A Lattice Basis Reduction Algorithm [*]

Franklin T. Luk
Department of Mathematics, Hong Kong Baptist University
Kowloon Tong, Hong Kong
Sanzheng Qiao
Department of Computing and Software, McMaster University
Hamilton, Ontario L8S4K1 Canada
Wen Zhang
Institute of Mathematics, School of Mathematical Science
Fudan University, Shanghai, 200433 P.R. China

Abstract

In this paper, we give a definition of an optimally reduced basis for a lattice in the sense that an optimally reduced basis is a shortest basis for the lattice. Then we present an algorithm for computing an approximation of an optimally reduced basis for a lattice using a novel unimodular transformation. To compare lattice bases, we propose a quantitative measure of the degree of the linear independence of lattice basis vectors.

Keywords Lattice, lattice basis reduction, unimodular transformation, linear independence.

1 Introduction

A lattice is a set of discrete points representing integer linear combinations of linearly independent vectors. The set of linearly independent vectors generating a lattice is called a basis for the lattice. A set of lattice points does not uniquely determine a basis. This leads to the problem of finding a “nearly” orthogonal basis. Intuitively, shorter basis vectors are “more” orthogonal. A basis reduction algorithm finds a reduced basis, that is, a basis whose vectors are reduced in length. The lattice reduction problem arises from fields such as integer programming [2], cryptology [6], number theory [4], and information theory [1]. In this paper, after a short introduction to lattices and bases in Section 2, various definitions of reduced basis are described in Section 3. They include the definitions of Minkowski-reduced basis, Hermite size-reduced basis, HKZ-reduced basis, and LLL-reduced basis. Then we introduce the definition of an optimally reduced basis, in the sense that an optimally reduced basis cannot be further reduced. Examples are given to illustrate the relations among the different kinds of reduced bases. Following a brief

[*] Sanzheng Qiao is partially supported by Key Laboratory of Mathematics for Nonlinear Sciences (Fudan University) and NSERC of Canada. Wen Zhang is partially supported by the National Natural Science Foundation of China under grant 10871051, Doctoral Program of the Ministryof Education under grant 20090071110003 and KLMM0901.

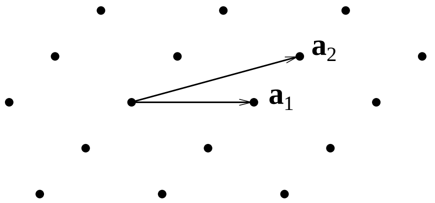

Figure 1: The column vectors $\mathbf{a}_1$ and $\mathbf{a}_2$ of A in (1) and the lattice points generated by them.

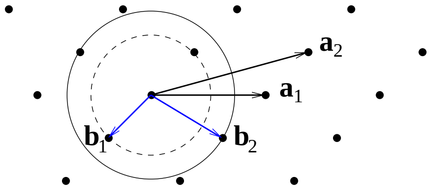

Figure 2: The columns of A and B and the lattice.

description of the LLL reduction algorithm in Section 4, we present an algorithm for computing an approximation of an optimally reduced basis in Section 5. The key to our algorithm is a novel lattice basis transformation. It allows us to transform a lattice basis into another basis whose first vector is a shortest nonzero vector in the lattice. In Section 6, we prove that our algorithm terminates in finite number of iterations. To compare reduction algorithms, in Section 7, we propose a quantitative measurement, called linear independence number, for lattice bases. We show that this number is a better measurement for lattice bases than the currently widely used matrix condition number. Finally, our experimental results shown in Section 8 demonstrate that our algorithm produces shorter bases than the HKZ and LLL algorithms.

2 Lattices and Bases

Given a real matrix $A \in R^{m \times n}$, $m \geq n$, of full column rank, the set

$$L = \{A\mathbf{z}, \quad \text{for all integer vectors } \mathbf{z} \in Z^n\},$$

containing discrete grid points, is called a lattice. The linearly independent columns of A form a *basis* for L and n is called the *rank* of L. For example, Figure 1 depicts the lattice points generated by the matrix

$$A = [\mathbf{a}_1 \;\; \mathbf{a}_2] = \begin{bmatrix} 2.0 & 2.7 \\ 0 & 0.7 \end{bmatrix}. \tag{1}$$

A set of lattice points does not uniquely determine a basis. For example, the matrix

$$B = [\mathbf{b}_1 \;\; \mathbf{b}_2] = \begin{bmatrix} -0.7 & 1.3 \\ -0.7 & -0.7 \end{bmatrix} \tag{2}$$

generates the same lattice in Figure 1. Figure 2 shows the lattice and the columns of A and B.

In general, if Z is an integer matrix whose inverse is also an integer matrix, then both A and AZ generate the same lattice. For example, the matrices in (1) and (2) are related by

$$B = AZ, \quad \text{where } Z = \left[\begin{array}{rr} 1 & 2 \\ -1 & -1 \end{array} \right].$$

From the following definition, a nonsingular integer matrix has an integer inverse if and only if it is a unimodular matrix.

Definition 1 (Unimodular) *A nonsingular integer matrix M is called unimodular if* $\det(M) = \pm 1$.

Obviously, from the definition, $|\det(A)| = |\det(AM)|$, for any unimodular M. Consequently, if A generates a lattice L, the quantity

$$d(L) = |\det(A)|$$

is independent of the choice of basis for L and thus defined as the *determinant* of a lattice L. We can view the determinant as the volume of the parallelepiped spanned by a basis for the lattice.

3 Reduced Bases

Since a lattice can have many bases, it is desirable to find a “good” basis. It is reasonable to say that a basis consisting of shorter vectors is “better”, since we expect shorter basis vectors are “more” orthogonal to each other. We say that a short basis is *reduced.* In terms of matrices, if A is a lattice generator matrix, we want to find a unimodular matrix Z so that the columns of AZ are short. We see from Figure 2 that a “better”, or “more” orthogonal, basis is shorter, or reduced in length.

3.1 Minkowski Minima

The concept of reduced bases in the sense of Minkowski minima [12] is probably best illustrated by an example. Using the matrix B in (2) and the lattice L for which the columns of B form a basis, we consider the Euclidean length $\|B\mathbf{z}\|_2$ of a lattice point $B\mathbf{z}$. First we find a nonzero lattice point of the shortest Euclidean length. This can be done because there are only finite number of nonzero lattice points $B\mathbf{z}$ satisfying

$$\|B\mathbf{z}\|_2 \leq \|B\mathbf{e}_1\|_2 = \|\mathbf{b}_1\|_2 \approx 1.0,$$

where $\mathbf{e}_1$ is the first unit vector $[1\ 0]^{\mathrm{T}}$. Actually, as shown in Figure 2, in this case, the lattice point $B\mathbf{e}_1 = \mathbf{b}_1$ is a shortest nonzero lattice vector. Next, we find a shortest lattice point which is linearly independent of $\mathbf{b}_1$. This also can be done, because $\mathbf{b}_2$ is linearly independent of $\mathbf{b}_1$ and there are only finite number of lattice points satisfying

$$\|B\mathbf{z}\|_2 \leq \|B\mathbf{e}_2\|_2 = \|\mathbf{b}_2\|_2 \approx 1.5,$$

where $\mathbf{e}_2$ is the unit vector $[0 \ \ 1]^{\mathrm{T}}$. Indeed, $B\mathbf{e}_2$ is one of the shortest lattice points linearly independent of $\mathbf{b}_1$. Thus the columns of B form a "short" basis for the lattice. The two successive minima $\|\mathbf{b}_1\|_2$ and $\|\mathbf{b}_2\|_2$ are called the Minkowski minima. The columns of B form a basis for the lattice and their lengths simultaneously attain the successive minima in the same order. We say that the columns of B form a Minkowski-reduced basis for the lattice. In comparison, A in (1) is not reduced since its columns are not two shortest basis vectors for the lattice as shown in Figure 2.

Figure 2 shows a geometric interpretation of the Minkowski minima. Starting at the origin, we expand a circle. The nonzero lattice point given by the first column $\mathbf{b}_1$ of B is one of the first lattice points hitting the growing circle, shown as the dash circle in the figure. As the circle continues to grow, the lattice point given by the second column $\mathbf{b}_2$ of B is one of the lattice points that are linearly independent of the first column and first hit the circle, shown as the solid circle in the figure. The figure also shows that A is not reduced since its columns do not form a short basis for the lattice.

In general, we have the following definition of Minkowski minima.

Definition 2 (Minkowski Minima) *Using the Euclidean norm as a distance function, we say that λ_k, $1 \leq k \leq n$, is the kth successive minimum with respect to a lattice L, if λ_k is the lower bound of the radius λ of the sphere*

$$\|B\mathbf{z}\|_2 \leq \lambda$$

that contains k linearly independent lattice points [3, Page 201].

In other words, λ_k is the lower bound for $\max(\|\mathbf{x}_1\|_2, \|\mathbf{x}_2\|_2, ..., \|\mathbf{x}_k\|_2)$ over all sets of linearly independent lattice points $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_k$. Note that originally, the Minkowski minima are defined in terms of $\|\mathbf{x}_i\|_2^2$ in the context of quadratic form. Here, for simplicity, we define the Minkowski minima in terms of $\|\mathbf{x}_i\|_2$. Of course, the disadvantage of using $\|\mathbf{x}_i\|_2$ is that when $\mathbf{x_i}$ is a rational vector, $\|\mathbf{x}_i\|_2$ can be irrational, whereas $\|\mathbf{x}_i\|_2^2$ is rational. Clearly, the Minkowski minima satisfy the following two properties:

$$\begin{array}{l} 1.\ \lambda_1 \text{ is the length of a shortest nonzero lattice vector;} \\ 2.\ \lambda_1 \leq \lambda_2 \leq \cdots \lambda_n. \end{array} \tag{3}$$

In the above example, $\lambda_1 \approx 1.0$ and $\lambda_2 \approx 1.5$.

We then have the following definition of Minkowski-reduced basis for a lattice.

Definition 3 (Minkowski-reduced) *A lattice basis $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is called Minkowski-reduced if $\|\mathbf{b}_i\|_2 = \lambda_i$ for $i = 1, 2, ..., n$.*

3.2 LLL-reduced bases

Suppose that the columns $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n$ of matrix B are linearly independent and the application of the Gram-Schmidt orthogonalization yields the decomposition

$$B = Q^* U, \tag{4}$$

where Q^* has orthogonal columns $\mathbf{q}_i^*$ and $U = [u_{i,j}]$ is upper triangular with a unit diagonal, that is, $u_{i,j} = 0$ for $i > j$ and $u_{i,i} = 1$. Setting the diagonal matrix $D = \text{diag}(d_i)$ with $d_i = \|\mathbf{q}_i^*\|_2$, we further decompose Q^*:

$$Q^* = QD,$$

then $Q = [\mathbf{q}_1 \ ... \ \mathbf{q}_n]$ has orthonormal columns. Let $R = [r_{i,j}] = DU$, then we have the QR decomposition [5]:

$$B = QR. \tag{5}$$

Thus we have the following relations:

$$\mathbf{q}_i = \mathbf{q}_i^*/\|\mathbf{q}_i^*\|_2, \quad r_{i,i} = d_i = \|\mathbf{q}_i^*\|_2, \quad \text{and} \quad r_{i,j} = r_{i,i}u_{i,j} = d_i u_{i,j}. \tag{6}$$

To reduce the lengths of basis vectors, Hermite introduced a weak notion of reduction [13, Page 37].

Definition 4 (Size-reduced) *A lattice basis $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is called size-reduced if the Gram-Schmidt orthogonalization (4) of $B = [\mathbf{b}_1 \ \mathbf{b}_2 \ ... \ \mathbf{b}_n]$ satisfies*

$$|u_{i,j}| \leq \frac{1}{2}, \quad \textit{for } 1 \leq i < j \leq n$$

or its QR decomposition (5) satisfies

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

Often, size-reduced is a necessary condition for a reduced basis.

The LLL-reduced basis is defined as follows [10]. The columns $\mathbf{b}_i$ of a full column rank matrix B form an *LLL-reduced basis* for a lattice if the matrices Q^* and U in the decomposition (4) satisfy

$$|u_{i,j}| \leq 1/2, \quad j > i \quad \text{(size-reduced)},$$

and

$$\|\mathbf{q}_i^*\|_2^2 + u_{i-1,i}^2\|\mathbf{q}_{i-1}^*\|_2^2 \geq \omega\|\mathbf{q}_{i-1}^*\|_2^2,$$

where $1/4 < \omega < 1$. In terms of the QR decomposition (5), using the relations in (6), we have the following definition [11].

Definition 5 (LLL-Reduced) *Given an $\omega \in (0.25, 1.0)$, a lattice basis $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is called LLL-reduced if the upper triangular matrix R in the decomposition (5) of $B = [\mathbf{b}_1 \ \mathbf{b}_2 \ ... \ \mathbf{b}_n]$ satisfies*

$$|r_{i,j}| \leq |r_{i,i}|/2, \quad j > i \quad \textit{(size-reduced)}, \tag{7}$$

and

$$r_{i,i}^2 + r_{i-1,i}^2 \geq \omega\, r_{i-1,i-1}^2. \tag{8}$$

Since the conditions in the above definition solely depend on R, we may simply call R LLL-reduced. In the rest of the paper, we adopt this QR decomposition approach.

In addition to the size-reduced condition (7) in the above definition, the second condition (8) imposes some order on the column norms of the $(i-1, i)$ main diagonal two-by-two block of R.

To justify that an LLL-reduced basis is a reasonable approximation of a Minkowski-reduced basis, it is shown in [10] that if $\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n$ form an LLL-reduced basis for a lattice, then

$$\eta^{1-i}\lambda_i^2 \leq \|\mathbf{b}_i\|_2^2 \leq \eta^{n-1}\lambda_i^2, \tag{9}$$

where $\eta = (\omega - 1/4)^{-1}$ and $\lambda_1, \lambda_2, ..., \lambda_n$ are the Minkowski minima. In particular, when $\omega = 3/4$, then $\eta = 2$ and

$$2^{1-i}\lambda_i^2 \leq \|\mathbf{b}_i\|_2^2 \leq 2^{n-1}\lambda_i^2.$$

3.3 HKZ-reduced bases

In the nineteenth century, Korkine and Zolotarev, see [13, Page 37] and [8, 9], proposed a definition of a reduced basis by strengthening Hermite's size-reduction.

Definition 6 (HKZ-reduced) *A lattice basis $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is called HKZ-reduced if the upper triangular matrix R in the decomposition (5) of $B = [\mathbf{b}_1\ \mathbf{b}_2\ ...\ \mathbf{b}_n]$ is size-reduced and for each trailing $(n-i+1)$-by-$(n-i+1)$ submatrix, $1 \leq i < n$, its first column is a shortest nonzero vector in the lattice generated by the submatrix.*

Since HKZ-reduced considers the trailing $(n-i+1)$-by-$(n-i+1)$ submatrix while LLL-reduced considers only the leading two-by-two block of the trailing submatrix, an HKZ-reduced basis is LLL-reduced for any $\omega \in (0.25, 1.0)$.

3.4 Optimally reduced bases

A lattice may not have a Minkowski-reduced basis, first noticed by Korkine and Zolotarev in the nineteenth century [13, Page 33]. For example, consider the lattice basis formed by the columns of the following matrix

$$\begin{bmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

All its Minkowski minima equal two, since $[0\ 0\ 0\ 0\ 2]^{\mathrm{T}}$ is a lattice vector linearly independent of the first four columns and its length is two. However, this vector and the first four columns of the above matrix do not form a basis for the lattice. Indeed, they form a basis for a proper sublattice.

Since Minkowski-reduced basis may not always exist, we propose the following definition of reduced basis by strengthening the Minkowski minima.

Definition 7 (Optimally reduced) *A lattice basis $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is called optimally reduced if for each $\mathbf{b}_i$, $i = 1, 2, ..., n$, its length $\|\mathbf{b}_i\|_2 = \min(\|\hat{\mathbf{b}}_i\|_2, \|\hat{\mathbf{b}}_{i+1}\|_2, ..., \|\hat{\mathbf{b}}_n\|_2)$ over all sets $\{\hat{\mathbf{b}}_i, \hat{\mathbf{b}}_{i+1}, ..., \hat{\mathbf{b}}_n\}$ of lattice points such that $\{\mathbf{b}_1, ..., \mathbf{b}_{i-1}, \hat{\mathbf{b}}_i, ..., \hat{\mathbf{b}}_n\}$ form a basis for the lattice.*

In other words, each $\mathbf{b}_i$, for $i = 1, 2, ..., n-1$, is a shortest nonzero lattice vector such that $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_i\}$ is extendable to a basis for the lattice.

In comparison, the Minkowski minima are defined by short and linearly independent lattice vectors, that may not form a basis for the lattice, whereas the above definition is about short basis vectors. Since basis vectors are linearly independent, our definition is more stringent than the Minkowski minima, that is, a Minkowski-reduced basis, if it exists, is an optimally reduced basis in the sense of Definition 7.

It can be shown that when $n = 2$, an HKZ-reduced basis is optimally reduced.

It follows from the above definition that if $\{\mathbf{b}_1, \mathbf{b}_2, ..., \mathbf{b}_n\}$ is an optimally reduced basis defined above, then

- $\mathbf{b}_i$ is a shortest nonzero lattice vector in the sublattice generated by $\mathbf{b}_i, \mathbf{b}_{i+1}, ..., \mathbf{b}_n$;
- $\lambda_1 = \|\mathbf{b}_1\|_2 \le \|\mathbf{b}_2\|_2 \le \cdots \le \|\mathbf{b}_n\|_2$;
- $\|\mathbf{b}_i\|_2 \ge \lambda_i$, $1 \le i \le n$.

Note that the first two properties are consistent with the two properties (3) of Minkowski minima.

3.5 Examples

In this section, using three examples, we compare the reduced basis definitions in the previous section.

First, the following example shows that an LLL-reduced basis may not be Minkowski-reduced and a Minkowski-reduced basis may not be LLL-reduced.

Consider the upper triangular matrix

$$\begin{bmatrix} 1 & 0 & 1/2 \\ 0 & 1 & 0 \\ 0 & 0 & 1-\epsilon \end{bmatrix}.$$

When $1 - \sqrt{\omega} < \epsilon < 1$, this matrix is not LLL-reduced for any $\omega \in (0.25, 1.0)$. On the other hand, when $0 < \epsilon \le 1 - \sqrt{3}/2$, it is Minkowski-reduced, thus optimally reduced, since $(1-\epsilon)^2 + 1/4 \ge 1$. Thus when $1 > \omega > \sqrt{3}/2$ and $1 - \sqrt{\omega} < \epsilon \le 1 - \sqrt{3}/2$, the above matrix is not LLL-reduced but Minkowski-reduced or optimally reduced. Now, permuting the last two columns and last two rows of A, we get

$$\begin{bmatrix} 1 & 1/2 & 0 \\ 0 & 1-\epsilon & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

It can be verified that when $0 < \epsilon \le 1 - \sqrt{3}/2$, the above matrix is HKZ-reduced, thus LLL-reduced for any $\omega \in (0.25, 1.0)$, but not Minkowski-reduced or optimally reduced, since the third column is shorter than the second.

In general, since $\omega < 1$, the first vector in an LLL-reduced basis may not be a shortest lattice point. In other words, an LLL-reduced basis may not satisfy the two properties (3) of the Minkowski minima. The above example shows that this is because the condition (8) in the Definition 5 considers only a two-by-two diagonal block.

Second, recalling the matrix

$$C = \begin{bmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

in the previous section. Its corresponding lattice has no Minkowski-reduced basis. The trailing 2-by-2 submatrix indicates that it is not LLL-reduced when $\omega > 0.5$, thus not HKZ-reduced. However, it is optimally reduced defined in Definition 7. The basis formed by the columns of

$$D = \begin{bmatrix} 2 & 0 & 1 & -1 & -1 \\ 0 & 2 & 1 & -1 & -1 \\ 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 1 & -1 & -1 \end{bmatrix}$$

is HKZ-reduced. It can be verified that it is LLL-reduced for any $\omega \in (0.25, 1.0)$. Obviously, the columns of C form a shorter basis than those of D. Thus D is not optimally reduced.

Finally, we consider the matrix

$$A = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} & \cdots & -\frac{1}{2} \\ & 1 & -\frac{1}{2} & \cdots & -\frac{1}{2} \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & -\frac{1}{2} \\ & & & & 1 \end{bmatrix} \qquad (10)$$

of order n. It is HKZ-reduced, thus LLL-reduced. However, when $n > 2$, it is not optimally reduced. Indeed, when $n = 3$, the columns of the matrix

$$B = \begin{bmatrix} 1 & -\frac{1}{2} & 0 \\ 0 & 1 & \frac{1}{2} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 1 & -\frac{1}{2} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

form an optimally reduced or Minkowski-reduced basis.

4 The LLL Algorithm

Given a matrix A, whose columns form a basis for a lattice L, the LLL algorithm computes a matrix B, whose columns form an LLL-reduced basis, in the sense of Definition 5, for L. The algorithm consists of two stages. The first stage computes the QR decomposition [5]:

$$A = Q_1 S,$$

where Q_1 has orthonormal columns and $S = [s_{i,j}]$ is upper triangular. In the second stage, the LLL algorithm computes the QRZ decomposition:

$$S = Q_2 R Z^{-1}, \tag{11}$$

where Q_2 is orthogonal, Z is unimodular, and R is LLL-reduced defined in Definition 5. The parameter ω in (8) controls the termination of the algorithm. The smaller the ω is, the sooner the algorithm terminates. Let $Q = Q_1 Q_2$, then $A = QRZ^{-1}$. Thus the columns of $B = QR = AZ$ form an LLL-reduced basis for L.

The second stage essentially consists of two procedures that impose the two conditions (7) and (8). When $|r_{i,j}| > |r_{i,i}|/2$, for some i and $j > i$, the following procedure is invoked to enforce the size-reduction condition (7).

Procedure 1 (`Decrease`**(i,j))** . *Given R and Z, calculate $\gamma = \lceil r_{i,j}/r_{i,i} \rfloor$ ($\lceil a \rfloor$ denotes an integer that is closest to a), form $Z_{ij} = I_n - \gamma \mathbf{e}_i \mathbf{e}_j^{\mathrm{T}}$, where $\mathbf{e}_i$ is the ith unit vector, and apply Z_{ij} to both R and Z:*

$$R \leftarrow RZ_{ij} \quad \textit{and} \quad Z \leftarrow ZZ_{ij}.$$

Thus if $|r_{i,i}| < 2|r_{i,j}|$ in the current R, then in the updated R, we have $|r_{i,i}| \geq 2|r_{i,j}|$ satisfying the condition (7).

When the condition (8) is not satisfied, provided that $2|r_{i-1,i}| \leq |r_{i-1,i-1}|$, the columns $i-1$ and i are swapped and then the upper triangular structure of R is restored. When this happens, the algorithm steps back to recheck the conditions (7) and (8). For details of the algorithm, see [10] or [11].

5 A New Reduction Algorithm

We present an algorithm that computes an approximate of an optimally reduced basis defined in Definition 7.

5.1 Basic idea

Suppose that $A_n = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n]$, m-by-n ($m \geq n$), is a generator matrix for a lattice L. The basic idea behind the algorithm is:

1. Find a shortest nonzero lattice vector $\mathbf{b}_1$ in the lattice L;

2. Find a unimodular matrix that transforms the basis $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n$ into a new basis $\mathbf{b}_1$, $\tilde{\mathbf{b}}_2, ..., \tilde{\mathbf{b}}_n$ for L;

3. Deflate A_n to $A_{n-1} = [\tilde{\mathbf{b}}_2 \ ... \ \tilde{\mathbf{b}}_n]$, then the columns of A_{n-1} form a basis for an $(n-1)$-dimensional sublattice of L;

4. Size reduce A_{n-1};

5. Repeat the above steps until the dimension of the sublattice is deflated to one.

For step 1, there are methods for finding a shortest nonzero lattice point. See [1] and [13, Chapter 2], for example. The LLL algorithm can be used as a preprocessor to significantly accelerate the speed.

Step 2 is the key to the algorithm. It solves the following problem: Given a basis $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n\}$ for a lattice L and a shortest nonzero lattice point $\mathbf{b}_1 = A_n\mathbf{z}$, extend $\mathbf{b}_1$ to a new basis for L. In [7], Kannan gave an algorithm for this problem. Here we present a novel unimodular matrix transformation method.

A sufficient and necessary condition that $\mathbf{b}_1 = A_n\mathbf{z}$ is extendable to a basis is that the entries $z_1, z_2, ..., z_n$ of $\mathbf{z}$ have no common factor other than ± 1 [3, Page 14], that is, $\gcd(z_1, z_2, ..., z_n) = \pm 1$. If $\mathbf{b}_1 = A_n\mathbf{z}$ is a shortest nonzero lattice point, then $\gcd(z_1, z_2, ..., z_n) = \pm 1$, since otherwise if $\mathbf{z} = \alpha\tilde{\mathbf{z}}$ for some $|\alpha| > 1$ and $\tilde{\mathbf{z}} \in Z^n$, then $A_n\tilde{\mathbf{z}}$ would be shorter than $A_n\mathbf{z}$, a contradiction. Given $\mathbf{b}_1$, how do we extend it to a basis for L_n? In terms of matrices, that is to find a unimodular matrix Z whose first column is $\mathbf{z}$, in other words, $Z\mathbf{e}_1 = \mathbf{z}$, where $\mathbf{e}_1$ is the first unit vector. Then $A_nZ\mathbf{e}_1 = A_n\mathbf{z} = \mathbf{b}_1$, meaning that the first column of A_nZ, whose columns form a basis for L_n, is $\mathbf{b}_1$, as desired. In other words, the unimodular matrix Z transforms a basis into another basis whose first basis vector is $\mathbf{b}_1$. In the following section, we show how to compute a unimodular Z whose first column is a given integer vector $\mathbf{z} = [z_i]$, where $\gcd(z_i) = \pm 1$.

5.2 A unimodular transformation

If the first column of a unimodular matrix Z is a given integer vector $\mathbf{z}$, then $Z^{-1}\mathbf{z} = \mathbf{e}_1$, which says that Z^{-1} transforms $\mathbf{z}$ into the first unit vector $\mathbf{e}_1$. Thus the problem of finding a unimodular matrix with a specified first column is equivalent to the problem of transforming an integer vector into the first unit vector using a unimodular matrix.

We first present a plane unimodular transformation.

Algorithm 1 **(`Unim2`(p, q))** *Let* $[p \;\; q]^{\mathrm{T}}$ *be a nonzero integer vector and* $\gcd(p, q) = d$. *Using the extended Euclidean algorithm, find integers* a *and* b *such that* $ap + bq = d$. *The integer matrix*

$$M = \begin{bmatrix} p/d & -b \\ q/d & a \end{bmatrix}, \tag{12}$$

is unimodular and

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

The above algorithm shows that given a nonzero integer vector $[p \;\; q]^{\mathrm{T}}$, $\gcd(p, q) = d$, we can construct an integer unimodular matrix (12) whose first column is $[p/d \;\; q/d]^{\mathrm{T}}$. Moreover, its inverse can be applied to $[p \;\; q]^{\mathrm{T}}$ to annihilate its second entry. In particular, if $\gcd(p, q) = \pm 1$, then $[p \;\; q]^{\mathrm{T}}$ can be transformed into the first unit vector. In the trivial case when $p = 0$ and $q = \pm 1$, M is a permutation matrix.

Now we consider the general case. Let $\mathbf{z} = [z_i]$ be a nonzero integer vector and $\gcd(z_i) = \pm 1$, then a sequence of plane unimodular transformations described above can be applied to transform $\mathbf{z}$ into the first unit vector. Consequently, given the integer coordinate vector $\mathbf{z}$ of a lattice point under a given basis for the lattice, we can transform the given basis into a new one whose first vector is the given lattice point, as shown in the following algorithm,

Procedure 2 (Transform($k, \mathbf{z}$)) *Given R, Z, Q, and an integer vector $\mathbf{z} \in Z^{n-k+1}$, $1 \leq k \leq n-1$, such that* $\gcd(z_i) = \pm 1$,

```
for $j = n - k + 1$ downto 2
    $M_j =$ Unim2$(z_{j-1}, z_j)$;
    $U_j = \mathrm{diag}(I_{j-2}, M_j, I_{n-j-k+1})$;
    $\mathbf{z} \leftarrow U_j^{-1}\mathbf{z}$;
    $Z_j = \mathrm{diag}(I_{k-1}, U_j)$;
    $R \leftarrow RZ_j$;
    $Z \leftarrow ZZ_j$;
    find a plane reflection $Q_j$ to restore the structure of $R$;
    $R \leftarrow Q_jR$;
    $Q \leftarrow QQ_j$;
end
```

As in the LLL algorithm, each time when a new basis is constructed, we enforce the size-reduced condition (7).

Procedure 3 (BlockDecrease(k)) *Given R and Z,*

```
for $i = n - 1$ downto 1
    for $j = n$ downto $\max(i + 1, k)$
        Decrease$(i, j)$;
    end
end
```

Putting all things together, we present our first lattice basis reduction algorithm. Note that in the kth iteration, the structure of $R_k = [r_{i,j}]$ is: n-by-$(n-k+1)$ and $r_{i,j} = 0$, for $i > j+k-1$.

Algorithm 2 (One-pass algorithm) *Given a lattice generator matrix A, this algorithm computes the QRZ decomposition $A = QRZ^{-1}$, where the columns of $QR = AZ$ form an approximation of an optimally reduced basis defined by Definition 7 for the lattice.*

```
Initial QRZ decomposition: $A = QR$ and $Z = I$;
BlockDecrease(1);
for $k = 1$ to $n - 1$
    Let $R_k$ consist of the last $n - k + 1$ columns of $R$;
    Find a nonzero integer vector $\mathbf{z}$ so that $R_k\mathbf{z}$ is a shortest nonzero point
        in the sublattice generated by $R_k$;
    Transform$(k, \mathbf{z})$;
    BlockDecrease$(k)$;
end
```

As shown above, our algorithm computes a basis whose first vector is a shortest nonzero lattice point in the lattice. The subsequent basis vectors are shortest nonzero lattice points in some size-reduced sublattices. Thus the above algorithm computes an approximation of a reduced basis defined by Definition 7.

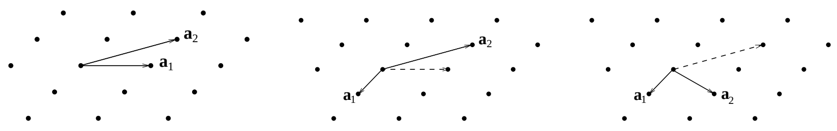

Figure 3: From left to right, starting from the basis formed by the columns of A (1), Algorithm 2 first finds a shortest nonzero lattice point as the first basis vector, then size reduces the basis.

Given a Minkowski-reduced basis, since it is optimally reduced, our algorithm will keep the basis unchanged. In contrast, since a Minkowski-reduced basis may not be LLL-reduced as shown in section 3.5, the LLL algorithm may change a Minkowski-reduced basis into a longer basis.

Like any reduction algorithm that requires size reduction, our algorithm uses the integer unimodular matrices of the form

$$\begin{bmatrix} 1 & -\gamma \\ 0 & 1 \end{bmatrix}$$

in the procedure `Decrease`. However, our algorithm differs from the LLL algorithm in that in addition to the above unimodular matrix, it also uses the unimodular matrices of the form (12) whereas the LLL algorithm exclusively uses permutation, which is a trivial case of (12) when $p = 0$ and $q = \pm 1$. For example, when Algorithm 2 is applied to the lattice generator matrix A in (1), it first finds a shortest nonzero lattice point

$$A \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} -0.7 \\ -0.7 \end{bmatrix}$$

and then extends the shortest vector to a new basis

$$\begin{bmatrix} -0.7 & 2.7 \\ -0.7 & 0.7 \end{bmatrix} = A \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}.$$

Finally, after a size reduction, a reduced basis is obtained:

$$\begin{bmatrix} -0.7 & 1.3 \\ -0.7 & -0.7 \end{bmatrix} = A \begin{bmatrix} 1 & 2 \\ -1 & -1 \end{bmatrix}.$$

Figure 3 depicts the process.

When the LLL algorithm is applied to the same matrix A in (1), it first size reduces the second vector,

$$\begin{bmatrix} 2.0 & 0.7 \\ 0 & 0.7 \end{bmatrix} = A \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}.$$

followed by permuting two basis vectors,

$$\begin{bmatrix} 0.7 & 2.0 \\ 0.7 & 0 \end{bmatrix} = A \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix}.$$

then again applies size reduction

$$\begin{bmatrix} 0.7 & 1.3 \\ 0.7 & -0.7 \end{bmatrix} = A \begin{bmatrix} -1 & 2 \\ 1 & -1 \end{bmatrix}.$$

Figure 4 shows the process.

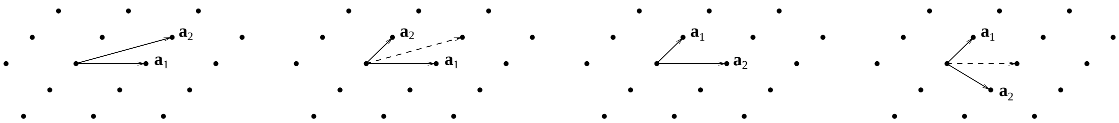

Figure 4: From left to right, starting from the basis formed by the columns of A (1), the LLL-algorithm first applies a size reduction, followed by swapping the two basis vectors, then size reduces the second basis vector against the first.

5.3 Ordering lengths

The basis vectors produced by Algorithm 2 may not be ordered in their lengths, which is a property of an optimally reduced basis defined in Definition 7. It is illustrated by the following simple example.

Let the lattice generator matrix

$$A = \begin{bmatrix} 1 & 0.62 & 0.42 \\ 0 & 0.15 & -0.22 \\ 0 & 0 & -0.22 \end{bmatrix}.$$

At the end of the first iteration, $k = 1$, the upper triangular matrix R and the unimodular matrix Z in the QRZ decomposition are respectively

$$R \approx \begin{bmatrix} 0.23431 & -0.06359 & 0.10968 \\ & 0.98532 & 0.36666 \\ & & -0.14294 \end{bmatrix} \quad \text{and} \quad Z = \begin{bmatrix} 1 & -2 & -1 \\ -1 & 1 & 1 \\ -1 & 1 & 0 \end{bmatrix}.$$

The third column of R is shorter than the second. Thus in the next iteration, $k = 2$, the algorithm finds that the third column is a shortest lattice point in the sublattice generated by the second and third columns. Then the algorithm swaps the second and third columns and restores the upper triangular structure, resulting the following upper triangular matrix and unimodular matrix:

$$R \approx \begin{bmatrix} 0.23431 & -0.10968 & -0.06359 \\ & 0.39353 & -0.91803 \\ & & -0.35789 \end{bmatrix} \quad \text{and} \quad Z = \begin{bmatrix} 1 & 1 & -2 \\ -1 & -1 & 1 \\ -1 & 0 & 1 \end{bmatrix}.$$

Notice that the above R is not size-reduced. Consequently, at the end of the iteration $k = 2$, a size-reduction is performed and the upper triangular matrix and the unimodular matrix become

$$R \approx \begin{bmatrix} 0.23431 & -0.10968 & -0.04865 \\ & 0.39353 & -0.13096 \\ & & -0.35789 \end{bmatrix} \quad \text{and} \quad Z = \begin{bmatrix} 1 & 1 & 1 \\ -1 & -1 & -2 \\ -1 & 0 & 0 \end{bmatrix}.$$

Now the length 0.38419 of the last column of R is shorter than the length 0.40853 of the second column. This also shows that after size-reduction, the sublattice generated by the last two columns of R is changed. To enforce the ordering of the basis vectors, we present the following modification of Algorithm 2: After the size-reduction at the end of the kth iteration in Algorithm 2, we check the column norms. If there is a column j, $j > k$, whose length is shorter than a previous column i, $i \leq k$, we retreat to the ith step.

Algorithm 3 (Modified algorithm) *Given a lattice generator matrix A, this algorithm computes the QRZ decomposition $A = QRZ^{-1}$, where the columns of $QR = AZ$ form an approximation of an optimally reduced basis for the lattice in the sense of Definition 7.*

```
Initial QRZ decomposition: $A = QR$ and $Z = I$;
BlockDecrease(1);
$k = 1$;
while $k < n$
    Let $R_k$ consist of the last $n - k + 1$ columns of $R$;
    Find a nonzero integer vector $\mathbf{z}$ so that $R_k\mathbf{z}$ is a shortest nonzero point
        in the sublattice generated by $R_k$;
    Transform($k, \mathbf{z}$);
    BlockDecrease($k$);
    $next = k + 1$;
    Partition $R = [R_1 \ \ R_2]$, where $R_1$ consists of the first $k$ columns of $R$;
    Search for the smallest index $p$ of a column in $R_1$ that is longer than
        the shortest column in $R_2$;
    if $p \leq k$, $next = p$;
    $k = next$;
end
```

Since the above algorithm produces a basis not longer than the one computed by Algorithm 2, the algorithm gives a better approximation of an optimally reduced basis.

6 Complexity and Termination

The major computational cost of Algorithm 2 is the search for a shortest nonzero lattice point in a sublattice. There are exact and approximate methods for the shortest lattice point problem. See [13, Page 39] and references there. In particular, for example, the LLL algorithm or the blockwise algorithms can be used as a preprocessing to provide an approximate shortest nonzero lattice point and significantly speed up the search, such as sphere decoding, for an exact shortest nonzero lattice point.

The other major contributor to the computation is the procedure `Transform`. For each value of k, it requires $O(n(n-k))$ floating-point operations. Thus the total cost from `Transform` is $O(n^3)$.

The first part of Algorithm 3 is identical to Algorithm 2. In the second part, it retreats to an earlier step when necessary. This raises the question of the termination of the program. We will show that the program terminates in finite number of iterations. Indeed, when the program detects that the length of the jth column is shorter than that of the ith column, $j > i$, it retreats to the ith step. It then finds the shortest nonzero lattice point in the sublattice containing the jth column and replaces the ith column. Thus the length of the ith column is reduced by at least the difference between the lengths of the ith and jth columns. Since there are finite number of lattice vectors whose lengths are between the shortest nonzero length and the length of the longest column, the differences among the lengths of the columns are bounded below by a positive constant. Moreover, the length of any nonzero lattice vector is bounded below by the the length of a shortest nonzero lattice vector. Therefore, the length of any column can be reduced by finite number of times, implying that Algorithm 3 terminates.

7 A Measurement

Lattice reduction is to transform a lattice basis into another that is “more” orthogonal, or “more” linearly independent. How do we measure the degree of linear independence of a reduced basis produced by a lattice basis reduction algorithm? The condition number for matrix inversion could be a candidate. However, it does not always precisely measure the degree of linear independence of the columns of a matrix. For example, the condition number of the diagonal matrix

$$\begin{bmatrix} 1 & 0 \\ 0 & 10^k \end{bmatrix}$$

is 10^k when $k > 0$, which is large when k is large. However, the columns are orthogonal, or highly linearly independent. In this section, we propose a more precise measure of the degree of linear independence of basis vectors.

It suffices to consider the upper triangular matrix R in the QRZ decomposition. We begin with the two-dimension case

$$\begin{bmatrix} r_{1,1} & r_{1,2} \\ 0 & r_{2,2} \end{bmatrix}.$$

The cosine of the angle θ_2 between the two columns is

$$\cos\theta_2 = \frac{r_{1,1}r_{1,2}}{|r_{1,1}|\sqrt{r_{1,2}^2 + r_{2,2}^2}}.$$

Denoting the column length $l_2 = \sqrt{r_{1,2}^2 + r_{2,2}^2}$ and assuming $0 < \theta_2 < \pi$, we have

$$\sin\theta_2 = \frac{|r_{2,2}|}{l_2}.$$

Let the determinant of a lattice denote its volume V_2, then

$$V_2 = |r_{1,1}r_{2,2}|.$$

Defining $V_1 = |r_{1,1}|$, $l_1 = |r_{1,1}|$, and $\theta_1 = \pi/2$, we have

$$V_2 = (l_1 \sin\theta_1)(l_2 \sin\theta_2) = V_1(l_2 \sin\theta_2). \tag{13}$$

If we view V_1 as the base, then V_2/V_1 gives the height. Thus $\sin\theta_2 = (V_2/V_1)/l_2$ is the ratio between the height and the length as usual.

Now we generalize (13) into order n by letting

$$V_k = V_{k-1}(l_k \sin\theta_k) = \prod_{i=1}^{k}(l_i \sin\theta_i), \quad 1 \le k \le n.$$

Since $V_k = \prod_{i=1}^{k} |r_{i,i}|$, we have

$$\sin \theta_k = \frac{|r_{k,k}|}{l_k}.$$

Similarly,, $|r_{k,k}| = V_k / V_{k-1}$ can be viewed as the height. Indeed, $[0, ..., 0, r_{k,k}]^T$ is the projection of the kth column onto the subspace orthogonal to the subspace spanned by the first $k-1$ columns of R. Thus, θ_k represents the angle between the kth column and the subspace spanned by the first $k-1$ columns of R. It is then reasonable to use $\sin \theta_k$ as a measure for the degree of linear independence of the kth column from the subspace spanned by the first $k-1$ columns of R. Therefore, we propose a linear independence number σ defined by

$$\sigma^n = \prod_{i=1}^{n} \sin \theta_i = \prod_{i=1}^{n} \frac{|r_{i,i}|}{l_i} = \frac{d(L)}{\prod_{i=1}^{n} l_i} \tag{14}$$

as a measure for the degree of linear independence of a lattice basis. Thus σ is the geometric mean of $\sin \theta_i$. Note that $0 \leq \sigma \leq 1$, and $\sigma = 1$ for any diagonal matrix, and $\sigma = 0$ for any singular matrix. Since the volume $V_n = \prod_{i=1}^{n} |r_{i,i}| = d(L)$ is a constant for a given lattice L, by reducing the lengths l_i, the linear independence number σ in (14) is improved. This explains why reduction algorithms improve the degree of linear independence by reducing the lengths of basis vectors.

For example, the matrix

$$C = \begin{bmatrix} 2 & 0 & 0 & 0 & 1 \\ 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 1 \\ 0 & 0 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

is optimally reduced but not HKZ-reduced. Its condition number and the linear independence number are respectively

$$\kappa_C \approx 4.27 \quad \text{and} \quad \sigma_C \approx 0.8493.$$

The matrix

$$D = \begin{bmatrix} 2 & 0 & 1 & -1 & -1 \\ 0 & 2 & 1 & -1 & -1 \\ 0 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 1 & -1 & -1 \end{bmatrix}$$

is HKZ-reduced for the same lattice, but not optimally reduced. Its condition number and the linear independence number are respectively

$$\kappa_D \approx 4.09 \quad \text{and} \quad \sigma_D \approx 0.8143.$$

Although C is worse conditioned than D, the basis formed by C is better reduced than the basis formed by D, which is revealed by the linear independence numbers. This example shows that the linear independence number defined in (14) is a better measurement for reduced bases.

column	1	2	3	4	5	6	7	8	9	10	11	12
$\|\mathbf{a}_i\|_2^2$	1.0	1.25	1.5	1.75	2.0	2.25	2.5	2.75	3.0	3.25	3.5	3.75
$\|\mathbf{b}_i\|_2^2$	1.0	1.25	1.5	1.75	1.75	1.75	2.0	2.0	2.0	2.25	2.25	2.25

Table 1: Norms of the columns $\mathbf{a}_i$ of A (10) the norms of the columns $\mathbf{b}_i$ of B (15).

8 Experimental Results

In this section, we present our experimental results on comparing Algorithm 3 with the LLL-algorithm. We first compare the algorithms by the lengths of the reduced basis vectors computed by the algorithms. Our criterion is: the shorter the basis vectors, the better. Then we compare the algorithms using both the condition numbers κ and the linear independent numbers σ of the matrices reduced by the algorithms. As we know, the smaller the κ or the closer to one the σ, the better. We present our results on two types of matrices: The matrices of the form A in (10) in section 3.5 and random matrices.

Recalling the matrix A in (10), it is HKZ-reduced, thus LLL-reduced for any $\omega \in (0.25, 1.0)$. However, we have shown that it is not optimally reduced when $n > 2$. Algorithm 3 does not change A for $n < 11$. When $n = 12$, however, it produces an approximation

$$B = \left[\begin{array}{cccccccccccc} 1 & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 0 & 1 & -\frac{1}{2} & -\frac{1}{2} & 0 & 0 & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{2} \\ 0 & 0 & 0 & 1 & 0 & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 1 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \qquad (15)$$

of an optimally reduced basis. Table 1 compares the column norms of A and B. Notice that B has multiple columns of the same length. Interestingly, as n increases, there are more multiple basis vectors of the same length.

Now, we use this example to explain why Algorithm 3 only computes an approximation of an optimally reduced basis. Consider the case when $n = 3$. Applying Algorithm 3 keeps the matrix unchanged. Howver, if we transform A into

$$\widehat{A} = \left[\begin{array}{ccc} 1 & \frac{1}{2} & -\frac{1}{2} \\ 0 & 1 & -\frac{1}{2} \\ 0 & 0 & 1 \end{array}\right] \qquad (16)$$

by adding the first column of A to its second column, then Algorithm 3 produces

$$B = \left[\begin{array}{ccc} 1 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 1 \\ 0 & 1 & 0 \end{array}\right] = \left[\begin{array}{ccc} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & 1 & -\frac{1}{2} \\ 0 & 0 & 1 \end{array}\right] \left[\begin{array}{ccc} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{array}\right],$$

size n	8	12	16	20
LLL or HKZ	23.002	171.46	1179.8	7604.8
Algorithm 3	23.002	15.094	17.059	40.490

Table 2: Condition numbers κ of reduced matrices by applying Algorithm 3 to A in (10) of various sizes. Since A is LLL-reduced, the condition number of an LLL-reduced matrix equals the condition number of A.

size n	8	12	16	20
LLL or HKZ	0.7492	0.6739	0.6186	0.5757
Algorithm 3	0.7492	0.7527	0.7208	0.6828

Table 3: Linear independence numbers σ corresponding to the condition numbers in Table 2.

which is optimally or Minkowski-reduced. The reason is that after the first iteration, the algorithm works on the sublattice generated by the last two columns. While the sublattice generated by the last two columns of $\widehat{A}$ in (16) contains the lattice point $[0 \ \ \frac{1}{2} \ \ 1]$, which is an optimally reduced basis vector, the sublattice generated by last two columns of A does not contain an optimally reduced basis vector. In other words, Algorithm 3 works on some particular sublattices.

Table 2 lists the condition numbers of the reduced matrices by applying the LLL algorithm and Algorithm 3 to A in (10) of various sizes. Since A is LLL-reduced, the condition number of LLL-reduced matrix equals that of A. We know that the condition number of A grows exponentially as n increases.

Table 3 lists the linear independence numbers of the matrices corresponding to Table 2. It shows that in this case the linear independence numbers are consistent with the condition numbers

Now we present our results on random matrices. A random matrix with a predetermined condition number $\kappa \geq 1$ was generated as follows. First, n singular values evenly spaced between 1.0 and κ^{-1} were generated. Then two random orthogonal matrices U and V were obtained from the QR decompositions of two random matrices. Finally, a random matrix with condition number κ was constructed by $U\Sigma V^{\mathrm{T}}$, with Σ the diagonal singular value matrix. In our experiments, the size n was set to 20. Table 4 shows the condition numbers of matrices reduced by the LLL algorithm, the HKZ algorithm, and Algorithm 3. The parameter ω in the LLL algorithm was set to 0.95. Each condition number in the table is an average of five cases. The table shows that all three algorithms significantly improved the conditioning of an ill-conditioned random matrix and Algorithm 3 consistently outperformed the LLL algorithm and the HKZ algorithm. Table 5 lists the linear dependence numbers corresponding to the condition numbers in Table 4. Table 5 shows that Algorithm 3 produces better reduced bases than the LLL algorithm and the HKZ algorithm. In our experiments, occasionally the condition numbers of the matrices produced by Algorithm 3 were slightly worse than those from the LLL algorithm and the HKZ algorithm, whereas their linear independence numbers were always better, i.e., closer to one, than those from the other two algorithms.

original	10^3	10^5	10^7
LLL	17.007	15.392	16.404
HKZ	16.932	15.978	15.360
Algorithm 3	11.399	11.987	12.331

Table 4: Condition numbers κ of random matrices of order 20 and the condition numbers of the reduced matrices by the LLL algorithm, the HKZ algorithm, and Algorithm 3.

original	0.5350	0.4235	0.3370
LLL	0.6823	0.6874	0.6815
HKZ	0.6838	0.6953	0.6870
Algorithm 3	0.6939	0.7046	0.6986

Table 5: Linear independence numbers σ corresponding to the condition numbers in Table 4.

Conclusion

In this paper, we have introduced the definition of an optimally reduced lattice basis and presented an algorithm for computing an approximation of an optimally reduced basis for a lattice. We term it an optimally reduced basis, because a Minkowski-reduced basis, if it exists, is optimally reduced by our definition. The pivotal part of our reduction algorithm is a novel unimodular transformation. Given a lattice vector extendable to a basis, it allows us to transform a lattice basis into another basis containing the given lattice vector. In particular, it can be used to transform a basis for a lattice into another basis that contains a shortest nonzero lattice vector. It distinguishes our algorithm from the LLL algorithm, which, other than size reduction, is restricted to permutations of basis vectors based on their projections.

To compare lattice bases, the currently commonly used measurement is the matrix condition number. The smaller the condition number of a basis matrix, the better the basis. In this paper, we have proposed a quantitative measurement called linear independent number. The closer to one the linear independence number, the better the basis. We believe that in the context of lattice bases this number is a better measurement than the matrix condition number. When basis vectors are orthogonal to each other, the linear independence number equals one, whereas the condition number can be arbitrarily large. Moreover, the linear independence number reveals that shorter basis vectors tend to be "more" orthogonal, because the determinant (volume) of a lattice is a constant. Our experiments have shown that our algorithm produces shorter basis vectors than the HKZ and LLL algorithms. Thus our algorithm is suitable for the applications where a shortest basis is desirable. This advantage comes with the price of more computational cost. Nevertheless, the efficiency of our algorithm can be improved, for example, by replacing an exact method for finding a shortest nonzero lattice point by an more efficient but approximate method. The impact of our algorithm, the unimodular matrix transformation, and the linear independent number on applications requires further investigation.

References

[1] Erik Agrell, Thomas Eriksson, Alexander Vardy, and Kenneth Zeger. Closest point search in lattices. *IEEE Transactions on Information Theory*, vol. 48, no. 8, 2002, 2201–2214.

[2] A. Barvinok. *A Course in Convexity.* Graduate Studies in Mathematics 54. American Mathematical Society, Providence, RI, 2002.

[3] J.W.S. Cassels. *An Introduction to the Geometry of Numbers, Second Printing.* Springer-Verlag, Berlin, Heidelberg, 1997.

[4] H. Cohen. *A Course in Computational Algebraic Number Theory.* Graduate Texts in Mathematics 138, Second corrected printing. Springer, Berlin, 1995.

[5] G.H. Golub and C.F. Van Loan. *Matrix Computations, Third Edition.* The Johns Hopkins University Press, Baltimore, MD, 1996.

[6] A. Joux and J. Stern. Lattice reduction: A toolbox for the cryptanalyst. *Journal of Cryptology*, 11(**3**), 1998, 161–185.

[7] Ravi Kannan. Minkowski's convex body theorem and integer programming. *Mathematics of Operations Research*, 3(**12**), 1987, 415–440.

[8] A. Korkine and G. Zolotareff. Sur les formes quadratiques positives ternaires. *Maht. Ann.*, **5**, 1872, 581–583.

[9] A. Korkine and G. Zolotareff. Sur les formes quadratiques. *Maht. Ann.*, **6**, 1873, 336–389.

[10] A.K. Lenstra, H.W. Lenstra, Jr. and L. Lovász. Factorizing polynomials with rational coefficients. *Mathematicsche Annalen*, **261**, 1982, 515–534.

[11] F.T. Luk and D.M. Tracy. An improved LLL algorithm. *Linear Algebra and its Applications*, **428**(2–3), 2008, 441–452.

[12] H. Minkowski. *Geometrie der Zahlen.* Teubner, Leipzig, 1896.

[13] The LLL Algorithm: Survey and Applications. *Information Security and Cryptography, Texts and Monographs.* Editors Phong Q. Nguyen and Brigitte Vallée. Springer Heidelberg Dordrecht London New York, 2010.