ALGORITHMS TO CONSTRUCT MINKOWSKI REDUCED AND HERMITE REDUCED LATTICE BASES

Bettina HELFRICH

Fachbereich Mathematik, Johann Wolfgang Goethe-Universität, 6000 Frankfurt am Main, Fed. Rep. Germany

Communicated by M.S. Paterson
Received February 1985
Revised August 1985

Abstract. Up to now, the problem of constructing Minkowski reduced lattice bases has been solved only for the two- and three-dimensional case. This paper presents an algorithm to solve the problem for arbitrary dimension. For fixed dimension, the runtime is polynomial. The algorithm hinges on the previous reduction algorithms of Lenstra, Lenstra and Lovász (1982) and Kannan (1983). Moreover, we shall improve Kannan's algorithm to construct Hermite reduced lattice bases.

1. Introduction

A *lattice* in $\mathbb{R}^d$ is a set $L=\sum_{i=1}^{n} \boldsymbol{b}_i\mathbb{Z}$, where $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ are linearly independent. The ordered set $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is a *basis* of L, and n is the *dimension* of L. A basis of L is not unique.

Already in the work of Gauss [4], the idea of selecting certain 'reduced' bases occurs (the early work on this topic is formulated in terms of quadratic forms instead of lattices). 'Reduced' bases have 'nice' properties, which usually means that they consist of 'short' and 'fairly orthogonal' vectors.

The definition of a 'reduced' basis is not canonical. Hermite [6] generalized the two-dimensional definition of Gauss to arbitrary dimensions. A strengthened definition given by Korkine and Zolotareff [10] is in the literature [3] attributed to Hermite, and so the corresponding bases are called 'Hermite reduced'. In the 1890s, Minkowski [16] defined 'Minkowski reduced' bases, requiring that each basis vector is 'as short as possible'.

It is not known whether the problem of constructing a lattice vector that is shortest with respect to the L_2-norm is NP-hard (but the problem is known to be NP-hard with respect to the L_∞-norm [19]).

In 1982, Lenstra, Lenstra and Lovász [15] achieved a breakthrough by constructing 'LLL reduced' bases and thereby approximating the shortest nonzero lattice vector up to a factor $2^{(n-1)/2}$. Although this definition of reducedness is weak, in the meantime, their algorithm has been widely applied, for example, to cryptography

0304-3975/85/$3.30

[13], factorisation of polynomials [15, 18], linear diophantine approximation [1, 12], etc.

In 1983, Kannan [8] presented an algorithm to construct Hermite reduced bases, which in particular contain a shortest nonzero lattice element. The algorithm is exponential, but polynomial for fixed dimension. His algorithm yields the asymptotically best known algorithm for integer linear programming.

In 1984, Schnorr [17] defined a hierarchy of polynomial time algorithms that contains for every $k > 1$ an algorithm approximating the shortest lattice element up to a factor $(k^{1/k})^n$. The runtime of the algorithms exponentially grows in k.

Like Hermite reduced bases, Minkowski reduced bases contain a shortest nonzero lattice vector. Moreover, the concept of Minkowski reduction is of fundamental importance in the geometry of numbers (cf. [2] also for a guide to the literature). Up to now, however, Minkowski reduced bases could be constructed only for dimensions 2 and 3 [11]. In this paper, an algorithm to solve the problem for arbitrary dimension is presented. Again, its runtime is exponential, but polynomial for fixed dimension.

In Section 2 we shall present the basic definitions and facts concerning lattices. In Section 3 we shall improve Kannan's algorithm for the construction of Hermite reduced bases. The dependence of the dimension n in the estimates for the number of arithmetical operations performed will decrease from $(4n)^{1.5n+O(1)}$ to $n^{0.5n+o(n)}$. A better runtime analysis for the algorithm to solve the closest lattice point problem presented in [8] will be performed in Section 4. Section 5 will present the new algorithm to construct Minkowski reduced lattice bases.

2. Lattices

Let $\mathbb{R}^d$ denote the usual d-dimensional Euclidean space with norm $|\cdot|$ and standard dot product $\langle \cdot, \cdot \rangle$. The linear subspace generated by some subset of $\mathbb{R}^d$ is denoted by $\langle\!\langle \cdot \cdots \cdot \rangle\!\rangle$, its orthogonal complement by $\langle\!\langle \cdot \cdots \cdot \rangle\!\rangle^{\perp}$. Ordered finite sets are denoted by round brackets $(\cdot, \ldots, \cdot)$.

Now, let $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{R}^d$ be a set of linearly independent vectors and $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) = \sum_{i=1}^{n} \boldsymbol{b}_i \mathbb{Z}$ be the lattice *spanned* by them. For simplicity of notation, $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ will often be abbreviated by L.

The *determinant* $\det(L)$ of L is defined as $\det((\langle \boldsymbol{b}_i, \boldsymbol{b}_j \rangle)_{1 \leq i,j \leq n})^{1/2}$, it is independent of the choice of the basis [2].

A vector $\boldsymbol{v} \in L$ is a *shortest nonzero lattice element* if $\boldsymbol{v} \neq \boldsymbol{0}$ and $|\boldsymbol{v}| \leq |\boldsymbol{w}|$ for all $\boldsymbol{w} \in L \setminus \{\boldsymbol{0}\}$. The set of shortest nonzero lattice elements is finite. In the following, the word 'nonzero' will often be omitted in this context.

In order to bound the length of shortest lattice vectors from above, the *Hermite constants* γ_n for $n \in \mathbb{N}$ are defined by

$$\gamma_n = \sup_{L\ n-\text{dimensional lattice}} \left\{ \inf_{\boldsymbol{v} \in L \setminus \{\boldsymbol{0}\}} \{|\boldsymbol{v}|^2 \det(L)^{-2/n}\} \right\}.$$

These constants are explicitly known only for $n \leqslant 8$ [2]. To deal with γ_n, we state the following proposition.

Proposition 2.1

(a) *For all* $n \in \mathbb{N}$, $\gamma_n \leqslant n$.

(b) $$\frac{1}{2\pi e} \leqslant \liminf_{n\to\infty} \frac{1}{n}\gamma_n \leqslant \limsup_{n\to\infty} \frac{1}{n}\gamma_n \leqslant \frac{1}{\pi e}.$$

Proof. (a) follows from Minkowski's convex body theorem [20], and (b) is proved in [14]. □

Let $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be a fixed basis. For $\boldsymbol{a} \in \mathbb{R}^d$ and $k \in \{1, \ldots, n\}$, we denote by $\boldsymbol{a}(k)$ (respectively $L_k(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$) the projection of $\boldsymbol{a}$ (respectively $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$) on $\langle\!\langle \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k-1} \rangle\!\rangle^{\perp}$. In particular, $\boldsymbol{a}(1) = \boldsymbol{a}$ and $L_1(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) = L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$. When no confusion can arise, we shall use L_k as a short notation for $L_k(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

The projections of the basis vectors themselves are illustrated in Fig. 1. Obviously, $(\boldsymbol{b}_k(k), \ldots, \boldsymbol{b}_n(k))$ is a basis of L_k for $k = 1, \ldots, n$.

$$\begin{array}{lccccccc}
L_1 & \boldsymbol{b}_1(1) & \boldsymbol{b}_2(1) & \cdots & \boldsymbol{b}_k(1) & \boldsymbol{b}_{k+1}(1) & \cdots & \boldsymbol{b}_n(1) \\
L_2 & & \boldsymbol{b}_2(2) & \cdots & \boldsymbol{b}_k(2) & \boldsymbol{b}_{k+1}(2) & \cdots & \boldsymbol{b}_n(2) \\
\vdots & & & \ddots & \vdots & \vdots & & \vdots \\
L_k & & & & \boldsymbol{b}_k(k) & \boldsymbol{b}_{k+1}(k) & \cdots & \boldsymbol{b}_n(k) \\
L_{k+1} & & & & & \boldsymbol{b}_{k+1}(k+1) & \cdots & \boldsymbol{b}_n(k+1) \\
\vdots & & & & & & \ddots & \vdots \\
L_n & & & & & & & \boldsymbol{b}_n(n)
\end{array}$$

Fig. 1. Projections of a basis.

Applying Gram–Schmidt orthogonalisation, the vectors $\boldsymbol{b}_k(i)$, $1 \leqslant k \leqslant i \leqslant n$, can be obtained by

$$\boldsymbol{b}_k(i) = \boldsymbol{b}_k - \sum_{j=1}^{i-1} \mu_{kj}\boldsymbol{b}_j(j), \quad \text{where } \mu_{kj} = \frac{\langle \boldsymbol{b}_k, \boldsymbol{b}_j(j)\rangle}{\langle \boldsymbol{b}_j(j), \boldsymbol{b}_j(j)\rangle}.$$

Lemma 2.2

(a) $$\det(L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)) = \prod_{i=1}^{n} |\boldsymbol{b}_i(i)|.$$

(b) *Let* $\boldsymbol{v} = \sum_{i=1}^{n} v_i\boldsymbol{b}_i = \sum_{i=1}^{n} v_i' \boldsymbol{b}_i(i)$. *Then, for* $i = 1, \ldots, n$, $v_i' = v_i + t_i$, *where*

$$t_i = \sum_{j=i+1}^{n} v_j \frac{\langle \boldsymbol{b}_j(i), \boldsymbol{b}_i(i)\rangle}{\langle \boldsymbol{b}_i(i), \boldsymbol{b}_i(i)\rangle}.$$

(c) *For* $k=1,\ldots,n$, *let*

$$d_k = \prod_{i=1}^{k} |\boldsymbol{b}_k(k)|^2 = \det(((\langle \boldsymbol{b}_i, b_j\rangle))_{1\leqslant i,j\leqslant k}).$$

If $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)\subset\mathbb{Z}^d$, *then* d_k *is integral and, for* $\boldsymbol{a}\in\mathbb{Z}^d$, $\boldsymbol{a}(k)d_{k-1}\in\mathbb{Z}^d$ *for all* $k=1,\ldots,n$.

Proof. (a) can be seen by interpreting det as a volume function, (b) is left to the reader, and for (c) we refer to [8]. □

The basis $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$ is *proper* if $|\mu_{kj}|\leqslant\frac{1}{2}$ holds for $1\leqslant j<k\leqslant n$. A basis can easily be transformed into a proper one by replacing $\boldsymbol{b}_k$ by $\boldsymbol{b}_k-[\mu_{kj}]\boldsymbol{b}_j$ for $k=2,\ldots,n$ and $j=k-1,\ldots,1$ in this order. Here, $[\mu_{kj}]$ denotes the integer nearest to μ_{kj}.

Now let $\boldsymbol{v}(k+1)=\sum_{i=k+1}^{n} v_i\boldsymbol{b}_i(k+1)\in L_{k+1}$. Then, computing

$$\bar{\boldsymbol{v}}(k)=\sum_{i=k+1}^{n} v_i\boldsymbol{b}_i(k) \quad\text{and}\quad \boldsymbol{v}(k)=\bar{\boldsymbol{v}}(k)-\boldsymbol{b}_k(k)\frac{\langle\bar{\boldsymbol{v}}(k),\boldsymbol{b}_k(k)\rangle}{\langle\boldsymbol{b}_k(k),\boldsymbol{b}_k(k)\rangle}$$

yields a shortest vector $\boldsymbol{v}(k)\in L_k$ whose projection on L_{k+1} is $\boldsymbol{v}(k+1)$ (respectively, $|\langle\boldsymbol{v}(k),\boldsymbol{b}_k(k)\rangle/\langle\boldsymbol{b}_k(k),\boldsymbol{b}_k(k)\rangle|\leqslant\frac{1}{2}$). We call this *lifting* the vector $\boldsymbol{v}(k+1)$ on L_k. If the numbers $v_{k+1},\ldots,v_n$ are not known, they can be computed by solving the system $\boldsymbol{v}(k+1)=\sum_{i=k+1}^{n} v_i\boldsymbol{b}_i(k+1)$ of linear equations.

The vectors involved with lifting in this paper will be rational ones, and, by Lemma 2.2(c), their denominators will be smaller than B^n for some $B\in\mathbb{R}$, $B\geqslant 2$. Moreover, their Euclidean length will be at most nB. As the reader may verify, in this case lifting can be achieved in $O(n^2 d\log B)$ arithmetical operations on numbers of *binary length* $O(n^2\log B)$. 'Binary length' denotes the number of bits required to express a rational or integral number.

For a vector $\boldsymbol{v}=\sum_{i=1}^{n} v_i\boldsymbol{b}_i$ there is a basis of $L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$ containing $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k,\boldsymbol{v})$ if and only if $\gcd\{v_{k+1},\ldots,v_n\}=1$ [20]. In this case, we shall say that $\boldsymbol{v}$ *can be extended to a basis* (*of* L) *with* $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k)$.

If $\gcd\{v_1,\ldots,v_n\}=1$, $\boldsymbol{v}$ is called *primitive in* L. The shortest lattice vector is primitive and can be extended to a basis, and each vector $\boldsymbol{v}$ for which $\boldsymbol{v}(k+1)$ is primitive in L_{k+1} can be extended to a basis of L with $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k)$.

We are now ready to define the 'reduced bases' we are going to deal with.

Definition 2.3. The basis $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$ of the lattice $L\subset\mathbb{R}^d$ is called:

(a) *LLL reduced* if it is proper and $\frac{3}{4}|\boldsymbol{b}_{i-1}(i-1)|^2\leqslant|\boldsymbol{b}_i(i-1)|^2$ holds for $i=2,\ldots,n$,

(b) *Hermite reduced* if it is proper and, for $i=1,\ldots,n$, $\boldsymbol{b}_i(i)$ is a shortest element of L_i,

(c) *Minkowski reduced* if, for $i=1,\ldots,n$, $\boldsymbol{b}_i$ is a shortest lattice element that can be extended to a basis with $(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1})$.

Note that Minkowski reduced bases need not be proper.

To close this section, we summarize from [15] what we shall need to know about the LLL-reduction algorithm.

Proposition 2.4. (a) *Let* $(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \subset \mathbb{R}^d$ *be an LLL reduced lattice basis. Then, for each nonzero lattice element* $\boldsymbol{v}$, $|\boldsymbol{a}_1|^2 \leqslant 2^{n-1}|\boldsymbol{v}|^2$ *holds.*

(b) *Given an integral basis* $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{Z}^d$ *consisting of vectors of length at most* $B^{1/2}$, $B \in \mathbb{R}$, $B \geqslant 2$, *the basis reduction algorithm presented in* [15] *computes an LLL reduced basis of* $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *in* $O(n^3 d \log B)$ *arithmetical operations, each performed on integers of binary length* $O(n \log B)$.

3. Kannan's algorithm to find a shortest nonzero lattice element improved

In [8], an algorithm to construct a shortest lattice element was presented. The basic idea of the algorithm is to recursively construct a Hermite reduced lattice basis. To that end, an 'almost' reduced basis is constructed that has the following property: There exists an easily computable set of 'moderate' size that contains a shortest nonzero lattice element. All elements of this set are 'tried out'.

There are essentially two differences between Kannan's original algorithm and the improved version we shall present in this section. The first one is to 'try out' only a proper subset of the set of vectors Kannan tried out. The second one is to decrease the number of the procedure's selfcalls not by just interchanging two vectors in a special situation (Step 5), but by Hermite reducing the two-dimensional lattice spanned by them. The latter idea is due to C.P. Schnorr.

The following lemma shows that, given a 'nice' basis, there are only a 'few' candidates for the shortest vector.

Lemma 3.1. *Let* $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *be a proper basis of the lattice* $L \subset \mathbb{R}^d$, *such that:*

(α) $|\boldsymbol{b}_1|^2 \leqslant \frac{4}{3}|\boldsymbol{b}_2|^2$, *and*

(β) $(\boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_n(2))$ *is Hermite reduced.*

Let $\boldsymbol{v} = \sum_{i=1}^{n} v_i \boldsymbol{b}_i$ *be a shortest nonzero element of L. Then there exists a set* $M \subset \mathbb{Z}^n$, *which satisfies:*

(γ) $(v_1, \ldots, v_n) \in M$ *or* $|\boldsymbol{v}| = |\boldsymbol{b}_1|$, *and*

(δ) $\# M \leqslant n^{0.5n+o(1)}$ *and M can be computed in* $n^{0.5n+O(1)} d$ *arithmetical operations.*

Proof. (1) Let $\boldsymbol{v} = \sum_{i=1}^{n} v_i \boldsymbol{b}_i = \sum_{i=1}^{n} v_i' \boldsymbol{b}_i(i)$, $v_i \in \mathbb{Z}$, $v_i' \in \mathbb{Q}$. Then, by Lemma 2.2(b), $v_i' = v_i + t_i$, where

$$t_i = \sum_{j=i+1}^{n} v_j \frac{\langle \boldsymbol{b}_j(i), \boldsymbol{b}_i(i) \rangle}{\langle \boldsymbol{b}_i(i), \boldsymbol{b}_j(i) \rangle}.$$

(2) If $\boldsymbol{b}_1$ is not a shortest lattice element, then $|v_i' \boldsymbol{b}_i(i)| < |\boldsymbol{b}_1|$ for $i = 1, \ldots, n$ and hence,

(i) $|v_i + t_i| < |\boldsymbol{b}_1| / |\boldsymbol{b}_i(i)|$ for all $i = 1, \ldots, n$,

since the $\boldsymbol{b}_i(i)$ are orthogonal. t_i is defined in (1).

Let

$$m = \begin{cases} \min\{i > 1 : |\boldsymbol{b}_i(i)| \geqslant |\boldsymbol{b}_1|\} & \text{if defined,} \\ n+1 & \text{otherwise.} \end{cases}$$

Since $(\boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_n(2))$ is Hermite reduced, $\boldsymbol{v}(m)=0$; otherwise, $|\boldsymbol{v}| \geqq |\boldsymbol{v}(m)| \geqq |\boldsymbol{b}_m(m)| \geqq |\boldsymbol{b}_1(1)|$ would hold. Hence,

(ii) $v_i = 0$ for all $i = m, \ldots, n$.

(3) $|b_2(2)| \leqq \sqrt{\gamma_{m-2}} \det(L(\boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_{m-1}(2)))^{1/(m-2)}$ follows from (β).

(4) $|\boldsymbol{b}_1|/|\boldsymbol{b}_2(2)| \leqq \sqrt{2}$ can easily be obtained using (α), Pythagoras' theorem, and the properness of $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

Let M be the set of all $(v_1, \ldots, v_n) \in \mathbb{Z}^n$ satisfying (i) and (ii) of (2). If $\boldsymbol{b}_1$ is not a shortest lattice element, then once having fixed $(v_{i+1}, \ldots, v_n)$ and hence t_i there are at most $2|\boldsymbol{b}_1|/|\boldsymbol{b}_i(i)|$ possibilities for v_i $(i = 1, \ldots, m-1)$. Hence, the cardinality of M is at most

$$\prod_{i=1}^{m-1} 2\frac{|\boldsymbol{b}_1|}{|\boldsymbol{b}_i(i)|} = 2^{m-1}\frac{|\boldsymbol{b}_1|^{m-2}}{\det(L(\boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_{m-1}(2)))}$$

$$\leqq 2^{m-1}|\boldsymbol{b}_1|^{m-2}\frac{\sqrt{\gamma_{m-2}}^{m-2}}{|\boldsymbol{b}_2(2)|^{m-2}} \quad \text{by (3)}$$

$$\leqq 2^{m-1}\sqrt{2}^{m-2}\sqrt{\gamma_{m-2}}^{m-2} \quad \text{by (4)}$$

$$\leqq (8\gamma_{m-2})^{n/2} \leqq n^{0.5n+o(1)}.$$

The last inequality is derived from Proposition 2.1(a), (b). The time bound can easily be derived from the construction of the set M. □

From a basis fulfilling the conditions of Lemma 3.1, a shortest lattice vector can easily be constructed by just enumerating all 'candidates' in M. So we are now able to present Procedure SHORTEST transforming an arbitrary lattice basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ into a Hermite reduced one.

Procedure SHORTEST$(n, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$

Step 1. If $n=1$, $(\boldsymbol{b}_1)$ is Hermite reduced, return, else do Steps 2 to 9.

Step 2. Replace $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ by an LLL reduced basis of $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

Step 3. $(\boldsymbol{b}'_2, \ldots, \boldsymbol{b}'_n) :=$ SHORTEST$(n-1, \boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_n(2))$.

Step 4. For $i=2$ to n replace $\boldsymbol{b}_i$ by a shortest lattice element whose projection on $\langle\langle \boldsymbol{b}_1 \rangle\rangle^{\perp}$ is $\boldsymbol{b}'_i$.

Step 5. If $|\boldsymbol{b}_1|^2 > \frac{4}{3}|\boldsymbol{b}_2|^2$, then $(\boldsymbol{b}_1, \boldsymbol{b}_2) :=$ SHORTEST$(2, \boldsymbol{b}_1, \boldsymbol{b}_2)$, goto Step 3.

Step 6. Construct a shortest lattice element $\boldsymbol{v}$ according to Lemma 3.1.

Step 7. Replace $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ by a basis containing $\boldsymbol{v}$ as first element.

Step 8. $(\boldsymbol{b}'_2, \ldots, \boldsymbol{b}'_n) :=$ SHORTEST$(n-1, \boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_n(2))$.

Step 9. For $i=1$ to n replace $\boldsymbol{b}_i$ by a shortest lattice element whose projection on $\langle\langle \boldsymbol{b}_1 \rangle\rangle^{\perp}$ is $\boldsymbol{b}'_i$.

An induction on n shows that the basis returned by Procedure SHORTEST is Hermite reduced. Lemma 3.3 will show that the algorithm always halts. Whenever Step 6 is reached, the conditions of Lemma 3.1 are fulfilled. Step 2 is performed in order to bound the number of loops through Steps 3, 4, and 5 (cf. Lemma 3.3).

Steps 4 and 9 are done by lifting the vectors $\boldsymbol{b}_i'$ as described in Section 2. Step 7 can be done in polynomial time by Procedure SELECTBASIS described in the original paper of Kannan [8]. This procedure recursively constructs a basis with $\boldsymbol{v}$ as first element by lifting a basis of the lattice obtained from projecting $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ onto $\langle\langle \boldsymbol{v} \rangle\rangle^{\perp}$.

The rest of this section is devoted to an analysis of Procedure SHORTEST. We shall assume that the input $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ consists of integral vectors, and that $|\boldsymbol{b}_i|^2 \leqslant B$ holds for some $B \in \mathbb{R}$, $B \geqslant 2$.

First, we show that the numbers involved do not increase too much.

Lemma 3.2. (a) *All numbers produced while executing* SHORTEST$(n, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *are rationals; the binary length of their numerators and denominators is at most* $O(n^2(\log n + \log B))$.

(b) *The output basis is proper, and* $|\boldsymbol{b}_i(i)|^2 \leqslant B$ *holds for* $i = 1, \ldots, n$ *while executing* SHORTEST$(n, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.

Proof. For the proof we assume the reader to be familiar with Kannan's original paper [8]. Kannan shows first that his version of SHORTEST runs on numbers of binary length $O(n^2(\log n + \log B))$, and second that all numbers produced while *not* executing the LLL-algorithm have binary length $O(n \log n + \log B)$. However, his analysis contains one small gap: When executing SELECTBASIS and when 'deprojecting' the vectors in Steps 4 and 9, (uniquely solvable) systems of linear equations with integral entries smaller than nB^{n+1} have to be solved. Here, numbers of binary length $O(n^2 \log B)$ might occur. But this does not affect the overall upper bound.

Our algorithm differs from Kannan's only in two points. First, in Step 6 we only enumerate a subset of the set of candidates he enumerated. But the construction is the same; hence, the bound on the length of the numbers involved holds.

Second, if the condition of Step 5 is fulfilled, we not only swap $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ but replace them by a Hermite reduced basis of the lattice spanned by them. However, it can be checked that Kannan's analysis still holds, especially that $\max_{i=1,\ldots,n} |\boldsymbol{b}_i(i)|$ does not increase while executing the algorithm (which proves part (b) of the lemma). A detailed analysis would result in a revision of Kannan's paper, and is therefore omitted here. □

We are now going to analyse the number of arithmetical operations performed by the algorithm. To this end, we prove the following lemma.

Lemma 3.3. SHORTEST$(n, \ldots)$ *calls* SHORTEST$(n-1, \ldots)$ *at most* $O(1) + \log n$ *times.*

Proof. Let $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be the actual basis when arriving at Step 5, let $\boldsymbol{v}$ be a shortest lattice element of $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$, and let $r = |\boldsymbol{b}_1|/|\boldsymbol{v}|$. In Step 5, $\boldsymbol{b}_1$ is replaced by $\boldsymbol{b}_1'$, a shortest element of the lattice $L(\boldsymbol{b}_1, \boldsymbol{b}_2)$. We show that

$$|\boldsymbol{b}_1'| \leqslant 2|\boldsymbol{b}_1|/\sqrt{r}. \tag{1}$$

When $r=1$, inequality (1) is trivial. Otherwise, $\boldsymbol{b}_1$ is not a shortest lattice element of $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$, and hence $|\boldsymbol{b}_2(2)| \leqslant |\boldsymbol{v}|$ since $(\boldsymbol{b}_2(2), \ldots, \boldsymbol{b}_n(2))$ is Hermite reduced. So,

$$\det(L(\boldsymbol{b}_1, \boldsymbol{b}_2)) = |\boldsymbol{b}_1||\boldsymbol{b}_2(2)| \leqslant |\boldsymbol{b}_1||\boldsymbol{v}|$$

holds and (1) follows from

$$|\boldsymbol{b}_1'| \leqslant \sqrt{\gamma_2}\sqrt{|\boldsymbol{b}_1||\boldsymbol{v}|} \leqslant 2|\boldsymbol{b}_1|/\sqrt{|\boldsymbol{b}_1|/|\boldsymbol{v}|}.$$

By Proposition 2.4(a), $|\boldsymbol{b}_1|/|\boldsymbol{v}| \leqslant 2^{(n-1)/2}$ holds after executing Step 2. So, after passing Steps 3, 4, and 5 at most $\log(\frac{1}{2}(n-1))$ times, $|\boldsymbol{b}_1|/|\boldsymbol{v}| \leqslant 8$ holds by (1). Each further passing decreases $|\boldsymbol{b}_1|^2$ by (more than) a factor $\frac{4}{3}$, hence the claim of the lemma follows. □

The results of the runtime analysis are summarized in the following proposition.

Proposition 3.4. *Given vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{Z}^d$ of length at most $B^{1/2}$, $B \in \mathbb{R}$, $B \geqslant 2$,* SHORTEST$(n, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *performs $d(n^{0.5n+O(1)} + n^{o(n)} \log B)$ arithmetical operations on integral numbers with binary length $O(n^2(\log n + \log B))$.*

Proof. The number of arithmetical operations performed by Steps 1, 4, 7, and 9 can be bounded by $p_1(n)d$, where p_1 is a polynomial (for the analysis of Step 7, we refer to [8]), and by Proposition 2.4(b), Step 2 takes at most $O(n^3 d \log B)$ operations. For $n=2$, by the properties of LLL reduced bases, the condition in Step 5 is never fulfilled, so Step 5 takes $O(d \log B)$ operations. By Lemma 2.2(c) and since $\max_{i=1,\ldots,n}\{|\boldsymbol{b}_i(i)|\}$ does not increase while executing Procedure SHORTEST [8], we may assume that all selfcalls of SHORTEST are applied to integral vectors with components of binary length $O(n \log B)$.

Let $T(n, d)$ denote the number of arithmetical operations performed by SHORTEST$(n, \ldots)$. Then, for some polynomial p,

$$T(n, d) \leqslant (O(1) + \log n)(T(n-1, d) + dp(n) \log B) + n^{0.5n+O(1)}d,$$

and thus

$$T(n, d) \leqslant d(n^{0.5n+O(1)} + n^{o(n)} \log B)$$

holds. □

4. Kannan's algorithm to solve the closest lattice point problem improved

The *closest lattice point problem* is the following: Given a lattice $\boldsymbol{L} = L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{R}^d$ and a vector $\boldsymbol{a} \in \mathbb{R}^d$, find a lattice element $\boldsymbol{v} = \sum_{i=1}^n v_i \boldsymbol{b}_i$ so that $|\boldsymbol{a} - \boldsymbol{v}| \leqslant |\boldsymbol{a} - \boldsymbol{w}|$ for all $\boldsymbol{w} \in L$.

Kannan [8] presented an algorithm to solve this problem. He showed with similar techniques as in Procedure SHORTEST that for some particular $i \in \{1, \ldots, n\}$ there

exists a subset of $\mathbb{Z}^{n-i+1}$ of cardinality at most $n^{3(n-i+1)/2}$ that contains $(v_1, \ldots, v_n)$, if the basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is Hermite reduced.

Now if $\bar{\boldsymbol{v}}$ solves the closest lattice point problem for the lattice $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1})$ and the vector $\boldsymbol{a} - \sum_{j=i}^{n} v_j \boldsymbol{b}_j$, then $\bar{\boldsymbol{v}} + \sum_{j=i}^{n} v_j \boldsymbol{b}_j$ solves the problem for $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ and $\boldsymbol{a}$. So the original problem is reduced to $n^{3(n-i+1)/2}$ $(i-1)$-dimensional subproblems.

We improve Kannan's algorithm by a simple observation. If $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is Hermite reduced, so is $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{i-1})$. Thus, the basis needs to be reduced only once, and not again for each iteration as done in Kannan's algorithm. This observation yields a new closest lattice point algorithm.

Proposition 4.1. *Let* $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{Z}^d$ *be a basis of the lattice* L, $\boldsymbol{a} \in \mathbb{Z}^d$, $|\boldsymbol{a}|^2 \leqslant B$, $|\boldsymbol{b}_i|^2 \leqslant B$ *for* $i = 1, \ldots, n$ *and* $B \in \mathbb{R}$, $B \geqslant 2$. *Then the closest lattice point problem for* L *and* $\boldsymbol{a}$ *can be solved in* $d(n^{1.5n+O(1)} + n^{o(n)} \log B)$ *arithmetical operations. All numbers involved are rationals, the binary length of their numerators and denominators is* $O(n^2(\log n + \log B))$.

If $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *is Hermite reduced, the problem can be solved in* $n^{1.5n+O(1)} d$ *arithmetical operations on integers of binary length* $O(n \log B)$.

Proof. Let firstly $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ be Hermite reduced and let $S(n, d)$ be the number of arithmetical operations performed by the improved algorithm. Since the original problem is reduced to $n^{3(n-i+1)/2}$ $(i-1)$-dimensional subproblems,

$$S(n, d) \leqslant n^{3(n-i+1)/2}(S(i-1, d) + dq(n))$$

holds. Here, $q(n)$ is a polynomial that counts the number of arithmetical operations performed while computing the inputs of the lower dimensional subproblems, testing 'candidates' etc. Hence, the claim on the number of arithmetical operations follows.

We do not prove details here, since we did not present the details of the improved algorithm. For the same reason, the claim on the binary length of the numbers is not proved here. The interested reader is referred to [5]; in [8], worse bounds are mentioned.

The performance analysis for not Hermite reduced bases can immediately be derived from Proposition 3.4, and the analysis of the performance for Hermite reduced bases can be derived since the input basis needs to be reduced only once. Lemma 3.2(b) guarantees that reducing a basis increases the lengths of its elements at most by a factor n. □

5. The algorithm to construct Minkowski reduced lattice bases

Let $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_n)$ be a Minkowski reduced basis of the lattice $L \subset \mathbb{R}^d$. In this section, we present an algorithm that constructs $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_n)$ from an arbitrary given basis of L.

The first vector $\boldsymbol{m}_1$ will be constructed using Procedure SHORTEST. Now, if $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n)$ is a basis of L, then certainly

$$|\boldsymbol{m}_{k+1}(k+1)| \leqq |\boldsymbol{m}_{k+1}| \leqq |\boldsymbol{b}_{k+1}|.$$

Lemma 5.1 will show that then, if the basis is 'nice' there are only 'few' possibilities for $\boldsymbol{m}_{k+1}(k+1) \in L_{k+1}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n)$.

If $\boldsymbol{m}_{k+1}(k+1)$ is given, $\boldsymbol{m}_{k+1}$ can be found: it has to be the shortest vector in $\boldsymbol{L}$ projecting on $\boldsymbol{m}_{k+1}(k+1)$. So, by 'deprojecting' all 'possible' $\boldsymbol{m}_{k+1}(k+1)$, we can find $\boldsymbol{m}_{k+1}$.

Lemma 5.1. *Let* $0 \leqq k \leqq n-1$ *and* $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ *be a basis of the lattice* $L \subset \mathbb{R}^d$, *such that*:

(α) $\exists \boldsymbol{m}_{k+1}, \ldots, \boldsymbol{m}_n : (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k, \boldsymbol{m}_{k+1}, \ldots, \boldsymbol{m}_n)$ *is a Minkowski reduced basis of* $\boldsymbol{L}$,

(β) $(\boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_n(k+1))$ *is Hermite reduced,*

(γ) $|\boldsymbol{b}_{k+1}| = \min\{|\boldsymbol{w}| : \boldsymbol{w} \in L, \boldsymbol{w}(k+1) = \boldsymbol{b}_{k+1}(k+1)\}$.

Let $\boldsymbol{m}_{k+1} = \sum_{i=1}^{n} v_i \boldsymbol{b}_i$, $v_i \in \mathbb{Z}$. *Then there exists a set* $M \subset \mathbb{Z}^{n-k}$ *which satisfies*:

(δ) $(v_{k+1}, \ldots, v_n) \in M$ *or* $|\boldsymbol{m}_{k+1}| = |\boldsymbol{b}_{k+1}|$, *and*

(ε) $\# M \leqq (\frac{5}{4})^{n^3/(4-o(1))}$ *and* M *can be computed in* $d(\frac{5}{4})^{n^3/(4-o(1))}$ *arithmetical operations.*

Proof. (1) Let $(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_r)$ be a Minkowski reduced basis of the lattice $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_r)$. Then [20],

$$|\boldsymbol{a}_1| \cdots |\boldsymbol{a}_r| \leqq \rho_r \sqrt{\gamma_r}^r \det(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_r)),$$

$$\text{where } \rho_n = \begin{cases} (\frac{5}{4})^{(n-3)(n-4)/4} & \text{for } n > 4, \\ 1 & \text{for } n \leqq 4. \end{cases}$$

(2) Since $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1})$ is a Minkowski reduced basis of $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1})$ and $\det(L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{k+1})) = |\boldsymbol{b}_1(1)| \cdots |\boldsymbol{b}_{k+1}(k+1)|$, (1) implies

$$\frac{|\boldsymbol{b}_{k+1}|}{|\boldsymbol{b}_{k+1}(k+1)|} \leqq \rho_k \sqrt{\gamma_k}^k \leqq \rho_n \sqrt{\gamma_k}^n.$$

(3) Let $\boldsymbol{m}_{k+1} = \sum_{i=1}^{n} v_i \boldsymbol{b}_i = \sum_{i=1}^{n} v_i' \boldsymbol{b}_i$, $v_i \in \mathbb{Z}$, $v_i' \in \mathbb{R}$. Then, by Lemma 2.2(b), $\boldsymbol{v_i' =}$ $v_i + t_i$, where

$$t_i = \sum_{j=i+1}^{n} v_j \frac{\langle \boldsymbol{b}_j(i), \boldsymbol{b}_i(i) \rangle}{\langle \boldsymbol{b}_i(i), \boldsymbol{b}_j(i) \rangle}.$$

(4) (β) implies

$$|\boldsymbol{b}_{k+1}(k+1)|^{m-k-1} \leqq \sqrt{\gamma_{m-k-1}}^{m-k-1} \det(L(\boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_{m-1}(k+1))).$$

(5) If $|\boldsymbol{m}_{k+1}| < |\boldsymbol{b}_{k+1}|$, then $|v_i' \boldsymbol{b}_i(i)| < |\boldsymbol{b}_{k+1}|$, and hence

(i) $|v_i + t_i| < |\boldsymbol{b}_{k+1}| / |\boldsymbol{b}_i(i)|$ for all $i = 1, \ldots, n$,

where t_i is defined in (3).

Let

$$m = \begin{cases} \min\{j \geqq k : |\boldsymbol{b}_j(j)| \geqq |\boldsymbol{b}_{k+1}|\} & \text{if defined,} \\ n+1 & \text{otherwise.} \end{cases}$$

Then $\boldsymbol{m}_{k+1}(m) = 0$, and hence

(ii) $v_i = 0$ for all $i = m, \ldots, n$,

holds, since $(\boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_n(k+1))$ is Hermite reduced.

Let M be the set of all $(v_{k+1}, \ldots, v_n) \in \mathbb{Z}^{n-k}$ satisfying (i) and (ii) of (5). If $|\boldsymbol{m}_{k+1}| < |\boldsymbol{b}_{k+1}|$, then once having fixed $(v_{i+1}, \ldots, v_n)$, there are at most $2|\boldsymbol{b}_{k+1}|/|\boldsymbol{b}_i(i)|$ possibilities for v_i $(i = k+1, \ldots, m-1)$. Hence, the cardinality of M is at most

$$\prod_{i=k+1}^{m-1} 2\frac{|\boldsymbol{b}_{k+1}|}{|\boldsymbol{b}_i(i)|} = 2^{m-1-k} \frac{|\boldsymbol{b}_{k+1}|^{m-k-1}}{\det(L(\boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_{m-1}(k+1)))}$$

$$\leqq 2^n \sqrt{\gamma_{m-k-1}}^{m-k-1} \left(\frac{|\boldsymbol{b}_{k+1}|}{|\boldsymbol{b}_{k+1}(k+1)|} \right)^{m-k-1} \quad \text{by (4)}$$

$$\leqq \rho_n^n \sqrt{n}^{n^2+n} \leqq \left(\tfrac{5}{4}\right)^{n^3/(4-o(1))} \quad \text{by (2).}$$

The time bound of $d\#M$ can be easily derived from the construction of M. □

'Deprojecting' the vectors in L_{k+1} defined by the tuples $(v_{k+1}, \ldots, v_n) \in M$ with $\gcd\{v_{k+1}, \ldots, v_n\} = 1$ produces lattice elements extendable to a basis together with $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k)$. However, it should be remarked that 'deprojecting' the vector $\boldsymbol{v}(k+1) = \sum_{i=k+1}^{n} v_i \boldsymbol{b}_i(k+1)$ cannot be done by lifting it to L_k, L_{k-1} and so on. This need not lead to the shortest vector in L projecting on $\boldsymbol{v}(k+1)$. Correctly 'deprojecting' $\boldsymbol{v}(k+1)$ is achieved by constructing

$$\boldsymbol{v} = \mathrm{CLP}\left(\sum_{i=k+1}^{n} v_i \boldsymbol{b}_i(k+1) - \sum_{i=k+1}^{n} v_i \boldsymbol{b}_i, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \right) + \sum_{i=k+1}^{n} v_i \boldsymbol{b}_i$$

(here $\mathrm{CLP}(\boldsymbol{a}, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_r)$ solves the closest lattice point problem for $\boldsymbol{a}$ and $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_r)$ according to Section 4).

So, the following procedure constructs $\boldsymbol{m}_{k+1}$ given a 'nice' lattice basis.

Procedure FINDMIN$(k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$

Input: $k \in \{0, \ldots, n-1\}$,

$(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{R}^d$, that fulfill the conditions of Lemma 5.1,

$(\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$ Hermite reduced basis of the lattice $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$.

Output: $\boldsymbol{m}_{k+1}$ so that $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k, \boldsymbol{m}_{k+1})$ is extendable to a Minkowski reduced lattice basis of $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$,

Procedure

Step 1. Enumerate $M_1 := \{(v_{k+1}, \ldots, v_n) \in M : \gcd\{v_{k+1}, \ldots, v_n\} = 1\}$ (M is the set described in Lemma 5.1).

Step 2. For each $(v_{k+1}, \ldots, v_n) \in M_1$ compute

$$\boldsymbol{v}' := \mathrm{CLP}\left(\sum_{j=k+1}^{n} v_j(\boldsymbol{b}_j(k+1) - \boldsymbol{b}_j), \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k \right) + \sum_{j=k+1}^{n} v_j \boldsymbol{b}_j.$$

Step 3. Let $\boldsymbol{v}$ be the shortest of the vectors $\boldsymbol{v}'$ constructed in Step 2.
Step 4. If $M_1 = \emptyset$ or $|\boldsymbol{b}_{k+1}| \leqslant |\boldsymbol{v}|$, then $\boldsymbol{m}_{k+1} := \boldsymbol{b}_{k+1}$ else $\boldsymbol{m}_{k+1} := \boldsymbol{v}$.

Before defining the algorithm to construct Minkowski reduced lattice bases, we still need one auxiliary procedure that, once having found $\boldsymbol{m}_{k+1}$, inserts it in a given basis.

Procedure INSERTBASIS($k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{v}$)
Input: $k \in \{0, \ldots, n-1\}$,
$(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ basis of the lattice $L \subset \mathbb{R}^d$,
$\boldsymbol{v} \in L$, so that $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k, \boldsymbol{v})$ can be extended to a basis of L.
Output: a basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k, \boldsymbol{v}, \boldsymbol{b}'_{k+2}, \ldots, \boldsymbol{b}'_n)$ of L.
Procedure
Step 1. $(\boldsymbol{v}(k+1), \boldsymbol{a}_{k+2}(k+1), \ldots, \boldsymbol{a}_n(k+1)) := \text{SELECTBASIS}(\boldsymbol{v}(k+1), \boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_n(k+1))$;
Step 2. for $j := k+1$ to n do compute $\boldsymbol{b}'_j \in L$ that fulfills $\boldsymbol{b}'_j(k+1) = \boldsymbol{a}_j(k+1)$.

Procedure SELECTBASIS described in [8] constructs a basis containing some given primitive element $\boldsymbol{v}$ as first vector. So, INSERTBASIS works correctly.

We are now able to define the procedure M-RED that constructs a Minkowski reduced lattice basis from an arbitrary given one. As indicated before, $\boldsymbol{m}_1$ will be found using Procedure SHORTEST, and then $\boldsymbol{m}_2, \ldots, \boldsymbol{m}_n$ will be constructed successively. Fig. 2 demonstrates what the system of projected bases shown in Fig. 1 looks like before constructing $\boldsymbol{m}_{k+1}$.

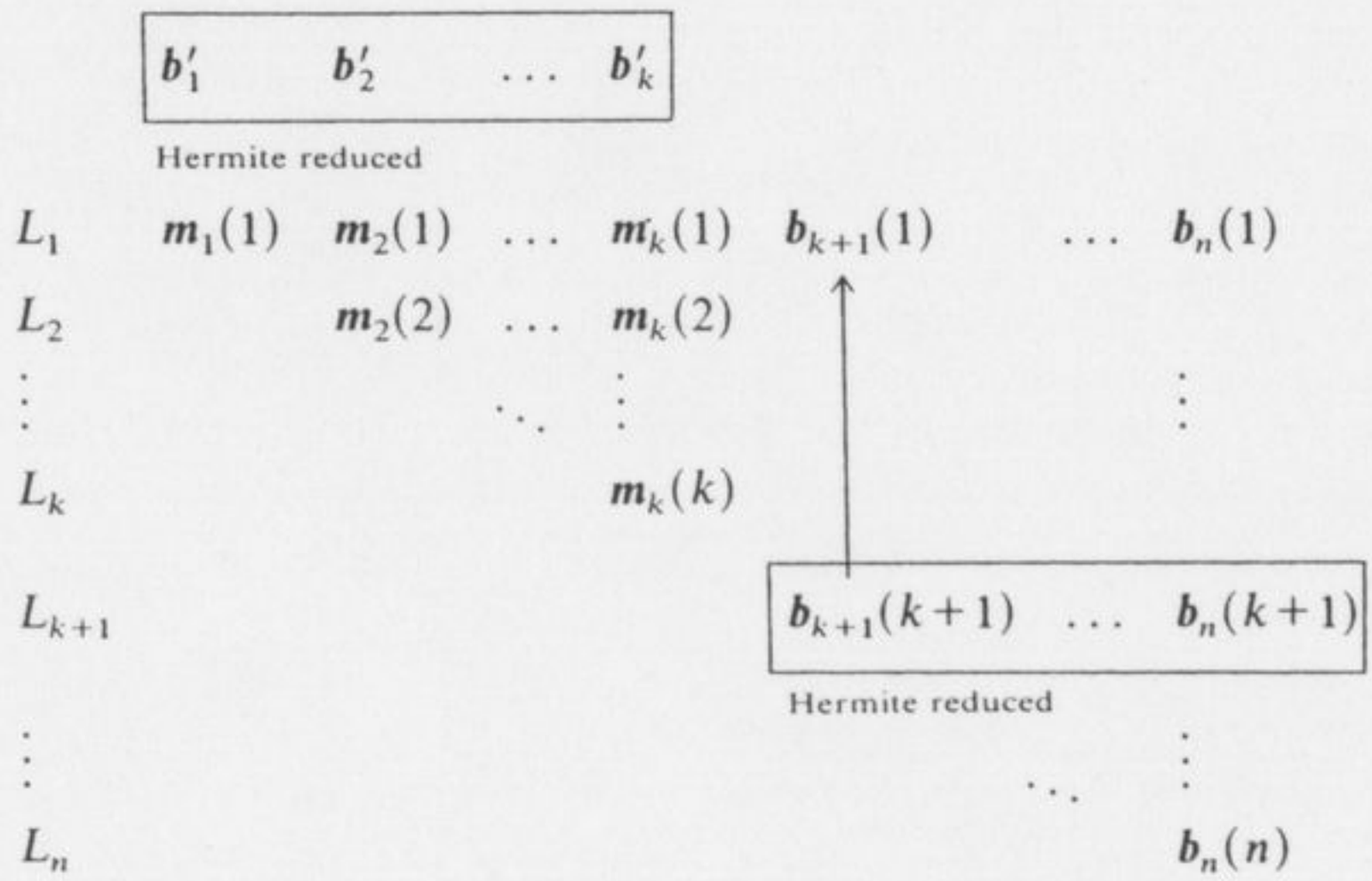

Fig. 2. Projections of the basis before constructing $\boldsymbol{m}_{k+1}$.

Procedure M-RED$(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$
Input: $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset \mathbb{R}^d$ basis of the lattice L.
Output: $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_n)$ Minkowski reduced basis of L.
Procedure
Step 1. $(\boldsymbol{m}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n) := \text{SHORTEST}(n, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$.
Step 2. $\boldsymbol{b}'_1 := \boldsymbol{m}_1$.
For $k := 1$ to $n-1$ do Steps 3 to 8.
Step 3. $\boldsymbol{m}_{k+1} := \text{FINDMIN}(k, \boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n, \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$.
Step 4. $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{k+1}, \boldsymbol{b}_{k+2}, \ldots, \boldsymbol{b}_n) := \text{INSERTBASIS}(k, \boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n, \boldsymbol{m}_{k+1})$.
Step 5. $(\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_{k+1}) := \text{SHORTEST}(k+1, \boldsymbol{m}_1, \ldots, \boldsymbol{m}_{k+1})$.
Step 6. $(\boldsymbol{a}_{k+2}(k+2), \ldots, \boldsymbol{a}_n(k+2)) := \text{SHORTEST}(n-k-1, \boldsymbol{b}_{k+2}(k+2)), \ldots, \boldsymbol{b}_n(k+2))$.
Step 7. For $j := k+2$ to n do compute $\boldsymbol{b}_j \in L$ that fulfills $\boldsymbol{b}_j(k+2) = \boldsymbol{a}_j(k+2)$.
Step 8. $\boldsymbol{b}_{k+2} := \text{CLP}(\boldsymbol{b}_{k+2}(k+2) - \boldsymbol{b}_{k+2}, \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_{k+1}) + \boldsymbol{b}_{k+2}$.

An induction on k shows that before looping through Steps 3–8 for the kth time (i) to (v) hold:

(i) $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n)$ is a basis of L.
(ii) $(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k)$ are the first k elements of a Minkowski reduced basis of L.
(iii) $(\boldsymbol{b}_{k+1}(k+1), \ldots, \boldsymbol{b}_n(k+1))$ is a Hermite reduced basis of the lattice $L_{k+1}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n)$.
(iv) $|\boldsymbol{b}_{k+1}| = \min\{|\boldsymbol{w}|: \boldsymbol{w} \in L,\ \boldsymbol{w}(k+1) = \boldsymbol{b}_{k+1}(k+1)\}$,
(v) $(\boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$ is a Hermite reduced basis of $L(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k)$.

This shows that the procedures called by M-RED are applied to admissible inputs and hence yield the correctness of the algorithm.

In order to analyse the algorithm M-RED, we first analyse Procedures INSERTBASIS and FINDMIN.

Lemma 5.2. *Let* $(k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$ (*respectively* $(k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{v})$) *be admissible integral inputs for Procedures* FINDMIN (*respectively* INSERTBASIS) *consisting of vectors of length at most* $B^{1/2}$.

(a) FINDMIN$(k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{b}'_1, \ldots, \boldsymbol{b}'_k)$ *carries out at most* $((\frac{5}{4})^{n^3/(4-o(1))})d \log B$ *arithmetical operations on integers of binary length* $O(n^2 \log B)$.
(b) INSERTBASIS$(k, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n, \boldsymbol{v})$ *carries out* $p(n)d \log B$ *arithmetical operations on integers of binary length* $O(n^3 \log B)$, *where* p *is some polynomial.*

Proof. (a) Let $(v_{k+1}, \ldots, v_n) \in M$ and $(v'_{k+1}, \ldots, v'_n)$ be the corresponding tuple defined by $\sum_{i=k+1}^n v_i \boldsymbol{b}_i(k+1) = \sum_{i=k+1}^n v'_i \boldsymbol{b}_i(i)$. By Lemma 2.2(c), $|v'_i| \leqslant |\boldsymbol{b}_{k+1}|/|\boldsymbol{b}_i(i)| \leqslant B^{n+1}$ for all $i = k+1, \ldots, n$, and an induction yields $|\boldsymbol{v}_i| \leqslant 2^n B^{n+1}$ for all i. So, the vectors CLP is applied to in Step 8 are shorter than $(2B)^{n+1}$. Hence, by Lemma 2.2(c), we may assume that the inputs of CLP are integral vectors each shorter than $2^{n+1} B^{2n+1}$. Now the claim on the size of numbers follows from Proposition 4.1.

The set M can be enumerated in $\#Md$ arithmetical operations, and $\#M \leqslant (\frac{5}{4})^{n^3/(4-o(1))}$ by Lemma 5.1. The components of each element $(v_{k+1}, \ldots, v_n)$ are bound in absolute value by $(2B)^{n+1}$. For each element, $\gcd\{v_{k+1}, \ldots, v_n\}$ is evaluated in $n^{O(1)} \log B$ arithmetical operations, and one application of CLP is performed in $n^{1.5n+O(1)}d$ operations. Hence, the claim on the number of arithmetical operations follows.

(b) We may assume that lifting and SELECTBASIS are applied to integral vectors of length at most nB^{n+1}, which can always be achieved by multiplying the vectors involved by the least common multiple of the denominators of their coefficients. Now, (b) can readily be shown. □

We are now going to finish the analysis of Procedure M-RED. Again, let $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{Z}^d$ be shorter than $B^{1/2}$.

The reader may show, by induction on k, that all vectors $\boldsymbol{m}_i$, $\boldsymbol{b}'_i$, and $\boldsymbol{b}_i$ produced before looping through Steps 3–8 the kth time are shorter than $n^{2k+1}B^{1/2}$, except while executing subroutines.

All occurring nonintegral numbers not involved in subroutines are coefficients of an element of some $L_{k+1}(\boldsymbol{m}_1, \ldots, \boldsymbol{m}_k, \boldsymbol{b}_{k+1}, \ldots, \boldsymbol{b}_n)$. Hence, by the above and Lemma 2.2(c) their denominators are smaller than $(n^{2k+1}B^{1/2})^n$. So we may assume that procedures are applied only to integral arguments smaller than $(n^{2k+1}B^{1/2})^{n+1}$. Now, Propositions 3.4 and 4.1, and Lemma 5.2 show, that the binary length of all numbers involved in M-RED$(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ is $O(n^4(n \log n + \log B))$.

It can easily be shown that M-RED performs $((\frac{5}{4})^{n^3/(4-o(1))})d \log B$ arithmetical operations. Hence, we have proved the following proposition.

Proposition 5.3. *For fixed dimension n,* M-RED *constructs a Minkowski reduced lattice basis from an arbitrary given basis in polynomial time. As a function of n, the runtime is exponential.*

On input vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{Z}^d$ of Euclidean length at most $B^{1/2}$, $B \in \mathbb{R}$, $B \geqslant 2$, M-RED *performs $((\frac{5}{4})^{n^3(4-o(1))})d \log B$ arithmetical operations on integers of binary length $O(n^4(n \log n + \log B))$.*

Acknowledgment

The results presented in this paper are part of the author's Ph.D. Thesis guided by Prof. Dr. C.P. Schnorr. I would like to thank him for many helpful discussions.

References

[1] L. Babai, On Lovász' lattice reduction and the nearest lattice point problem, *Symp. on Theoretical Aspects of Computer Science* (1985) 13–20.

[2] J.W.S. Cassels, *An Introduction to the Geometry of Numbers* (Springer, Berlin/Heidelberg, 1959).

[3] J.W.S. Cassels, *Rational Quadratic Forms* (Academic Press, New York, 1978).
[4] C.F. Gauss, *Disquisitiones Arithmeticae* (Springer, Berlin, 1889) (German translation).
[5] B. Helfrich, Reduktionsalgorithmen für Gitterbasen, Master Thesis, FB Mathematik, Univ. Frankfurt am Main, 1984.
[6] C.H. Hermite, Deuxième lettre à Jacobi, *Oeuvres de Hermite I* (Gauthier-Villary, Paris, 1905) 122–135.
[7] R. Kannan, A.K. Lenstra and L. Lovász, Polynomial factorisation and nonrandomness of bits of algebraic and some transcendental numbers, Tech. Rept., Computer Science Dept., Carnegie-Mellon Univ., 1984.
[8] R. Kannan, Improved algorithms for integer programming and related problems, *15th ACM Symp. on Theory of Computing* (1983) 193–206.
[9] R. Kannan, Lattices, basis reduction and the shortest vector problem, in: *Theory of Algebra*, Colloquia Mathematica Societatis Janos Bolyai **44** (North-Holland, Amsterdam, 1986).
[10] A. Korkine and G. Zolotareff, Sur les formes quadratiques, *Math. Ann.* **6** (1873) 366–389.
[11] J.C. Lagarias, Worst-case complexity bounds for algorithms in the theory of integral quadratic forms, *J. Algorithms* **1** (1980) 142–186.
[12] J.C. Lagarias, Computational complexity of simultaneous diophantine approximation problems, *23rd Symp. on the Foundations of Computer Science* (1982) 32–39.
[13] J.C. Lagarias and A.M. Odlyzko, Solving low density subset sum problems, *24th Ann. Symp. on the Foundations of Computer Science* (1983) 1–10.
[14] C.G. Lekkerkerker, *Geometry of Numbers* (North-Holland, Amsterdam, 1969).
[15] A.K. Lenstra, H.W. Lenstra, Jr. and L. Lovász, Factoring polynomials with rational coefficients, *Math. Ann.* **261** (1982) 513–534.
[16] H. Minkowski, Über die positiven quadratischen Formen und über kettenbruchähnliche Algorithmen, *J. Reine Angew. Math.* **107** (1891) 278–297.
[17] C.P. Schnorr, A hierarchy of polynomial time basis reduction algorithms, in: *Theory of Algebra*, Colloquia Mathematica Societatis Janos Bolyai **44** (North-Holland, Amsterdam, 1986).
[18] A. Schönhage, Factorization of univariate integer polynomials by Diophantine approximation and by an improved basis reduction algorithm, Preprint, Tübingen, 1983.
[19] P. van Emde Boas, Another NP-complete partition problem and the complexity of computing short vectors in a lattice, Rept. 81-04, Dept. of Mathematics, Univ. of Amsterdam, 1981.
[20] B.L. van der Waerden and H. Gross, *Studien zur Theorie der Quadratischen Formen* (Birkhäuser, Basel, 1968).