and SIC decoding with LLL ($\omega = 3/4$), HKZ, and M-reduced bases.

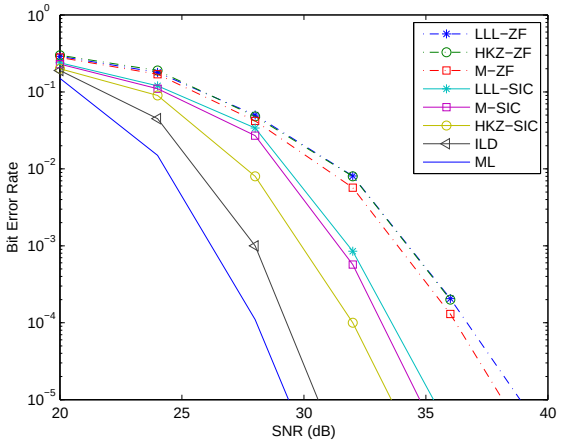

Figure 16: The BER performance of ILD, ML decoding, and ZF and SIC decoding with LLL, HKZ, and M-reduced bases in an uncoded 8×8 complex-valued MIMO system with a 64-QAM constellation.

9 Concluding Remarks

In this paper, we first present a new HKZ-reduction algorithm using the unimodular transformation. Our experiments show that our new algorithm is a significant, about one magnitude order, improvement of the existing HKZ algorithm in [5]. Also, by solving the constraint SVP (28) and using the unimodular matrices (31), we propose two M-reduction algorithms. The second M-reduction algorithm improves the first one by an early detection of the gcd constraint, and both of them are much faster and more general than the existing algorithm in [32]. To compare the quality of different reduced bases produced by LLL algorithm and our new algorithms, we discussed the orthogonality defect of LLL, HKZ, and M-reduced bases. Fig. 13 shows that in practice M-reduced bases always have the smallest orthogonality defect, which suggests a potential of improving the theoretical upper bound (35). Another topic discussed in this paper is the application of different reduced bases in approximate decoding algorithms. We employ the concept of proximity factors defined in [20] to assess the performance of decoding algorithms. For the proximity factors of SIC and ZF decoding aided by LLL-reduction, we derive new improved bounds (46) and (61). For decoding algorithms aided by M-reduction, the upper bounds on the proximity factors are also derived. Fig. 15 and Fig. 16 show that for SIC decoding, HKZ-reduction has the best performance, while for ZF decoding, M-reduction has the best performance.

10 Appendix

[Proof of Lemma 3] From the definition of $\mathbf{A}_i$, $(\mathbf{A}_i^{-1})_{1,1}$ is the squared Euclidean length of the first row of $\mathbf{R}(i:n, i:n)^{-1}$. For $i \leq k < n$, we denote $\mathbf{S}_k = \mathbf{R}(i:k, i:k)^{-1}$. Then it is easy

to verify that $\mathbf{S}_i = 1/r_{i,i}$ and

$$\mathbf{S}_k = \begin{bmatrix} \mathbf{S}_{k-1} & \frac{1}{r_{k,k}} \cdot \mathbf{S}_{k-1} \cdot \mathbf{R}(i:k-1,k) \\ 0 & \frac{1}{r_{k,k}} \end{bmatrix} \tag{69}$$

for $i < k \leq n$.

From (13) and (14), one can derive that if $n \leq 4$,

$$\|\mathbf{R}(i:k-1,k)\|_2^2 \leq \|\mathbf{R}(1:k-1,k)\|_2^2 \leq (\gamma_k^k - 1) r_{k,k}^2; \tag{70}$$

else,

$$\|\mathbf{R}(i:k-1,k)\|_2^2 \leq (\gamma_k^k \cdot (5/4)^{\frac{(n-3)(n-4)}{2}} - 1) r_{k,k}^2. \tag{71}$$

It follows from (69), (70) that if $n \leq 4$

$$\begin{aligned} \|\mathbf{S}_k(1,:)\|_2^2 &\leq \|\mathbf{S}_{k-1}(1,:)\|_2^2 + \frac{\|\mathbf{S}_{k-1}(1,:)\|_2^2 \cdot \|\mathbf{R}(i:k-1,k)\|_2^2}{r_{k,k}^2} \\ &\leq \gamma_k^k \|\mathbf{S}_{k-1}(1,:)\|_2^2 \end{aligned} \tag{72}$$

From (72), we can derive by induction that

$$(\mathbf{A}_i^{-1})_{1,1} = \|\mathbf{S}_n(1,:)\|_2^2 \leq \mathbf{S}_i^2 \cdot \prod_{k=i+1}^{n} \gamma_k^k \leq r_{i,i}^{-2} \cdot \prod_{k=i+1}^{n} \gamma_k^k \tag{73}$$

Based on (69) and (71), the inequality (64) for the case $n > 4$ can be easily obtained by using an induction approach similar with (72). Thus the proof is complete.

References

[1] J. W. S. Cassels, *An Introduction to The Geometry of Numbers*, 2nd ed. Berlin, Germany: Springer-Verlag, 1997.

[2] P. Q. Nguyen and B. Vallée, Eds., *The LLL Algorithm: Survey and Applications.* Berlin, Germany: Springer-Verlag, 2009.

[3] D. E. Knuth, *The Art of Computer Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1981.

[4] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization.* Berlin, Germany: Springer-Verlag, 1993.

[5] 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, Aug. 2002.

[6] Y. H. Gan, C. Ling, and H. M. 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.

[7] D. Wübben, D. Seethaler, J. Jaldén, and G. Marz, "Lattice reduction: a survey with applications in wirless communications," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 70–91, May 2011.

[8] A. Joux and J. Stern, "Lattice reduction: A toolbox for the cryptanalyst," *J. Cryptology*, vol. 11, no. 3, pp. 161–185, 1998.

[9] D. Micciancio and S. Goldwasser, *Complexity of Lattice Problems: A Cryptographic Perspective*. Boston, MA: Kluwer Academic Publishers, 2002.

[10] K. J. Kim and R. A. Iltis, "Joint constrained data detection and channel estimation algorithms for QS-CDMA signals," in *Proc. Asilomar Conf. Signals, Systems and Computers*, vol. 1, Pacific Grove, CA, 2001, pp. 397–398.

[11] X. Ma, W. Zhang, and A. Swami, "Lattice-reduction aided equalization for OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1608–1613, Apr. 2009.

[12] M. O. Damen, H. E. Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.

[13] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I: Expected complexity," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2806–2818, Jul. 2005.

[14] J. Jaldén and B. Ottersen, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1474–1484, Mar. 2005.

[15] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Programming*, vol. 66, pp. 181–199, 1994.

[16] M. Phost, "On the computation of lattice vectors of minimal length, successive minima and reduced bases with applications," *ACM SIGSAM Bull*, vol. 15, pp. 37–44, Feb. 1981.

[17] L. Babai, "On lovász's lattice reduction and the nearest lattice point problem," *Combinatorica*, vol. 6, pp. 1–13, 1986.

[18] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela, "V-blast: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. IEEE Int. Symp. Signals. Syst. Electron. Conf. (ISSSE'98)*, Pisa, Italy, Sep. 1998, pp. 295–300.

[19] W. H. Mow, "Universal lattice decoding: Principle and recent advances," *Wireless Commun. Mobile Comput., Special Issue on Coding and Its Appl. Wireless CDMA Syst.*, vol. 3, pp. 553–569, Aug. 2003.

[20] C. Ling, "Towards characterizing the performance of approximate lattice decoding," in *Proc. Int. Symp. Turbo Codes/Int. Conf. Source Channel Coding '06*, Munich, Germany, Apr. 2006.

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

[22] C. Hermite, "Extraits de lettres de M. Hermite à M. Jacobi sur différents objets de la théorie des nombres," *J. Reine Angew. Math.*, vol. 40, pp. 279–290, 1850.

[23] A. Korkine and G. Zolotareff, "Sur les formes quadratiques," *Math. Ann.*, vol. 6, pp. 366–389, 1873.

[24] R. Kannan, "Improved algorithms for integer programming and related lattice problems," in *Proc. ACM Symp. Theo. Comp.*, Boston, MA, Apr. 1983, pp. 193–206.

[25] B. Helfrich, "Algorithms to construct Minkowski reduced and Hermite reduced lattice bases," *Theor. Comput. Sci.*, vol. 41, no. 2-3, pp. 125–139, 1985.

[26] R. Kannan, "Minkowski's convex body theorem and integer programming," *Math. Oper. Res.*, vol. 12, pp. 415–440, Aug. 1987.

[27] A. H. Banihashemi and A. K. Khandani, "On the complexity of decoding lattices using the korking-zolotarev reduced basis," *IEEE Trans. Inf. Theory*, vol. 44, no. 1, pp. 162–171, Jan. 1998.

[28] H. Minkowski, "Über die positiven quadratischen formen und über kettenbruchähnliche algorithmen," *J. Reine und Angewandte Math.*, vol. 107, pp. 278–297, 1891.

[29] J. L. Lagrange, "Recherches d'arithmétique," *Nouv. Mém. Acad. Berlin*, 1773.

[30] I. Semaev, "A 3-dimensional lattice reduction algorithm," in *Proc. Cryptography and Lattices Conf. (CALC'01)*, Rhode Island, USA, Mar. 2001, pp. 181–193.

[31] P. Q. Nguyen and D. Stehlé, "Low-dimensional lattice basis reduction revisited," *ACM Trans. Algor.*, vol. 5, no. 4, Oct. 2009.

[32] L. Afflerbach and H. Grothe, "Calculation of Minkowski-reduced lattice bases," *Computing*, vol. 35, no. 3-4, pp. 269–276, 1985.

[33] D. Micciancio, "The hardness of the closest vector problem with preprocessing," *IEEE Trans. Inf. Theory*, vol. 47, no. 3, pp. 1212–1215, Mar. 2001.

[34] S. Arora, L. Babai, and J. Stern, "The hardness of approximate optima in lattices, codes, and systems of linear equations," *J. Comput. Syst. Sci*, vol. 54, no. 2, pp. 317–331, Apr. 1997.

[35] A. K. Lenstra, H. Lenstra, Jr, and L. Lovász, "Factorizing polynomials with rational coefficients," *Math. Ann.*, vol. 261, pp. 515–534, Dec. 1982.

[36] D. Boneh, "Twenty years of attacks on the RSA cryptosystem," *Notices Amer. Math. Soc.*, vol. 46, pp. 203–213, 1999.

[37] C. P. Schnorr, "A hierarchy of polynomial lattice basis reduction algorithms," *Theor. Comput. Sci.*, vol. 53, no. 2-3, pp. 201–224, 1987.

[38] F. T. Luk and D. M. Tracy, "An improved LLL algorithm," *Linear Algebra Appl.*, vol. 428, no. 2-3, pp. 441–452, Jan. 2008.

[39] F. T. Luk and S. Qiao, "A pivoted LLL algorithm," *Linear Algebra Appl.*, vol. 434, no. 11, pp. 2296–2307, Jun. 2011.

[40] P. Q. Nguyen and D. Stehlé, "An LLL algorithm with quadratic complexity," *SIAM J. Comput.*, vol. 39, no. 3, pp. 874–903, 2009.

[41] Y. H. Gan and W. H. Mow, "Complex lattice reduction algorithms for low-complexity MIMO detection," in *Proc. IEEE Global Telecommun. Conf. (IEEE GLOBECOM '05)*, 2005, pp. 5–5.

[42] D. Wübben, R. Böhnke, J. Rinas, V. Kühn, and K. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *Electron. Lett.*, vol. 37, no. 22, pp. 1348–1350, Oct. 2001.

[43] Y. H. Gan and W. H. Mow, "Novel joint sorting and reduction technique for delay-constrained LLL-aided MIMO detection," *IEEE Signal Process. Lett.*, vol. 15, pp. 194–197, 2008.

[44] C. Ling and N. Howgrave-Graham, "Effective LLL reduction for lattice decoding," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Nice, France, Jun. 2007.

[45] F. T. Luk, S. Qiao, and W. Zhang, "Lattice basis reduction algorithm," Institute for Computational Mathematics, Hong Kong Baptist University, Tech. Rep. 10-04, Apr. 2010.

[46] D. Wübben, R. Böhnke, V. Kühn, and K. D. Kammeyer, "Near-maximum-likelihood detection of MIMO systems using MMSE-based lattice reduction," in *Proc. Int. Commun. Conf. (ICC' 04)*, Jun. 2004, pp. 798–802.

[47] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: The Johns Hopkins University Press, 1996.

[48] D. Wübben, R. Böhnke, V. Kühn, and K. D. Kammeyer, "MMSE-based lattice-reduction for near-ML detection of MIMO systems," in *Proc. Int. ITG Workshop on Smart Antennas*, Munich, Germany, Mar. 2004, pp. 106–113.

[49] I. Morel, D. Stehlé, and G. Villard, "H-LLL: Using Housholder inside LLL," in *Proc. Int. Symp. on Symb. and Alg. Comp. (ISSAC' 09)*, Seoul, Korea, Jul. 2009, pp. 271–278.

[50] X. W. Chang and G. H. Golub, "Solving ellipsoid-constrained integer least squares problems," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 1071–1089, 2009.

[51] J. C. Lagrias, H. W. Lenstra, and C. P. Schnorr, "Korkine-Zolotarev bases and successive minima of a lattice and its reciprocal lattice," *Combinatorica*, vol. 10, no. 4, pp. 333–348, 1990.

[52] B. L. van der Waerden and H. Gross, *Studien zur Theorie der Quadratischen Formen*. Birkhäuser Basel, 1968.

[53] J. Jaldén, D. Seethaler, and G. Matz, "Worst- and average-case complexity of LLL lattice reduction in MIMO wireless systems," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, NV, Apr. 2008, pp. 2685–2688.

[54] 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, Dec. 2007.

[55] A. Vardy and Y. Be'ery, "Maximum-likelihood decoding of the Leech lattice," *IEEE Trans. Inf. Theory*, vol. 39, no. 4, pp. 1435–1444, Jul. 1993.

[56] A. H. Banihashemi and I. F. Blake, "Trellis complexity and minimal trellis diagrams of lattices," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1829–1847, Sep. 1998.

[57] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, no. 170, pp. 463–471, Apr. 1985.

[58] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.

[59] M. Ajtai, R. Kumar, and D. Sivakumar, "A sieve algorithm for the shortest lattice vector problem," in *Proc. ACM STOC '01*, Crete, Greece, Jul. 2001, pp. 601–610.

[60] P. Q. Nguyen and T. Vidick, "Sieve algorithms for the shortest vector problem are practical," *J. Math. Crypt.*, vol. 2, no. 2, pp. 181–207, 2008.

[61] D. Micciancio and P. Voulgaris, "Faster exponential time algorithms for the shortest vector problem," in *Proc. ACM/SIAM SODA '10*, Austin, Texas, Jan. 2010, pp. 1468–1480.

[62] P. Q. Nguyen and D. Stehlé, "LLL on the average," in *Proc. International Algorithmic Number Theory Symposium (ANTS-VII)*, vol. 4076, Berlin, Germany, Jul. 2006, pp. 238–256.

[63] J. Jaldén 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, Oct. 2010.