This article will appear in: *Surveys in algorithmic number theory*, Mathematical Sciences Research Institute Publications, Cambridge University Press version 20070327

Lattices

H. W. Lenstra jr.

1. Introduction

A *lattice* is a discrete subgroup of a Euclidean vector space, and *geometry of numbers* is the theory that occupies itself with lattices. Since the publication of Hermann Minkowski's *Geometrie der Zahlen* in 1896, lattices have become a standard tool in number theory, especially in the areas of diophantine approximation, algebraic number theory, and the arithmetic theory of quadratic forms.

The theory of continued fractions, largely developed by Leonhard Euler (1707–1783), is in substance concerned with algorithmic aspects of lattices of rank 2. A significant advance in the algorithmic theory of lattices of general rank occurred in the early 1980's, with the development of the powerful lattice basis reduction algorithm that came to be called the LLL algorithm [21]. The LLL algorithm has found numerous applications in both pure and applied mathematics.

In algorithmic number theory, geometry of numbers now plays a role that is comparable to the role that linear programming plays in optimization theory, and that linear algebra plays throughout mathematics. This is due to a similar combination of circumstances: good algorithms are available for solving the basic problems, and many commonly encountered problems reduce to those basic problems. Just as a multitude of problems in mathematics can be linearized, so can many others be addressed by the introduction of a suitable lattice. Typically, this applies to problems that have both a discrete and a continuous component, such as the search for a system of integers that satisfies certain inequalities. Algorithmic number theory abounds in such problems.

The main purpose of the present introduction to the subject is to impart to the reader the ability to recognize situations in which a lattice basis reduction algorithm is useful. For this reason, all definitions and algorithms have been formulated in conceptual terms, appealing to the geometric rather than the algebraic intuition. At the same time, coordinates will be chosen when they have an actual role to play, which is unavoidably the case whenever the algorithms are to be translated into genuine computer programs.

A generous sample of applications of lattice basis reduction to algorithmic number theory has been included; in many cases, the main point consists of recognizing a lattice behind a problem. For applications to integer programming, one may consult [1].

Complete proofs have not been provided for all results mentioned, though in many cases one will find a sketch of a proof or a 'convincing argument'. Generally, the subject matter is elementary enough that the readers can supply the details themselves, and in any case they can turn to the references at the end. The same applies to running time estimates of algorithms.

2. Lattices

Euclidean vector spaces. A *Euclidean vector space* is a finite-dimensional vector space E over the field $\mathbf{R}$ of real numbers equipped with a map $\langle\ ,\ \rangle\colon E \times E \to \mathbf{R}$ satisfying

$$\langle w + x, y\rangle = \langle w, y\rangle + \langle x, y\rangle, \qquad \langle rx, y\rangle = r\langle x, y\rangle,$$
$$\langle x, y\rangle = \langle y, x\rangle, \qquad \langle z, z\rangle > 0$$

for all $r \in \mathbf{R}$ and w, x, y, $z \in E$, $z \neq 0$. The map $\langle\ ,\ \rangle$ is called the *inner product* on E. For $z \in E$, we write $\|z\| = \langle z, z\rangle^{1/2}$, and we refer to this number as the *length* of the vector z. Any Euclidean vector space E is a *metric space* with distance function $d\colon E \times E \to \mathbf{R}$ defined by $d(x, y) = \|x - y\|$. If E, E' are Euclidean vector spaces, then a map $\psi\colon E \to E'$ is an *isomorphism of Euclidean vector spaces* if it is an isomorphism of vector spaces over $\mathbf{R}$ that preserves inner products, in the sense that for all x, $y \in E$ one has $\langle\psi(x), \psi(y)\rangle = \langle x, y\rangle$. For each non-negative integer n, the vector space $\mathbf{R}^n$ is a Euclidean vector space with the *standard inner product* defined by $\langle (x_i)_{i=1}^n, (y_i)_{i=1}^n\rangle = \sum_{i=1}^n x_i y_i$. For each Euclidean vector space E there is an isomorphism $\mathbf{R}^{\dim E} \cong E$ of Euclidean vector spaces, where $\dim E$ denotes the dimension of E as a vector space over $\mathbf{R}$.

Lattices. A subset L of a Euclidean vector space E is *discrete* if the metric on E defines the discrete topology on L; in other words, if for each $x \in L$ there is a positive real number ε such that the only $y \in L$ with $d(x, y) < \varepsilon$ is given by $y = x$. A *lattice* is an additive subgroup L of a Euclidean vector space E such that L is discrete as a subset of E; given that L is a subgroup, discreteness is equivalent to the existence of a positive real number ε such that the only vector $y \in L$ with $\|y\| < \varepsilon$ is given by $y = 0$.

A subset L of a Euclidean vector space E is a lattice if and only if there are $\mathbf{R}$-linearly

independent vectors $b_1, \ldots, b_n \in E$ such that

$$L = \sum_{i=1}^{n} \mathbf{Z} b_i = \Big\{ \sum_{i=1}^{n} c_i b_i : c_i \in \mathbf{Z} \text{ for } i = 1, \ldots, n \Big\}.$$

If this is the case, then $b_1, \ldots, b_n$ are said to form a *basis* for L (over $\mathbf{Z}$), and L is isomorphic to $\mathbf{Z}^n$ as an abelian group; from $\#L/2L = 2^n$ one sees that n is determined by the structure of L as an abelian group, and it is called the *rank* of L, notation: $\operatorname{rk} L$.

One can also define lattices without reference to a Euclidean vector space. Namely, let L be an abelian group, and let $q\colon L \to \mathbf{R}$ be a map. Then L can be embedded as a lattice in a Euclidean vector space E with $q(x) = \|x\|^2$ for all $x \in L$ if and only if L is finitely generated and the following three conditions are satisfied:

$$q(x+y) + q(x-y) = 2q(x) + 2q(y) \text{ for all } x, y \in L,$$
$$q(x) \neq 0 \text{ for all } x \in L,\ x \neq 0,$$
$$\{x \in L : q(x) \leq r\} \text{ is finite for each real number } r.$$

The proof of the 'if'-part (see [23, Prop. 4.1]) shows that one may take $E = L \otimes_{\mathbf{Z}} \mathbf{R}$, the inner product being such that $\langle x, y \rangle = \big(q(x+y) - q(x) - q(y)\big)/2$ for all x, $y \in L$. Thus, one can define a lattice to be a finitely generated abelian group L equipped with a map $q\colon L \to \mathbf{R}$ satisfying the three conditions just listed. The first of these properties is called the *parallelogram law*, since it expresses that the sum of the squares of the lengths of the two diagonals of a parallelogram equals the sum of the squares of the lengths of its four sides. In general, if L, q constitute a lattice, then one has $q(x) \geq 0$ for each $x \in L$, one thinks of $q(x)$ as the square of the length of x, and the function $d\colon L \times L \to \mathbf{R}$ defined by $d(x, y) = q(x-y)^{1/2}$ is a metric on L.

We shall often refer to a lattice as a pair L, q, emphasizing that all we need to know is the group L and the lengths of all of its elements; when q is clear from the context, it may be dropped. Often, it will tacitly be assumed that such a lattice is embedded in a Euclidean vector space E, and then it is always understood that $q(x) = \|x\|^2 = \langle x, x \rangle$ for all $x \in L$. The notation $q(x) = \langle x, x \rangle$ will also be used for other elements x of E. Sometimes it is understood that L is of *full rank* in E, which means that one has $\operatorname{rk} L = \dim E$; one can always achieve this by replacing E by the subspace of E spanned by L.

Isometries. An isometry of a lattice L, q to a lattice L', q' is a bijection $f\colon L \to L'$ that preserves distances. One can compose each isometry with a translation to achieve that it

maps 0 to 0, and each isometry mapping 0 to 0 is automatically a group isomorphism. One cares about lattices only up to isometry.

Sublattices. Let L, q be a lattice. Every subgroup M of L becomes a lattice upon restricting q to M; such a lattice is called a *sublattice* of L. A sublattice M of L is called *pure* if L/M is *torsion-free* as an abelian group, which means that L/M has no non-zero element of finite order. If M is a pure sublattice of L, then $N = L/M$ acquires a natural lattice structure in the following way: embed L in a Euclidean vector space E, let E' be the subspace spanned by M, write $E'^{\perp}$ for the orthogonal complement $\{x \in E : \langle x, y\rangle = 0 \text{ for all } y \in E'\}$ of E' in E, and $\pi\colon E \to E'^{\perp}$ for the orthogonal projection (so π is $\mathbf{R}$-linear, zero on E', and the identity on $E'^{\perp}$, and π is uniquely determined by those properties); then πL is a discrete subgroup of the Euclidean vector space $E'^{\perp}$ and therefore a lattice, and the natural isomorphism $N = L/M \to \pi L$ induced by π identifies πL with N, which therefore becomes a lattice as well.

The dual lattice. Let L be a lattice of full rank in a Euclidean vector space E. Then $L^{\dagger} = \{x \in E : \langle x, L\rangle \subset \mathbf{Z}\}$ is also a lattice of full rank in E, the *dual* (or *polar*) of L. If b_1, $\ldots$, b_n form a basis for L, then the unique elements $b_1^{\dagger}$, $\ldots$, $b_n^{\dagger} \in E$ satisfying $\langle b_i^{\dagger}, b_j\rangle = 1$ or 0 according as $i + j = n + 1$ or $i + j \neq n + 1$ form a basis for $L^{\dagger}$. (This is the 'cobasis' of E corresponding to the basis b_1, $\ldots$, b_n, numbered backwards for later convenience.) One has $\operatorname{rk} L^{\dagger} = \operatorname{rk} L$ and $L^{\dagger\dagger} = L$.

3. Examples in algebraic number theory

In this section we discuss three types of lattices that are naturally encountered in algebraic number theory. The examples are not typical of the examples that we shall encounter later on, and readers without an interest in algebraic number theory may safely skip this section.

Additive groups of algebraic numbers. Let K be an algebraic number field, i. e., a field that is a finite extension of the field $\mathbf{Q}$ of rational numbers, and let L be a finitely generated subgroup of the additive group of K; for example, one may take L to be the ring $\mathbf{Z}_K$ of algebraic integers in K, or a fractional $\mathbf{Z}_K$-ideal. Then L carries a natural lattice structure, which is defined by

$$q(x) = \sum_{\sigma} |\sigma x|^2$$

for $x \in L$, with σ ranging over the set of field embeddings of K in the field $\mathbf{C}$ of complex numbers, and where $|\ |$ denotes the usual absolute value on $\mathbf{C}$.

Multiplicative groups of algebraic numbers. One can deal with multiplicative subgroups in a similar manner. Let K again be an algebraic number field, and denote by μ the set of roots of unity in K, which is a finite cyclic subgroup of the multiplicative group K^* of K. Let now L be a finitely generated subgroup of the quotient group K^*/μ. Then L has a natural lattice structure, which this time is defined by

$$q(x\mu) = \sum_p \sum_\sigma (\log |\sigma x|_p)^2$$

for $x\mu \in L \subset K^*/\mu$; here p ranges over the set $\{\infty, 2, 3, 5, 7, \ldots\}$ of 'primes' of $\mathbf{Q}$, and σ ranges, for fixed p, over the set of field embeddings of K in an algebraic closure $\bar{\mathbf{Q}}_p$ of the p-adic completion $\mathbf{Q}_p$ of $\mathbf{Q}$; each $\bar{\mathbf{Q}}_p$ is chosen once and for all, and $|\ |_p$ denotes, for $p < \infty$, the p-adic absolute value on $\bar{\mathbf{Q}}_p$ with $|p|_p = 1/p$, whereas on $\bar{\mathbf{Q}}_\infty = \mathbf{C}$ one takes $|\ |_\infty = |\ |$. If one takes $L = \mathbf{Z}_K^*/\mu$, where $\mathbf{Z}_K^*$ denotes the group of units of $\mathbf{Z}_K$, then all terms with $p \neq \infty$ vanish, and one obtains a lattice of which the rank is one less than the number of infinite places of K.

Elliptic curves. Consider an elliptic curve $\mathcal{E}$ over $\mathbf{Q}$, defined by a Weierstrass equation $y^2z + a_1xyz + a_3yz^2 = x^3 + a_2x^2z + a_4xz^2 + a_6z^3$, with all $a_i \in \mathbf{Q}$. It is well-known that the set $\mathcal{E}(\mathbf{Q})$ of points $(x : y : z)$ in the projective plane $\mathbf{P}^2(\mathbf{Q})$ that satisfy the equation, is in a natural way an abelian group, the *Mordell-Weil group* of $\mathcal{E}$ over $\mathbf{Q}$. Denote by $\mathcal{E}(\mathbf{Q})_{\mathrm{tor}}$ its subgroup of elements of finite order. Then $L = \mathcal{E}(\mathbf{Q})/\mathcal{E}(\mathbf{Q})_{\mathrm{tor}}$ is a lattice with

$$q(\bar{P}) = \frac{1}{2} \cdot \lim_{n\to\infty} \frac{h(2^nP)}{4^n}$$

for $P \in \mathcal{E}(\mathbf{Q})$, where $\bar{P}$ denotes the image of P in L and where for an element $(x : y : z) \in \mathbf{P}^2(\mathbf{Q})$, with $\mathbf{Z}x + \mathbf{Z}y + \mathbf{Z}z = \mathbf{Z}$, one defines $h(x : y : z) = \log \max\{|x|, |y|, |z|\}$; the number $q(\bar{P})$ is known as the *canonical height* of P.

4. Representing lattices

Two different normalizations. Suppose that L is a lattice of full rank in a Euclidean vector space E. Writing $n = \operatorname{rk} L$, one has an isomorphism $L \cong \mathbf{Z}^n$ of groups as well as an isomorphism $E \cong \mathbf{R}^n$ of Euclidean vector spaces. However, these two isomorphisms are generally not compatible, and if, for whatever reason, one wishes to introduce coordinates, then one needs to choose between the two. Each option has its virtues, and the usefulness of the concept of lattices is in no small part due to the possibility of thinking about them in two different ways.

As we shall see, in many applications of lattices one takes L equal to $\mathbf{Z}^n$ and q equal to a function that reflects the problem at hand. On the other hand, when thinking about lattices one will often find it useful to imagine them as being embedded in ordinary Euclidean n-space, with $q(x)$ proportional to the square of the distance from x to the origin. Here n is bounded only by the limits of one's imagination. Experience shows that, even when the fourth dimension proves too hard to picture in one's mind, one can still avoid the common pitfall of implicitly assuming that the rank n of L is small, such as 2 or 3. Several subtle phenomena occur only for large n, and the fact that the LLL algorithm runs in polynomial time even when n varies, is one of the keys to its success.

Representing lattices numerically. If one wishes to run an algorithm on a lattice, one needs to specify the lattice and its elements in some numerical manner. There are many ways of doing this, and the two most important ones correspond to the two possibilities mentioned above. The first is to specify a lattice by writing down a real positive definite symmetric $n \times n$ matrix $\mathbf{A} = (a_{ij})_{1 \le i,j \le n}$; the lattice L is then understood to be the abelian group $\mathbf{Z}^n$, its elements are represented as (column) vectors with n integral entries, and q is given by $q(x) = x^T \mathbf{A} x$ for $x \in L$, the superscript T denoting passage to the transpose. In order to be able to write down $\mathbf{A}$ by means of a finite number of bits, one may require that all the a_{ij} are rational, and that they are represented as $a_{ij} = a'_{ij}/d$ where d and all a'_{ij} are integers represented in binary, and $d > 0$.

The second way of specifying a lattice is by writing down a real $m \times n$ matrix $\mathbf{B} = (b_{ij})_{1 \le i \le m, 1 \le j \le n}$ of rank n; in this case, L is understood to be the subgroup $\sum_{j=1}^{n} \mathbf{Z} b_j$ of $\mathbf{R}^m$, where $b_j = (b_{ij})_{i=1}^{m}$ and where $\mathbf{R}^m$ has the standard inner product. The elements of L are then represented as real m-vectors. Again, one may require the entries of $\mathbf{B}$ to be rational, so that the coordinates of all elements of L are rational as well.

Conversion. Whenever we discuss algorithms for lattices, it will always be assumed that

lattices are specified in one of the two ways just described, by means of a matrix with rational entries. Which of the two one uses is immaterial, since there are polynomial-time algorithms for converting each type of presentation into the other. In one direction this is easy: the second type is converted into the first by the formula $\mathbf{A} = \mathbf{B}^T \cdot \mathbf{B}$. The conversion in the other direction is a little more laborious, and for lack of a suitable reference we give a quick sketch of a possible way to proceed. Given $\mathbf{A}$, one first uses the Gram-Schmidt process to diagonalize the induced quadratic form on $\mathbf{Q}^n$ (see Section 10). This has the effect of writing $\mathbf{A} = \mathbf{C}_1^T \cdot \mathbf{D} \cdot \mathbf{C}_1$, where $\mathbf{C}_1$ is an upper triangular $n \times n$ matrix over $\mathbf{Q}$, with 1's on the diagonal, and $\mathbf{D}$ is a diagonal matrix with n positive rational diagonal entries d_j. Using a naive greedy algorithm, one writes each d_j as the sum of $m_j = O(\log \max\{2, \log d'_j\})$ squares of non-zero rational numbers, where d'_j denotes the product of the numerator and the denominator of d_j (if one allows a probabilistic algorithm, as in [27], one can take $m_j \leq 4$). With $m = \sum_j m_j$, this leads to an $m \times n$ matrix $\mathbf{C}_2$ over $\mathbf{Q}$, with exactly one non-zero entry per row, such that $\mathbf{D} = \mathbf{C}_2^T \cdot \mathbf{C}_2$; and now the matrix $\mathbf{B} = \mathbf{C}_2 \cdot \mathbf{C}_1$ has rank n and satisfies $\mathbf{A} = \mathbf{B}^T \cdot \mathbf{B}$, as desired. This procedure, while running in polynomial time, does give rise to a fairly large value for m, which is not bounded by a function of n alone. The probabilistic algorithm from [27] leads to $m \leq 4n$. Theoretically, one can achieve $m \leq n + 3$ (see [6, Chapter 6, Example 8]), but I do not know whether this can be done by means of an algorithm that is efficient in any sense of the word.

Whenever we assert that a lattice algorithm runs in polynomial time, then we mean that its run time is bounded by a polynomial function of the number of bits of the input, where all lattices forming part of the input or output of the algorithm are specified by a rational matrix $\mathbf{A}$ or $\mathbf{B}$ as above; that length will be at least the rank of the input lattice.

Other representations. There are other natural ways of specifying a lattice. For example, if $f: L \to L'$ is a group homomorphism from a lattice L to a lattice L', then the kernel and the image of f are sublattices of L and L', respectively. If L and L' are specified by an $n \times n$ matrix $\mathbf{A}$ and an $n' \times n'$ matrix $\mathbf{A}'$ as above, so that $L = \mathbf{Z}^n$ and $L' = \mathbf{Z}^{n'}$, then the map $f: \mathbf{Z}^n \to \mathbf{Z}^{n'}$ is given by an $n' \times n$ matrix $\mathbf{F}$ over $\mathbf{Z}$; the three matrices $\mathbf{A}$, $\mathbf{A}'$, $\mathbf{F}$ can then serve to specify both the kernel and the image of f. One may convert this type of presentation into one of the earlier ones by means of the kernel and image algorithm presented in Section 14.

The examples from Section 3 show that sometimes lattices can be specified in ways that are very difficult to convert to any of our standard formats. For example, one can specify an algebraic number field K by means of a defining equation over $\mathbf{Q}$, and this

defining equation is then sufficient to specify the lattice $L = \mathbf{Z}_K$. However, no polynomial-time algorithm is known for actually finding a basis for $L = \mathbf{Z}_K$ over $\mathbf{Z}$ (even when one restricts to the case $[K : \mathbf{Q}] = 2$; see [4]), and for typical fields K with $[K : \mathbf{Q}] > 2$ the function q is not $\mathbf{Q}$-valued. Similar comments apply to the unit lattice $L = \mathbf{Z}_K^*/\mu$, for which a $\mathbf{Z}$-basis appears to be even harder to compute, and to the Mordell-Weil lattices $L = \mathcal{E}(\mathbf{Q})/\mathcal{E}(\mathbf{Q})_{\mathrm{tor}}$, for which $\mathbf{Z}$-bases are not even known to be computable.

5. The determinant

Definition of the determinant. After the rank, the most important numerical invariant attached to a lattice L is its *determinant*, denoted by $d(L)$. It is defined by

$$d(L) = \lim_{r\to\infty} \frac{\operatorname{vol} B(\sqrt{r})}{\#\{y \in L : q(y) \le r\}},$$

where for $n = \operatorname{rk} L$ we define $B(\sqrt{r})$ to be the ball $\{x \in \mathbf{R}^n : \langle x, x\rangle \le r\}$ of radius $\sqrt{r}$ in $\mathbf{R}^n$, and vol denotes the standard n-dimensional volume. One has $\operatorname{vol} B(\sqrt{r}) = r^{n/2} \cdot \operatorname{vol} B(1) = r^{n/2} \cdot \pi^{n/2}/\frac{n}{2}!$, where $\frac{n}{2}!$ is inductively defined by $0! = 1$, $\frac{1}{2}! = \sqrt{\pi}/2$, and $\frac{n}{2}! = \frac{n}{2} \cdot \frac{n-2}{2}!$ for $n \ge 2$. (One has $\frac{n}{2}! = \Gamma(1 + \frac{n}{2})$.) To understand the definition of $d(L)$, and to show that the limit exists, one may assume L to be embedded in the standard Euclidean vector space $\mathbf{R}^n$. Let $\mathbf{B}$ be a non-singular real $n \times n$ matrix such that the columns b_j of $\mathbf{B}$ form a basis for L. Then the subset

$$F = \sum_{j=1}^{n} [0,1)b_j = \Big\{\sum_{j=1}^{n} c_j b_j : c_j \in \mathbf{R},\ 0 \le c_j < 1 \text{ for } 1 \le j \le n\Big\}$$

of $\mathbf{R}^n$ satisfies $\operatorname{vol} F = |\det \mathbf{B}|$, and F is a fundamental domain for L in the sense that each $x \in \mathbf{R}^n$ has a unique representation $x = y + z$ with $y \in L$ and $z \in F$. Restricting to the set of all $y \in L$ with $q(y) \le r$, one proves that the disjoint union $\bigcup_y (y+F)$, taken over those y, is a fair approximation to $B(\sqrt{r})$; more precisely, if one puts $s = \sup\{\langle z, z\rangle : z \in F\}$, then that union is contained in $B(\sqrt{r} + \sqrt{s})$, and for $r \ge s$ it contains $B(\sqrt{r} - \sqrt{s})$. Comparing volumes, one deduces

$$\lim_{r\to\infty} \frac{\#\{y \in L : q(y) \le r\} \cdot |\det \mathbf{B}|}{\operatorname{vol} B(\sqrt{r})} = 1.$$

It follows that $d(L)$ is well-defined, and that one has in fact $d(L) = |\det \mathbf{B}| = \operatorname{vol} F$. In particular, $\operatorname{vol} F$ is independent of the choice of the basis.

The zero lattice has determinant 1.

Hadamard's inequality. Let L, $b_1, \ldots, b_n$, F be as above. The volume of the parallelepiped F is at most the product of the lengths of the vectors b_i, so we have

$$d(L) \leq \prod_{i=1}^{n} \|b_i\|.$$

This is *Hadamard's inequality*, which is valid for any basis $b_1, \ldots, b_n$ of a lattice L. Equality holds if and only if the vectors b_i are pairwise orthogonal, in the sense that $\langle b_i, b_j \rangle = 0$ whenever $i \neq j$. In Section 10 we will see that if the basis $b_1, \ldots, b_n$ is *reduced* in a suitable sense, then one has the opposite inequality

$$\prod_{i=1}^{n} \|b_i\| \leq c_n \cdot d(L),$$

where c_n depends only on the rank n of the lattice. Thus, a 'reduced' basis may be thought of as being 'nearly orthogonal'.

Formulae for the determinant. There are many formulae that can be used in the computation of $d(L)$, in addition to the formula $d(L) = |\det \mathbf{B}|$ mentioned above. If L is given by means of a matrix $\mathbf{A}$ as in Section 4, then one has $d(L) = (\det \mathbf{A})^{1/2}$. These two formulae suffice for most algorithmic and numerical purposes. In a more theoretical context, they can be supplemented by the following rules. Let L be a lattice. If M is a sublattice of finite index $(L : M)$ of L, then one has $d(M) = (L : M) \cdot d(L)$. If M is a pure sublattice of L (see Section 2), then one has $d(L) = d(M) \cdot d(L/M)$. For the dual $L^\dagger$ of L, one has $d(L^\dagger) = 1/d(L)$. If L is embedded as a lattice of full rank in a Euclidean vector space E, and $\tau\colon E \to E$ is a non-singular linear map, then τL is a lattice, and one has $d(\tau L) = |\det \tau| \cdot d(L)$. The proofs are elementary and may be left to the reader.

The volume discrepancy. Let E_1 and E_2 be Euclidean vector spaces, and let $\tau\colon E_1 \to E_2$ be a linear map. We can associate to τ a positive real number $\gamma(\tau)$, the *volume discrepancy* of τ, in the following way. Let $(\ker \tau)^\perp$ be the orthogonal complement of the kernel of τ in E_1. Then τ restricts to a vector space isomorphism $(\ker \tau)^\perp \to \tau E_1$. Identifying each of $(\ker \tau)^\perp$ and τE_1, as Euclidean vector spaces, with $\mathbf{R}^{\operatorname{rank} \tau}$, we obtain a non-singular linear map $\tau'\colon \mathbf{R}^{\operatorname{rank} \tau} \to \mathbf{R}^{\operatorname{rank} \tau}$, and we define $\gamma(\tau) = |\det \tau'|$; the independence of the choice of identifications with $\mathbf{R}^{\operatorname{rank} \tau}$ can either be shown directly, or be deduced from the formula $d(\tau L) = \gamma(\tau) \cdot d(L)$, which is valid for any lattice L of full rank in $(\ker \tau)^\perp$. In the

case $E_1 = E_2$ one has $\gamma(\tau) = |\det \tau|$ if τ is non-singular, but not if τ is singular, since one has $\gamma(\tau) > 0$.

Write $\tau^\dagger \colon E_2 \to E_1$ for the linear map that is *adjoint* to τ; it is characterized by the property that $\langle x, \tau^\dagger y\rangle = \langle \tau x, y\rangle$ for all $x \in E_1$, $y \in E_2$. One has

$$\gamma(\tau) = \gamma(\tau^\dagger).$$

One can prove this by using that any square matrix and its transpose have the same determinant, or by considering dual lattices.

Some care is required with computing the volume discrepancy of a composed map. If E_3 is a third Euclidean vector space, and $\sigma \colon E_2 \to E_3$ is a linear map, then the formula $\gamma(\sigma\tau) = \gamma(\sigma)\gamma(\tau)$ is valid if one has $\tau E_1 = (\ker \sigma)^\perp$, but not in much greater generality.

The definition of the volume discrepancy given by Lang [19, Chapter V, Section 2] generalizes the definition just given: the number $\gamma(\tau)$ defined above equals the volume discrepancy, as defined by Lang, of the exact sequence $0 \to \ker \tau \to E_1 \xrightarrow{\tau} E_2 \to E_2/\tau E_1 \to 0$. A still more general perspective is offered by De Smit [9].

Determinants of kernels and images. Let L_1 and L_2 be lattices, and let $f \colon L_1 \to L_2$ be a group homomorphism. Embed L_1 and L_2 as lattices of full rank in Euclidean vector spaces E_1 and E_2, respectively, and write $f_{\mathbf{R}}$ for the $\mathbf{R}$-linear map $E_1 \to E_2$ induced by f. Then we have

$$d(\ker f) \cdot d(fL_1) = \gamma(f_{\mathbf{R}}) \cdot d(L_1)$$

with $\gamma(f_{\mathbf{R}})$ as defined above (cf. [19, Chapter V, Theorem 2.1]). To prove this, one observes that $\ker f_{\mathbf{R}}$ is the $\mathbf{R}$-subspace of E_1 spanned by the pure sublattice $\ker f$ of L_1, and that $L = L_1/\ker f$ may be viewed as a lattice of full rank in $(\ker f_{\mathbf{R}})^\perp$ satisfying $f_{\mathbf{R}} L = fL_1$. Next one uses the formulae $d(L_1) = d(\ker f) \cdot d(L_1/\ker f)$ and $d(f_{\mathbf{R}} L) = \gamma(f_{\mathbf{R}}) \cdot d(L)$ that we encountered earlier.

The adjoint $f_{\mathbf{R}}^\dagger$ of $f_{\mathbf{R}}$ restricts to a map $f^\dagger \colon L_2^\dagger \to L_1^\dagger$. From $\gamma(f_{\mathbf{R}}) = \gamma(f_{\mathbf{R}}^\dagger)$ and $d(L_1^\dagger) = d(L_1)^{-1}$ one obtains the *six lattices formula*

$$d(\ker f) \cdot d(fL_1) \cdot d(L_1^\dagger) = d(\ker f^\dagger) \cdot d(f^\dagger L_2^\dagger) \cdot d(L_2).$$

This formula is often helpful in computing determinants of lattices; see Sections 7 and 8 for illustrations.

6. The shortest vector problem

Existence of short vectors. The *shortest vector problem*, also known as the *homogeneous approximation problem*, is the following: given a lattice L of positive rank, find a non-zero element $x \in L$ with $q(x)$ smallest possible. The formulation may be interpreted in several ways: writing $\lambda(L) = \min\{q(x) : x \in L,\ x \neq 0\}$, one may actually wish to find $x \in L$ with $q(x) = \lambda(L)$; or one may, in an algorithmic context, take 'smallest possible' to mean: smallest possible given the time that one is willing to spend.

The main theoretical result about the problem is the following.

Theorem of Minkowski. *Each lattice L of positive rank n contains a non-zero element x with $q(x) \leq \frac{4}{\pi} \cdot \frac{n}{2}!^{2/n} \cdot d(L)^{2/n} \leq n \cdot d(L)^{2/n}$.*

To see why this is true, assume again $L \subset \mathbf{R}^n$, and put $\lambda = \lambda(L) = \min\{q(x) : x \in L,\ x \neq 0\}$. Then no two distinct points of L have distance smaller than $\sqrt{\lambda}$, so if one writes $B' = \{z \in \mathbf{R}^n : \langle z, z\rangle < \lambda/4\}$, then the open balls $y + B'$ of radius $\sqrt{\lambda}/2$ centered at the lattice points $y \in L$ are pairwise disjoint. Since the sets $y + F$ from the previous proof disjointly cover $\mathbf{R}^n$ as y ranges over L, one deduces that $\operatorname{vol} B' \leq \operatorname{vol} F = d(L)$. Using that $\operatorname{vol} B' = (\sqrt{\lambda}/2)^n \cdot \operatorname{vol} B(1)$ one obtains the first inequality, and the second follows from the fact that $B(1)$ contains a cube with edge length $2/\sqrt{n}$. By Stirling's theorem, one actually has $\frac{4}{\pi} \cdot \frac{n}{2}!^{2/n} = \frac{2+o(1)}{\pi \mathrm{e}} \cdot n$ for $n \to \infty$.

The Hermite constant. Both $\lambda(L)$ and $d(L)$ are *homogeneous functions* of L, of degrees 2 and n, respectively; that is, if inside $\mathbf{R}^n$ one replaces L by tL for some positive real number t (or, equivalently, the function q by $t^2 \cdot q$), then λ is replaced by $t^2 \cdot \lambda$ and one has $d(tL) = t^n \cdot d(L)$. Hence, $d(L)^{2/n}$ is the only power of $d(L)$ that has the same degree as $\lambda(L)$, and therefore the only power of $d(L)$ that can possibly occur in a result like Minkowski's theorem. The supremum of $\lambda(L)/d(L)^{2/n}$, taken over all lattices L of rank n, is called the *Hermite constant* and denoted by γ_n. Minkowski's theorem, as stated above, is equivalent to the inequalities $\gamma_n \leq \frac{4}{\pi} \cdot \frac{n}{2}!^{2/n} \leq n$. It is known that $n/(2\pi\mathrm{e}) \leq \gamma_n \leq n/(\pi\mathrm{e} + o(1))$ for $n \to \infty$; see [8, Chapter 1, Section 1] for more information and a slightly better result.

There is a sense in which, for a 'random' lattice of given positive rank n, the inequality $\lambda(L) \leq \gamma_n \cdot d(L)^{2/n}$ is close to best possible. However, the lattices that occur in many applications are by no means random. As we shall see, one often constructs a lattice in such a manner that it has an 'exceedingly short' non-zero vector if and only if a certain problem has a solution, and that solution can then be read off from the short vector. In such cases, Minkowski's theorem plays at best a secondary role.

Construction of short vectors. A salient feature of the proof of Minkowski's theorem is its non-constructive character. The existence of x is shown by a measure-theoretic version of the pigeon-hole principle, and no efficient algorithm for actually finding x can be read from the proof. Indeed, all known algorithms for computing $\lambda(L)$, or for finding a lattice vector x as in Minkowski's theorem, perform some sort of complete enumeration, and fail to run in polynomial time for varying n (cf. Section 12).

In Section 11 we shall see that the construction of a 'fair' approximation to the shortest non-zero element of L is a byproduct of so-called *lattice basis reduction* algorithms, such as the LLL algorithm. The LLL algorithm does run in polynomial time, but the non-zero vector $x \in L$ that it finds is not guaranteed to be the shortest one, or to be as short as in Minkowski's theorem. The quantity $q(x)/d(L)^{2/n}$ will be bounded by a function of n alone, but this is an exponential function rather than a linear function as in Minkowski's theorem. For example, the standard variant of the LLL algorithm produces a non-zero element $x \in L$ with

$$q(x) \leq 2^{n-1} \cdot \lambda(L), \qquad q(x) \leq 2^{(n-1)/2} \cdot d(L)^{2/n}$$

(see Section 11). It is both fortunate and surprising that these exponential aberrations are small enough for most applications.

Short vectors in the dual lattice. Let E be a Euclidean vector space. Write $': E - \{0\} \to E - \{0\}$ for inversion in the unit sphere, so that $x' = x/\langle x, x\rangle$; note that x' is a vector lying in the same direction from the origin as x, but with length equal to the inverse of the length of x. For each $x \in E - \{0\}$, one has $x'' = x$, and one also verifies easily that the subgroup $\{y \in E : \langle x, y\rangle \in \mathbf{Z}\}$ of E is the (orthogonal) sum of the subgroup $\mathbf{Z}x'$ generated by x' and the orthogonal complement $(\mathbf{R}x)^{\perp} = \{y \in E : \langle x, y\rangle = 0\}$ of the subspace spanned by x. Thus, $\{y \in E : \langle x, y\rangle \in \mathbf{Z}\}$ is the union, over $m \in \mathbf{Z}$, of the translates $(\mathbf{R}x)^{\perp} + mx'$ of the hyperplane $(\mathbf{R}x)^{\perp}$, and the successive distances between these translates are equal to $\|x'\| = 1/\|x\|$.

Next let $L \subset E$ be a lattice of full rank, and let $L^{\dagger}$ be its dual. For $x \in E - \{0\}$, one has $x \in L^{\dagger}$ if and only if L is contained in the set $\{y \in E : \langle x, y\rangle \in \mathbf{Z}\} = (\mathbf{R}x)^{\perp} + \mathbf{Z}x'$. Since x is 'short' if and only if x' is 'long', and since every hyperplane in E is of the form $(\mathbf{R}x)^{\perp}$, we conclude that the shortest vector problem for $L^{\dagger}$ is equivalent to the following problem posed in terms of L: given L, find a hyperplane H in E such that L is contained in a collection of maximally widely spaced translates of H. The latter problem is useful

in enumerating all lattice vectors that lie in a certain region, and it has applications in integer programming (see [1, 22]).

Minkowski's theorem now implies that for any lattice L of positive rank n there is a hyperplane H as above, such that the distance between the successive translates of H is at least $\gamma_n^{-1/2} \cdot d(L)^{1/n}$; and with the LLL algorithm one can find a hyperplane that is within a factor $2^{(n-1)/2}$ from optimal.

7. Diophantine approximation

This section and the next are devoted to some traditional applications of the shortest vector problem. For additional applications, see Sections 13–15 below.

Continued fractions. Suppose that α is a real number. Then the continued fraction expansion of α gives rise to a sequence p_0/q_0, p_1/q_1, p_2/q_2, ... of rational numbers, with $\mathbf{Z}p_i + \mathbf{Z}q_i = \mathbf{Z}$ and $q_i > 0$ for all i, such that $|\alpha - p_i/q_i| < 1/q_i^2$ for all i and such that any similarly written rational number p/q satisfying $|\alpha - p/q| < 1/(2q^2)$ occurs in the sequence (see [16, Chapter X]). If α is rational then the sequence is finite; likewise, when α is irrational but known or given to finite precision only, as is often the case in an algorithmic context, then only finitely many terms of the sequence are meaningful.

Thus, the continued fraction expansion gives rise to a sequence of rational approximations p/q to a given real number α that are 'good' in the sense that the error tends to 0 fairly quickly as a function of the denominator q of the approximation.

It is instructive to see how one can achieve a similar purpose with the help of a lattice. Let again α be a real number, and define the lattice L, q by $L = \mathbf{Z}^2$ and

$$q(x, y) = N \cdot (x - \alpha y)^2 + y^2 \qquad \text{for } x, y \in \mathbf{Z},$$

where N is a suitably chosen 'large' real number. One verifies that $\operatorname{rk} L = 2$ and $d(L) = N^{1/2}$, so there is a non-zero element $(x, y) \in L$ with $q(x, y) \le \gamma_2 N^{1/2}$, where $\gamma_2 = \sqrt{4/3}$ is the Hermite constant for $n = 2$ (see Section 9). Also, in algorithmic circumstances one can actually find such a vector efficiently (see Section 9). If $N > \gamma_2^2$ then from $(x - \alpha y)^2 \le q(x, y)/N \le \gamma_2/N^{1/2} < 1$ one deduces $y \ne 0$, and the inequality of the means implies $|N^{1/2} \cdot (x - \alpha y)| \cdot |y| \le q(x, y)/2 \le \gamma_2 N^{1/2}/2$, so that we have

$$\left|\alpha - \frac{x}{y}\right| \le \frac{\gamma_2/2}{y^2}, \qquad 0 < |y| \le \gamma_2^{1/2} N^{1/4}.$$

Thus one obtains a rational approximation to α that is of the same quality as what one obtains from the continued fraction algorithm. The main difference is that the continued

fraction algorithm yields a whole sequence of approximations; to achieve this with lattices, one would need to vary N and therefore consider a family of lattices. A discussion of techniques for doing this, and for deciding which values of N are the crucial ones, falls outside the scope of the present introduction. In most circumstances where 'good' rational approximations to a real number α are required, a single well-chosen number N will do.

Higher-dimensional diophantine approximation. The approximation problem just discussed allows several natural generalizations to higher dimensions, two of which will be discussed. Many corresponding higher-dimensional extensions of the continued fraction method have been proposed, but none appears to have all the properties that one desires. The translation into the shortest vector problem for a suitably constructed lattice generalizes readily to higher dimensions, and here again one encounters a proliferation of algorithms; that is, while in rank 2 there appears to exist only one reasonable lattice basis reduction algorithm (see Section 9), there is an entire family of them in rank greater than 2 (see Section 11).

Simultaneous diophantine approximation. Let k real numbers $\alpha_1, \ldots, \alpha_k$, with $k \geq 1$, be given, and suppose that one is interested in finding simultaneous rational approximations x_i/y to α_i, all with the same denominator y; for $k = 1$ this is the problem discussed above. For general k, one can introduce the lattice L, q defined by $L = \mathbf{Z}^{k+1}$ and

$$q(x_1, x_2, \ldots, x_k, y) = N \cdot \sum_{i=1}^{k} (x_i - \alpha_i y)^2 + y^2$$

for $(x_1, x_2, \ldots, x_k, y) \in \mathbf{Z}^{k+1}$, where N plays the same role as above. One has $\operatorname{rk} L = k+1$ and $d(L) = N^{k/2}$. In the same manner as for $k = 1$ one now deduces that for $N > \gamma_{k+1}^{k+1}$ there is a integer vector $(x_1, x_2, \ldots, x_k, y)$ with $y \neq 0$ and

$$y^2 \leq \gamma_{k+1} N^{k/(k+1)}, \qquad \sum_{i=1}^{k} (x_i - \alpha_i y)^2 \leq \frac{k \cdot (\gamma_{k+1}/(k+1))^{1+1/k}}{|y|^{2/k}}.$$

In addition, with the LLL algorithm one can actually find such a vector, but with $2^{k/2}$ replacing γ_{k+1}.

Here is a possible algorithmic application of simultaneous diophantine approximation. Suppose one is given a $k \times k$ matrix $\mathbf{C}$ with integer entries and with $\det \mathbf{C} \neq 0$, as well as a column vector $b \in \mathbf{Z}^k$. Then there is a unique column vector $z \in \mathbf{R}^k$ with $\mathbf{C}z = b$, and the entries $z_1, \ldots, z_k$ of z are rational; say $z_i = p_i/q$, where $q \in \mathbf{Z}$ is a common denominator. Further, suppose that one in interested in efficiently and *exactly* computing z, but that the

only linear algebra package at one's disposal works in real precision. Then one can proceed as follows. First, use the linear algebra package to compute an approximate solution vector α, so that the entries of $\mathbf{C}\alpha$ are very close to the entries of b and the entries α_i of α are very close to z_i. If the approximations are good enough, then the lattice defined above will for large enough N contain an exceptionally short vector, namely the (unknown!) vector $(p_1, \ldots, p_k, q)$. Next, one applies the LLL algorithm; it will find a non-zero vector $(x_1, \ldots, x_k, y) \in \mathbf{Z}^{k+1}$ that is at most 2^k times as long, and therefore still quite short; so short, that one estimates the integers entries of $\mathbf{C}x - yb$ (which is close to the tiny vector $y \cdot (\mathbf{C}\alpha - b)$) to be smaller than 1 in absolute value. Consequently, one has actually $\mathbf{C}x = yb$ and therefore $z = x/y$. The reader may enjoy filling in the details and working out explicit inequalities that make the argument valid.

There is a very similar but more complicated application of simultaneous rational approximations to linear programming (see [28]).

Approximate linear dependencies. In a second higher-dimensional generalization of the approximation problem, one is given k real numbers $\alpha_1, \ldots, \alpha_k$, with $k \geq 2$, and one is interested in finding an 'approximate' linear relation with integer coefficients among the α_i, that is, a sequence $x_1, \ldots, x_k$ of integers, not all zero, such that $\left|\sum_i x_i \alpha_i\right|$ is small in relation to the sizes of the x_i themselves. With $k = 2$, $\alpha_2 = 1$ this amounts to the problem of finding a good rational approximation to α_1 that we considered above. Generally, one can take $L = \mathbf{Z}^k$ and define q by

$$q(x_1, x_2, \ldots, x_k) = \sum_{i=1}^{k} x_i^2 + N \cdot \Big(\sum_{i=1}^{k} x_i \alpha_i\Big)^2 \qquad (x_i \in \mathbf{Z}),$$

where N is again a suitably large real number. We claim that one has

$$d(L) = \Big(1 + N \cdot \sum_{i=1}^{k} \alpha_i^2\Big)^{1/2}.$$

To prove this, consider the standard Euclidean vector spaces $E_1 = \mathbf{R}^k$ and $E_2 = E_1 \times \mathbf{R} = \mathbf{R}^{k+1}$, and define $\tau\colon E_1 \to E_2$ by $\tau\big((x_i)_{i=1}^k\big) = \big((x_i)_{i=1}^k, \sqrt{N} \cdot \sum_{i=1}^k \alpha_i x_i\big)$. The lattices $L_1 = \mathbf{Z}^k$ and $L_2 = \tau L_1 + \mathbf{Z} \cdot (0, 1)$ are of full rank in E_1 and E_2, and L may as a lattice be identified with τL_1. The six lattices formula from Section 5 now simplifies to the statement that $d(L)$ equals the determinant of the kernel of the map $\tau^\dagger\colon L_2^\dagger \to L_1^\dagger$; since the vector $\big((-\sqrt{N} \cdot \alpha_i)_{i=1}^k, 1\big)$ generates that kernel, its length $\big(1 + N \cdot \sum_{i=1}^k \alpha_i^2\big)^{1/2}$ equals $d(L)$.

One can now apply Minkowski's theorem to prove a general existence theorem for approximate linear dependencies. In addition, the LLL algorithm will find one.

In a typical practical application, one is interested in detecting a *true* linear dependency among certain numbers β_i, and each α_i is a good approximation to β_i. For example, with $\beta_i = \beta^{i-1}$ one may attempt to detect an algebraic number β from a numerical approximation. A very similar application will be encountered in Section 13.

Non-archimedean approximation. The approximation problems discussed so far were concerned with *real* numbers, and the quality of the approximations was measured by means of the *real* absolute value. Sometimes it is felt that a different notion of lattice would be required if instead we are concerned with *p-adic numbers* and the *p-adic absolute value*. This is not true: both problems just considered, when transferred to a p-adic context, can still be addressed by means of suitably constructed lattices. The problem of finding approximate linear dependencies may serve as illustration.

Let p be a prime number, denote by $\mathbf{Z}_p$ the ring of p-adic integers, by $\mathbf{Q}_p$ the field of fractions of $\mathbf{Z}_p$, and by $|\ |_p$ the p-adic absolute value on $\mathbf{Q}_p$ with $|p|_p = 1/p$. Given k elements $\alpha_1, \ldots, \alpha_k$ of $\mathbf{Q}_p$, with $k \geq 2$, one looks for integers $x_1, \ldots, x_k$ that are not 'too large' in the usual absolute value, and not all zero, such that $\sum_{i=1}^k x_i\alpha_i$ is p-adically very close to 0. As in the case of real numbers, the p-adic numbers α_i will in an algorithmic context need to be specified to some finite precision; and in fact, if one wishes that $|\sum_{i=1}^k x_i\alpha_i|_p \leq p^{-m}$ for some given integer m, then it suffices to know the α_i modulo $p^m\mathbf{Z}_p$. Thus, we shall assume that the α_i are specified by means of approximations α_i' that belong to the ring $\mathbf{Z}[1/p]$ of rational numbers whose denominator is a power of p, and that are guaranteed to satisfy $|\alpha_i - \alpha_i'|_p \leq p^{-m}$. If that is the case, then for $x_i \in \mathbf{Z}$ one has $|\sum_i x_i\alpha_i|_p \leq p^{-m}$ if and only if $\sum_i x_i\alpha_i' \in p^m\mathbf{Z}$. We describe two constructions of lattices that one can use to find 'small' integers x_i, not all zero, with the latter property.

In the first construction, one simply takes L to be the subgroup

$$\{x = (x_i)_{i=1}^k \in \mathbf{Z}^k : \textstyle\sum_i x_i\alpha_i' \in p^m\mathbf{Z}\}$$

of $\mathbf{Z}^k$, with $q(x) = \sum_i x_i^2$ for $x = (x_i)_{i=1}^k \in L$. One has $\operatorname{rk} L = k$ and $d(L) = p^{m-m'}$, where m' denotes the largest integer for which $p^{m'}\mathbf{Z}$ contains all α_i' as well as p^m. In many practical situations all α_i are in $\mathbf{Z}_p$ but not all are in $p\mathbf{Z}_p$, and $m \geq 0$; then one has $m' = 0$ and $d(L) = p^m$. A short non-zero vector in L, obtained with Minkowski's theorem or with LLL, gives rise to an approximate dependency as one requires. However, it should be observed that L has not been specified in one of the standard formats from Section 4.

Thus, before LLL can be applied, one needs to find a basis for L. One way of addressing this problem is found in Section 14. For now, we can achieve the same result by using the second construction instead.

In the second construction, one takes $L = \mathbf{Z}^{k+1}$ (so $\operatorname{rk} L = k+1$), with q defined by

$$q(x_1, x_2, \ldots, x_k, y) = \sum_{i=1}^{k} x_i^2 + N \cdot \Big(p^m y - \sum_{i=1}^{k} x_i \alpha_i'\Big)^2,$$

where N is a 'large' positive rational number. One has $d(L) = p^m N^{1/2}$. Suppose that $(x_1, x_2, \ldots, x_k, y)$ is a short non-zero lattice vector. Then the number $z = p^m y - \sum_{i=1}^{k} x_i \alpha_i'$ belongs to $p^{m'}\mathbf{Z}$, with m' as defined above, and if N is large enough then from the smallness of the vector and the inequality $z^2 \leq q(x_1, x_2, \ldots, x_k, y)/N$ one deduces $|z| < p^{m'}$. One concludes that $z = 0$, so that $\sum_{i=1}^{k} x_i \alpha_i' \in p^m \mathbf{Z}$. Therefore the x_i do yield an approximate linear dependency, and from $\sum_i x_i^2 \leq q(x_1, x_2, \ldots, x_k, y)$ one sees that the x_i are not too large.

As an interesting exercise, the reader may compare the quality of the approximations obtained from both constructions.

The p-adic absolute value that we just considered is a non-archimedean valuation of *mixed characteristic*, in the sense that the residue class field and the field on which the valuation is defined have different characteristics. One may also consider approximation problems for non-archimedean valuations of *equal characteristic*. These do give rise to a different notion of lattice, which we briefly treat in Section 16.

8. The nearest vector problem

Inhomogeneous approximation. The *nearest vector problem*, also known as the *inhomogeneous approximation problem*, is the following: given a lattice L in a Euclidean vector space E, and an element $x \in E$, find $y \in L$ with smallest possible distance $d(x, y)$. By analogy with the case $L = \mathbf{Z} \subset \mathbf{R} = E$, one can think of this problem as a 'rounding' problem. As with the shortest vector problem in Section 6, the formulation allows for a strict and for a more relaxed interpretation.

For given $x \in E$, the set $\{x - y : y \in L\}$ equals the coset $x + L$ of L in E, which is discrete in E; the nearest vector problem asks for an element of smallest possible length in this coset.

Let E_0 be the subspace of E spanned by L, and denote the orthogonal projection of $x \in E$ on E_0 by x_0. Then for all $y \in L$ one has $d(x, y)^2 = d(x, x_0)^2 + d(x_0, y)^2$, so the

nearest vector problem does not change if one replaces E, x by E_0, x_0. Thus without loss of generality one may assume that L spans E. In an algorithmic context one will usually also assume that the coordinates of x, when expressed on a basis for L, are rational numbers.

For $x = 0$ the nearest vector problem is solved by $y = 0$; so this special case is *not* the same as the shortest vector problem. Nevertheless, one thinks of the nearest vector problem as being harder than the shortest vector problem, and there are several observations that support this feeling. For one thing, the direct analogue of Minkowski's theorem is wrong; that is, if the rank n is greater than 1, then one cannot guarantee the existence, for each $x \in E$, of an element $y \in L$ for which $d(x, y)$ is bounded by a function of n and $d(L)$ alone (a suitable function of n, $d(L)$, and $\lambda(L)$ will do, however). There is also a formal result stating that the shortest vector problem reduces to no more than $n = \operatorname{rk} L$ nearest vector problems, in the following manner (cf. [12]). Let $b_1, \ldots, b_n$ be a basis for L, and for each $j = 1, 2, \ldots, n$, let L_j be the sublattice $\{\sum_i n_i b_i : n_i \in \mathbf{Z}, n_1, n_2, \ldots, n_j \text{ are } \textit{even}\}$ of L. Then each set $b_j + L_j$ is a coset of L_j in L. Their (disjoint) union, for $1 \leq j \leq n$, equals $L - 2L$, so if $x_j \in b_j + L_j$ has minimal length then the shortest among $x_1, \ldots, x_n$ will be a shortest non-zero element of L; and similarly one can reduce a relaxed version of the shortest vector problem to n instances of a relaxed version of the nearest vector problem.

The extended Euclidean algorithm. Let $a_1, \ldots, a_k$ be positive integers, with $k \geq 2$, and put $d = \gcd(a_1, \ldots, a_k)$. In the case $k = 2$, the Euclidean algorithm can be used to compute d when a_1 and a_2 are given, and with the extended Euclidean algorithm one can compute 'small' integers x_1 and x_2 with $x_1 a_1 + x_2 a_2 = d$ (see [5; 18, Section 4.5.2]). Proceeding by induction on k, one can compute $d = \gcd(\gcd(a_1, \ldots, a_{k-1}), a_k)$ in polynomial time when $a_1, \ldots, a_k$ are given, and one can also inductively compute integers $x_1, \ldots, x_k$ with $\sum_i x_i a_i = d$; however, for $k > 2$ the integers x_i computed in this manner will in general be very far from 'smallest possible'. Thus, one is faced with the question: given $a_1, \ldots, a_k$, as well as an integer solution $x = (x_i)_{i=1}^k$ to the equation $\sum_i x_i a_i = d$, find the smallest possible integer solution to the same equation. If we measure the 'size' of a solution by means of the Euclidean norm, then this is an instance of the nearest vector problem. Namely, let L be the lattice in $\mathbf{R}^k$ (with the standard inner product) defined by

$$L = \{y = (y_i)_{i=1}^k \in \mathbf{Z}^k : \sum_i y_i a_i = 0\}.$$

Then if $y \in L$ has smallest possible distance to x, the vector $x - y$ will be the smallest solution that one is looking for. One has $\operatorname{rk} L = k - 1$, and the six lattices formula from Section 5 readily implies $d(L) = \left(\sum_i (a_i/d)^2\right)^{1/2}$.

Note that L is not given in one of the standard formats from Section 4, so before one can apply a lattice basis reduction algorithm one needs to find a basis for L. It is possible to obtain such a basis as a byproduct of the inductive computation that yields d and the initial solution x. However, in Section 14 we shall see a much easier solution to the problem: if one works with the right lattice, then one can entirely forgo the inductive computation, and directly find both d and a 'small' solution to $\sum_i x_i a_i = d$ by means of a lattice basis reduction algorithm.

Finding the nearest vector. As for the shortest vector problem, all known algorithms for solving the nearest vector problem perform some sort of complete enumeration, and they fail to run in polynomial time when the rank of L varies (cf. Section 12). However, the LLL algorithm can be used to find an *approximate* solution. That is, the LLL algorithm computes a basis for a lattice L that is 'reduced' in a suitable sense, and once a reduced basis is available one can, for given $x \in E$, efficiently compute an element $y \in L$ such that

$$d(x, y) \leq 2^n \cdot \min\{d(x, y') : y' \in L\},$$

where $n = \operatorname{rk} L$ (see Sections 10 and 11). An alternative formulation of the same algorithm is given in Section 14: given L and x, a lattice L' is constructed such that a 'reduced' basis for L' immediately yields $y \in L$ as above.

9. Lattices of rank two

Lattices of rank two are easy to picture and to understand, and they play a pivotal role in lattice basis reduction algorithms.

Reduced bases in rank two. Let L be a lattice with $\operatorname{rk} L = 2$, embedded in a two-dimensional Euclidean vector space E, and let b_1, $b_2 \in L$. Define the real numbers a, b, c by

$$a = q(b_1), \qquad b = q(b_1 + b_2) - q(b_1) - q(b_2) = 2\langle b_1, b_2 \rangle, \qquad c = q(b_2).$$

Then for x, $y \in \mathbf{R}$ one has $q(xb_1 + yb_2) = ax^2 + bxy + cy^2$. We have $b^2 - 4ac \leq 0$, with strict inequality if and only if b_1, b_2 are linearly independent (over $\mathbf{R}$, or over $\mathbf{Z}$). The vectors b_1, b_2 form a basis for L if and only if one has $b^2 - 4ac = -4d(L)^2$. We call b_1, b_2 a *reduced basis* for L if one has

$$q(b_1) = \lambda(L) = \min\{q(x) : x \in L,\, x \neq 0\},$$
$$b_2 \in L - \mathbf{Z}b_1, \qquad q(b_2) = \min\{q(x) : x \in L - \mathbf{Z}b_1\}.$$

It is automatic that any reduced basis for L is a basis for L. Conversely, if b_1, b_2 form a basis for L, then they form a reduced basis if and only if one has $a > 0$ and $|b| \leq a \leq c$. It is clear from the definition that any lattice of rank 2 has a reduced basis.

The shortest and nearest vector problems. Let L and E be as above, and suppose that a reduced basis b_1, b_2 for L is available. Let a, b, c be defined as above. Then both the shortest vector problem and the nearest vector problem admit easy solutions. For the shortest vector problem this is obvious: b_1 is a shortest non-zero vector of L, and one has $\lambda(L) = q(b_1) = a \leq (4/3)^{1/2} d(L)$; the last inequality follows from $4d(L)^2 = 4ac - b^2 \geq 4a^2 - a^2 = 3a^2$. Considering the case $|b| = a = c > 0$ one proves that the Hermite constant γ_2 equals $(4/3)^{1/2}$.

The vector $-b_1$ is also a shortest non-zero vector of L, and the others, if any, are among $\pm b_2$, $\pm b_2 \pm b_1$.

For the nearest vector problem, assume $b = 2\langle b_1, b_2 \rangle \geq 0$, replacing b_2 by $-b_2$ if necessary. Define

$$F = \{z \in E : q(z) \leq q(z - y) \text{ for all } y \in \{\pm b_1, \pm b_2, \pm(b_1 - b_2)\}\}.$$

This is a hexagon if $b \neq 0$, and a rectangle if $b = 0$. Each $x \in E$ can be written as $x = y + z$ with $y \in L$ and $z \in F$, and in an algorithmic context such a representation is for given x not hard to find. For 'most' x it is unique, but whether or not it is unique, it is always true that z is an element of the coset $x + L$ of minimal length, and that y is a lattice element with minimal distance to x; so y solves the nearest vector problem for L and x.

It follows that the supremum, over all $x \in E$, of $\min\{q(x - y) : y \in L\}$ is equal to $\max\{q(z) : z \in F\}$. The latter number is given by the convenient formula

$$\max\{q(z) : z \in F\} = \frac{q(b_1) \cdot q(b_2) \cdot q(b_1 - b_2)}{4d(L)^2} = \frac{a \cdot c \cdot (a - b + c)}{-b^2 + 4ac},$$

where it is still assumed that $0 \leq b \leq a \leq c$. The reader may recognize the formula that expresses the circumradius of a plane triangle in terms of its area and the lengths of its sides.

Lattice basis reduction in rank two. Given a basis b_1, b_2 for a lattice L of rank 2, the following iterative procedure replaces b_1, b_2 by a reduced basis. Let m be an integer nearest to $\langle b_1, b_2 \rangle / \langle b_1, b_1 \rangle$, and replace b_2 by $b_2 - m b_1$. The new vector b_2 now satisfies $|\langle b_1, b_2 \rangle| \leq \frac{1}{2}\langle b_1, b_1 \rangle$. If it also satisfies $q(b_2) \geq q(b_1)$, then the basis b_1, b_2 is reduced, as desired; otherwise, interchange b_1 and b_2, and start all over again.

The procedure just described goes back to Gauss [11], who used the language of binary quadratic forms. There is a strong analogy with the Euclidean algorithm for the computation of the greatest common divisor of two non-zero integers a_1, a_2: in a typical iteration step of the latter, one replaces a_2 by $a_2 - ma_1$, where m equals a_2/a_1 rounded to an integer. The 'ideal' value $m = a_2/a_1$ would make the new value of a_2 equal to zero. Analogously, the ideal value $m = \langle b_1, b_2\rangle / \langle b_1, b_1\rangle$ would make the new vector b_2 orthogonal to b_1 in the sense that $\langle b_1, b_2\rangle = 0$; one recognizes the Gram-Schmidt orthogonalization process. The actual choice of m minimizes the value of $q(b_2 - mb_1)$ over $m \in \mathbf{Z}$; in particular, the new vector b_2 satisfies $q(b_2) \leq q(b_2 - b_1)$ and $q(b_2) \leq q(b_2 + b_1)$, which will be useful below.

Performing the procedure above for the sublattice L of $\mathbf{Z}^2$ with basis $b_1 = (Na_1, 0)$, $b_2 = (Na_2, 1)$ (where N is a suitably large integer) is in fact tantamount to the Euclidean algorithm for a_1, a_2.

Termination. The value of $q(b_1)$ decreases throughout the procedure just described. Since there are only finitely many vectors in L whose length is bounded by the length of the initially given vector b_1, this implies that the procedure terminates in all cases.

To find a good bound for the number of iteration steps, we prove that in each step, except possibly the last two, the value of $q(b_1)$ decreases by a factor 3 or higher. That is to say, if in a certain step it occurs that, after the replacement of b_2 by $b_2 - mb_1$, the new vector b_2 satisfies $q(b_2) > q(b_1)/3$, then that step is either the last one or the next-to-last one. Namely, suppose it is not the last one; then one has $q(b_2) < q(b_1)$. The inequality $|\langle b_1, b_2\rangle| \leq \frac{1}{2}\langle b_1, b_1\rangle < \frac{3}{2}\langle b_2, b_2\rangle$ then implies that the value for m in the next step will be one of 0, 1, -1, and since all of the vectors b_1, $b_1 - b_2$, $b_1 + b_2$ are at least as long as b_2, that next step will be the last one, as asserted.

It follows that an upper bound for the number of iteration steps is given by $2 + \big(\log(q(b_{1,\text{initial}})/q(b_{1,\text{final}}))\big)/\log 3$, where $b_{1,\text{initial}}$ and $b_{1,\text{final}}$ are the initially given basis vector b_1 and the basis vector b_1 as finally produced, respectively; here $q(b_{1,\text{final}}) = \lambda(L)$.

Suppose next that we are in an algorithmic context, and that L and its basis are specified by means of a rational matrix $\mathbf{A}$ (or $\mathbf{B}$) as in Section 4. Then $q(L)$ is contained in $\mathbf{Z}\frac{1}{d}$ (or $\mathbf{Z}\frac{1}{d^2}$) if d is a positive integer for which $\mathbf{Z}\frac{1}{d}$ contains the entries of $\mathbf{A}$ (or $\mathbf{B}$), and therefore one has $q(b_{1,\text{final}}) \geq \frac{1}{d}$ (or $\frac{1}{d^2}$). Combining this with the bound for the number of iteration steps just given, one now easily deduces that the entire algorithm runs in polynomial time.

10. Flags

Flags. It will be convenient to formulate lattice basis reduction algorithms for general rank not in terms of bases but in terms of flags. In this section, L denotes a lattice embedded in a Euclidean vector space E, with $n = \operatorname{rk} L = \dim E$. A *flag* of L is a sequence $\mathcal{F} = (L_i)_{i=0}^n$ of pure sublattices L_i of L (as defined in Section 2) satisfying $\operatorname{rk} L_i = i$ (for $0 \leq i \leq n$) and $L_{i-1} \subset L_i$ (for $0 < i \leq n$). Clearly one has $L_0 = \{0\}$ and $L_n = L$.

Every basis $b_1, \ldots, b_n$ for L gives rise to the flag $\left(\sum_{j \leq i} \mathbf{Z} b_j\right)_{i=0}^n$. Conversely, each flag is of this form, but generally not for a unique basis; more precisely, two bases $a_1, \ldots, a_n$ and $b_1, \ldots, b_n$ for L give rise to the same flag if and only if there are integers c_{ij}, for $1 \leq j \leq i \leq n$, such that $b_i = \sum_{j \leq i} c_{ij} a_j$ and $c_{ii} = \pm 1$ for all i. Thus, a flag may be said to carry a little less information than a basis.

Successive distances and the Gram-Schmidt process. Let $\mathcal{F} = (L_i)_{i=0}^n$ be a flag of L. For $1 \leq i \leq n$, the *ith successive distance* $l_i(\mathcal{F})$ of $\mathcal{F}$ is defined by $l_i(\mathcal{F}) = d(L_i/L_{i-1})$.

The successive distances are related to the Gram-Schmidt orthogonalization process. Let $b_1, \ldots, b_n$ be a basis for L that gives rise to $\mathcal{F}$. For each i, let b_i^* be the unique vector in $b_i + \sum_{j<i} \mathbf{R} b_j$ that is orthogonal to $\sum_{j<i} \mathbf{R} b_j$. The vectors b_i^* can be computed by means of the Gram-Schmidt orthogonalization process, that is, by an inductive application of the formula

$$b_i^* = b_i - \sum_{j<i} \mu_{ij} b_j^*, \qquad \text{where } \mu_{ij} = \frac{\langle b_i, b_j^* \rangle}{\langle b_j^*, b_j^* \rangle}.$$

One has $b_1^* = b_1$. With this notation, $l_i(\mathcal{F})$ is equal to the length $\|b_i^*\|$ of b_i^* or, equivalently, to the distance of b_i to the subspace $\sum_{j<i} \mathbf{R} b_j$ of E. In particular, one has $l_1(\mathcal{F}) = \|b_1\|$.

The size of a flag. Let $\mathcal{F} = (L_i)_{i=0}^n$ be a flag of L. The *size* $s(\mathcal{F})$ of $\mathcal{F}$ is defined by $s(\mathcal{F}) = \prod_{i=0}^n d(L_i)$. From $d(L_i) = \prod_{j \leq i} l_j(\mathcal{F})$ it follows that $s(\mathcal{F})$ can be expressed in terms of the successive distances by $s(\mathcal{F}) = \prod_{j=1}^n l_j(\mathcal{F})^{n+1-j}$.

It is not difficult to prove that a given lattice L has, for each real number r, only finitely many flags of size at most r. Imprecisely speaking, a flag will be interesting for us if it has small size $s(\mathcal{F}) = \prod_{j=1}^n l_j(\mathcal{F})^{n+1-j}$, and this will be the case if the 'weight' in the product $\prod_{j=1}^n l_j(\mathcal{F})$, which assumes the constant value $d(L)$, is shifted towards the factors with large j. This may serve as a motivation for the following definition, which describes more precisely the property that one desires a flag to have.

Reduced flags. Let c be a real number, $c \geq 1$, and let $\mathcal{F}$ be a flag for L. We say that $\mathcal{F}$ is *c-reduced* if for each j with $0 < j < n$ one has $l_{j+1}(\mathcal{F})^2 \geq l_j(\mathcal{F})^2/c$; for $c = 1$ this is

equivalent to the sequence of successive distances being non-decreasing, and the condition becomes weaker as c gets larger. Not every lattice has a flag that is 1-reduced, but as we shall see, each lattice has a flag that is 4/3-reduced, and for each $c > 4/3$ a c-reduced flag can be quickly found. The standard choice is to take $c = 2$.

The shortest vector problem. Suppose $n > 0$. A c-reduced flag $\mathcal{F} = (L_i)_{i=0}^n$ gives rise to an approximate solution to the shortest vector problem, the quality of the approximation being measured by c. Namely, put $L_1 = \mathbf{Z}b_1$. Then b_1 is 'almost' the shortest non-zero vector of L in the sense that

$$q(b_1) \leq c^{n-1} \min\{q(x) : x \in L - \{0\}\} = c^{n-1}\lambda(L).$$

To see this, let $x \in L - \{0\}$, and let i be minimal with $x \in L_i$; then $\|x\|$ is at least the ith successive distance $l_i(\mathcal{F})$, so

$$q(x) = \|x\|^2 \geq l_i(\mathcal{F})^2 \geq c^{1-i} l_1(\mathcal{F})^2 \geq c^{1-n} q(b_1),$$

as required. Combining the inequality just proved with Minkowski's theorem, we see that $q(b_1) \leq n \cdot c^{n-1} \cdot d(L)^{2/n}$, but this can be improved a little. Namely, multiplying together the inequalities $q(b_1) = l_1(\mathcal{F})^2 \leq c^{i-1} l_i(\mathcal{F})^2$ that we just proved, for $i = 1, \ldots, n$, and using that $\prod_{i=1}^n l_i(\mathcal{F}) = d(L)$, one finds

$$q(b_1) \leq c^{(n-1)/2} \cdot d(L)^{2/n}.$$

We also see from our inequalities that b_1 itself is actually a shortest non-zero vector of L if one has $l_1(\mathcal{F}) = \min\{l_i(\mathcal{F}) : 1 \leq i \leq n\}$, which occurs if $c = 1$.

The nearest vector problem. A c-reduced flag $\mathcal{F} = \left(\sum_{j \leq i} \mathbf{Z}b_i\right)_{i=0}^n$ also gives rise to an approximate solution to the nearest vector problem, the quality of the approximation again being measured by c. To see this, let b_i^* be as above, and write

$$F_i = \Big\{ \sum_{j=1}^{i} \mu_j b_j^* : \mu_i \in \mathbf{R}, -\tfrac{1}{2} < \mu_j \leq \tfrac{1}{2} \text{ for } 1 \leq j \leq i \Big\}, \qquad F = F_n.$$

By induction on i one checks that each $x \in \sum_{j \leq i} \mathbf{R}b_j$ admits a unique representation of the form $x = y + z$ with $y \in \sum_{j \leq i} \mathbf{Z}b_j$ and $z \in F_i$. In particular, each $x \in E$ can be written uniquely as $x = y + z$ with $y \in L$ and $z \in F$; moreover, in an algorithmic context this

representation is easy to find. Thus, a c-reduced flag can be used to find, for every $x \in E$, an element $y \in L$ with

$$d(x,y)^2 \leq \max\{\langle z,z\rangle : z \in F\} = \tfrac{1}{4} \cdot \sum_{i=1}^{n} l_i(\mathcal{F})^2.$$

Also, the approximation of a given element $x \in E$ by an element $y \in L$ obtained in this way is not far from optimal, in the sense that for each other $y' \in L$ one has

$$d(x,y)^2 \leq (1 + c + \ldots + c^{n-1}) \cdot d(x,y')^2.$$

To prove this, express $z = x - y$ and $z' = x - y'$ on the orthogonal basis $(b_j^*)_{j=1}^n$ of E:

$$z = \sum_{j=1}^{n} \mu_j b_j^*, \qquad z' = \sum_{j=1}^{n} \mu_j' b_j^*,$$

with μ_j, $\mu_j' \in \mathbf{R}$, $-\frac{1}{2} < \mu_j \leq \frac{1}{2}$. From $z - z' \in L - \{0\}$ one deduces that the largest i with $\mu_i \neq \mu_i'$ exists and satisfies $\mu_i - \mu_i' \in \mathbf{Z}$. Then one has the inequalities $|\mu_i'| \geq \frac{1}{2}$ and

$$q(z') = \sum_{j=1}^{n} {\mu_j'}^2 l_j(L)^2 \geq \tfrac{1}{4} l_i(L)^2 + \sum_{i<j\leq n} \mu_j^2 l_j(L)^2,$$

$$q(z) \leq \tfrac{1}{4} \cdot \sum_{j\leq i} l_j(L)^2 + \sum_{i<j\leq n} \mu_j^2 l_j(L)^2 \leq \tfrac{1}{4}(c^{i-1} + \ldots + c + 1) l_i(L)^2 + \sum_{i<j\leq n} \mu_j^2 l_j(L)^2,$$

which yield the desired inequality $q(z) \leq (1 + c + \ldots + c^{n-1}) \cdot q(z')$.

Specifying flags, size-reduced bases. If one wishes to do computations with flags, one will need a way of specifying them numerically. Assuming that the lattice and its elements are specified in one of the standard formats of Section 4, one can specify a flag $\mathcal{F} = (L_i)_{i=0}^n$ by listing the elements of a basis $b_1, \ldots, b_n$ for L that gives rise to $\mathcal{F}$. This representation is not unique, but it becomes unique, up to choosing n signs, if one requires in addition that for each i the vector $b_i - b_i^*$ belongs to the fundamental domain F_{i-1} for $L_{i-1} = \sum_{j<i} \mathbf{Z} b_j$ in $\sum_{j<i} \mathbf{R} b_j$ defined above. A basis with this property is called *size-reduced*. To change a given basis for a lattice into a size-reduced one that gives rise to the same flag, it suffices to subtract a suitable element of L_{i-1} from b_i, for each i.

In the course of computations, it may not be necessary to insist that no other bases than size-reduced ones be used for the purpose of specifying flags. However, size-reduced

bases are important both in practice and in theory, because they help both in preventing excessive coefficient growth and in obtaining low run time estimates.

It will be convenient to say that a basis $b_1, \ldots, b_n$ for L is *c-reduced*, for a real number $c \geq 1$, if it is size-reduced and the corresponding flag $\left(\sum_{j \leq i} \mathbf{Z} b_j\right)_{i=0}^{n}$ is c-reduced.

Near-orthogonality of c-reduced bases. Let c be a real number, $c \geq 1$, and suppose that $b_1, \ldots, b_n$ is a c-reduced basis for a lattice L. With the notation as above, we have $b_i = b_i^* + \sum_{j<i} \mu_{ij} b_j^*$ for certain real numbers μ_{ij} with $-\frac{1}{2} < \mu_{ij} \leq \frac{1}{2}$, and this implies

$$q(b_i) \leq q(b_i^*) + \tfrac{1}{4} \sum_{j<i} q(b_j^*) \leq q(b_i^*) + \tfrac{1}{4} \sum_{j<i} c^{i-j} q(b_i^*) = \left(1 + \tfrac{1}{4}(c^i - c)/(c-1)\right) \cdot q(b_i^*),$$

where $(c^i - c)/(c-1) = i - 1$ if $c = 1$. Taking the product over i and using that $\prod_i \|b_i^*\| = d(L)$ we find

$$\prod_{i=1}^{n} \|b_i\| \leq \prod_{i=1}^{n} \left(1 + \tfrac{1}{4}(c^i - c)/(c-1)\right)^{1/2} \cdot d(L).$$

Thus, for fixed c, a c-reduced basis is 'nearly orthogonal' in the sense of Section 5. If $c \geq 4/3$, then the inequalities just given can be simplified to

$$q(b_i) \leq c^{i-1} \cdot q(b_i^*) \text{ for } 1 \leq i \leq n,$$

$$\prod_{i=1}^{n} \|b_i\| \leq c^{n(n-1)/4} \cdot d(L).$$

Successive minima and c-reduced bases. For $1 \leq i \leq n$, the *ith successive minimum* $\lambda_i(L)$ of L is defined to be the infimum of the set of all real numbers r with the property that L contains at least i linearly independent vectors a with $q(a) \leq r$; equivalently, it is the *minimum* of that set of real numbers. Clearly, we have $\lambda_1(L) = \lambda(L)$. The following result shows that the successive minima can be approximately computed from a c-reduced basis.

Proposition. *Let c be a real number with $c \geq 4/3$, and let $b_1, \ldots, b_n$ be a c-reduced basis for a lattice L. Then we have*

$$c^{1-n} \cdot q(b_i) \leq \lambda_i(L) \leq \max\{q(b_j) : 1 \leq j \leq i\} \leq c^{i-1} \cdot q(b_i)$$

for $1 \leq i \leq n$.

Proof. Since $b_1, \ldots, b_i$ are i linearly independent vectors, the middle inequality is immediate from the definition of $\lambda_i(L)$. For $1 \leq j \leq i$ we have

$$q(b_j) \leq c^{j-1} \cdot q(b_j^*) \leq c^{i-1} \cdot q(b_i^*) \leq c^{i-1} \cdot q(b_i),$$

which implies the third inequality. For the lower bound, let $\mathcal{F} = (L_j)_{j=0}^n$ be the flag of L that $b_1, \ldots, b_n$ gives rise to. Choose k minimal such that L_k contains all $a \in L$ with $q(a) \leq \lambda_i(L)$. The set of such a has rank at least i, so we have $k \geq i$, and therefore

$$l_k(\mathcal{F})^2 \geq c^{i-k} \cdot l_i(\mathcal{F})^2 = c^{i-k} \cdot q(b_i^*) \geq c^{1-k} \cdot q(b_i) \geq c^{1-n} \cdot q(b_i).$$

By definition of k, at least one a does not belong to L_{k-1}, so we have $l_k(\mathcal{F})^2 \leq q(a) \leq \lambda_i(L)$. This proves the stated lower bound for $\lambda_i(L)$ and completes the proof of the Proposition.

The dual flag. Let L be a lattice with dual $L^\dagger$ (see Section 2), and let $\mathcal{F} = (L_i)_{i=0}^n$ be a flag of L. For $M \subset L$, write $M^\perp = \{x \in L^\dagger : \langle x, y\rangle = 0 \text{ for all } y \in M\}$; this is a pure sublattice of $L^\dagger$. Then $\mathcal{F}^\perp = (L^\perp_{n+1-i})_{i=0}^n$ is a flag of $L^\dagger$, and one has $\mathcal{F}^{\perp\perp} = \mathcal{F}$. If c is a real number with $c \geq 1$, then $\mathcal{F}$ is c-reduced if and only if $\mathcal{F}^\perp$ is c-reduced; this follows from the equality $l_i(\mathcal{F}) l_j(\mathcal{F}^\perp) = 1$ for $i + j = n + 1$. If $(b_i)_{i=1}^n$ is a basis for L that gives rise to $\mathcal{F}$, then the corresponding cobasis $(b_i^\dagger)_{i=1}^n$ (see Section 2) gives rise to $\mathcal{F}^\perp$. It is not generally true, for a real number $c \geq 2$, that $(b_i)_{i=1}^n$ is c-reduced if and only if $(b_i^\dagger)_{i=1}^n$ is c-reduced, though this is valid for $\operatorname{rk} L \leq 2$.

11. Finding a good flag

Flags in rank two. Suppose L is a lattice of rank 2. Giving a flag $\mathcal{F} = (L_i)_{i=0}^2$ of L is the same as giving a pure sublattice $L_1 = \mathbf{Z}b_1$ of rank 1 of L, since necessarily one has $L_0 = \{0\}$ and $L_2 = L$; the size $s(\mathcal{F})$ of such a flag is given by $s(\mathcal{F}) = l_1(\mathcal{F})d(L) = \|b_1\| \cdot d(L)$, so finding a flag of small size is equivalent to finding a non-zero vector of small length. Also, one has $l_2(\mathcal{F}) = d(L)/l_1(\mathcal{F})$, so if c is a real number ≥ 1 then $\mathcal{F}$ is c-reduced, as defined in the previous section, if and only if one has $q(b_1) \leq \sqrt{c} \cdot d(L)$. Since the Hermite constant γ_2 equals $\sqrt{4/3}$, it follows that L has a $4/3$-reduced flag; and there is a lattice of rank 2 that does not have a c-reduced flag for any $c < 4/3$.

In Section 9 we saw a procedure for finding a $4/3$-reduced flag of L. If we rephrase one iteration step from that procedure in the language of flags, then we obtain the following: *if a flag $\mathcal{F}$ of L is not $4/3$-reduced, then one can find a flag $\mathcal{F}'$ with smaller size: $s(\mathcal{F}') < s(\mathcal{F})$.* Namely, let b_1, b_2 be a size-reduced basis for L giving rise to $\mathcal{F}$. Then one has $b_2 = b_2^* + \mu b_1$ with $|\mu| \leq \frac{1}{2}$, and therefore

$$q(b_2) = q(b_2^*) + \mu^2 q(b_1) \leq \Big(\frac{l_2(\mathcal{F})^2}{l_1(\mathcal{F})^2} + \frac{1}{4}\Big) \cdot q(b_1).$$

Since $\mathcal{F}$ is not 4/3-reduced, we have $l_2(\mathcal{F})^2/l_1(\mathcal{F})^2 < 3/4$, and therefore $q(b_2) < q(b_1)$; so the flag $\mathcal{F}'$ corresponding to the basis b_2, b_1 is of smaller size than $\mathcal{F}$.

Improving a given flag. Suppose next that L is a lattice of any rank n, and that $\mathcal{F} = (L_i)_{i=0}^n$ is a flag of L that is not 4/3-reduced. Then just as in the case of rank 2, one can find a flag $\mathcal{F}'$ of smaller size. To do this, first choose a *pivot*, i. e., an index j with $0 < j < n$ for which $l_{j+1}(\mathcal{F})^2 < \frac{3}{4} l_j(\mathcal{F})^2$. Such an index exists, since by assumption the flag is not 4/3-reduced. Then $(L_i/L_{j-1})_{i=j-1}^{j+1}$ is a flag of the rank two lattice L_{j+1}/L_{j-1}, and that flag is not 4/3-reduced either. Thus, by the rank two case that we just did, one can replace it by a flag $(L'_i/L_{j-1})_{i=j-1}^{j+1}$ of smaller size; here one has $L'_{j-1}/L_{j-1} = \{0\}$ and $L'_{j+1}/L_{j-1} = L_{j+1}/L_{j-1}$. Writing $L'_i = L_i$ for all $i \neq j$, one now obtains a flag $\mathcal{F}' = (L'_i)_{i=0}^n$ of L with $s(\mathcal{F}') < s(\mathcal{F})$. Notice that $\mathcal{F}'$ and $\mathcal{F}$ differ only in the rank j sublattice.

Referring back to what we just proved for rank 2, we see that the inequality $s(\mathcal{F}') < s(\mathcal{F})$ can be sharpened to

$$s(\mathcal{F}') \leq \Big(\frac{l_{j+1}(\mathcal{F})^2}{l_j(\mathcal{F})^2} + \frac{1}{4} \Big)^{1/2} \cdot s(\mathcal{F}).$$

This will be useful below.

Finding a 4/3-reduced flag. Let L be a given lattice, and let $\mathcal{F}$ be the flag of L corresponding to a given basis $b_1, \ldots, b_n$ for L. If $\mathcal{F}$ is not 4/3-reduced, then as we just saw we can replace $\mathcal{F}$ by a flag $\mathcal{F}'$ that has smaller size. Since there are only finitely many flags of size smaller than the initially given flag, this procedure will, upon iteration, terminate with a flag of L that is 4/3-reduced. This tells us, first, that each lattice has a 4/3-reduced flag and, second, how to find one in an algorithmic situation. Considering a size-reduced basis that gives rise to such a flag, we also conclude that each lattice has a 4/3-reduced basis.

A basis reduction algorithm. An algorithm that, given a lattice L in one of the standard formats of Section 4, produces a basis for L that is reduced in a certain sense, is called a *basis reduction algorithm*. For example, the procedure that we just sketched produces a basis that is 4/3-reduced. In the case $n = 2$, this procedure is nothing but the algorithm that we described in Section 9. For larger rank, the procedure becomes an actual basis reduction algorithm if it is supplemented with rules for choosing pivots and for deciding at which stages the basis corresponding to the current flag is to be replaced by a size-reduced basis.

It is an open problem whether, with appropriate rulings, the basis reduction algorithm obtained in this manner runs in polynomial time. As we saw in Section 9, it does run in

polynomial time in the case $n = 2$, and in fact it runs in polynomial time for any fixed value of n (see [23]). The main obstacle towards proving such a result for varying n, is finding a good upper bound for the number of flags that the algorithm goes through.

It turns out that, in order to obtain a polynomial-time basis reduction algorithm, it suffices to be a little less demanding: if, instead of insisting on a flag or a basis that is 4/3-reduced, one allows a flag or a basis that is c-reduced with $c > 4/3$, then for any fixed value of c such a flag or basis can be found in polynomial time. This is what we consider next.

Finding a c-reduced flag. Let a real number c with $c > 4/3$ be fixed, and let L be a lattice. The procedure that we indicated for finding a 4/3-reduced flag can in an obvious way be shortened so as to find a flag that is merely c-reduced. One uses only pivots j with $l_{j+1}(\mathcal{F})^2 < l_j(\mathcal{F})^2/c$, and at each step the improved flag $\mathcal{F}'$ satisfies

$$s(\mathcal{F}') \leq \Big(\frac{l_{j+1}(\mathcal{F})^2}{l_j(\mathcal{F})^2} + \frac{1}{4}\Big)^{1/2} \cdot s(\mathcal{F}) < \sqrt{1/c + 1/4} \cdot s(\mathcal{F}),$$

where $\sqrt{1/c + 1/4} < 1$. Starting from an initially given flag $\mathcal{F}_{\text{initial}}$, one terminates with a flag $\mathcal{F}_{\text{final}}$ that is c-reduced. Each time the flag is changed, its size gets multiplied by a factor smaller than $\sqrt{1/c + 1/4}$, so the number of times this happens is at most $\big(\log(s(\mathcal{F}_{\text{initial}})/s(\mathcal{F}_{\text{final}}))\big)/|\log \sqrt{1/c + 1/4}|$. As in Section 9 one sees that in an algorithmic situation a good lower bound for $s(\mathcal{F}_{\text{final}})$ is available. This leads to an upper bound for the number of flags encountered in the course of the algorithm, an upper bound that is good enough to allow for a straightforward proof that the algorithm runs in polynomial time. The algorithm just described is the *LLL algorithm*. Properly speaking, the LLL algorithm is an entire family of algorithms, since there is considerable freedom in choosing c, in choosing the pivots, and in dealing with size-reduction.

The LLL algorithm. In summary, the LLL algorithm takes as input a lattice L, specified in one of the standard formats of Section 4, as well as a rational number $c > 4/3$; if no value for c is specified, we assume that $c = 2$. For any fixed value of c, the algorithm runs in polynomial time. The output of the algorithm is a basis for L that is c-reduced, as defined at the end of Section 10. If $n = \operatorname{rk} L > 0$, then the first basis vector b_1 of that basis yields an approximate solution to the shortest vector problem for L, in the sense that one has

$$q(b_1) \leq c^{n-1} \cdot \min\{q(x) : x \in L - \{0\}\}, \qquad q(b_1) \leq c^{(n-1)/2} \cdot d(L)^{2/n}.$$

Further, such a basis being available, one can approximately solve the nearest vector problem for L, in the sense of having a polynomial-time algorithm that given a vector x in the $\mathbf{Q}$-linear span of L finds $y \in L$ such that

$$d(x,y) \le (1 + c + \ldots + c^{n-1}) \cdot \min\{d(x,y') : y' \in L\}.$$

If $c = 2$, then the last inequality yields $d(x,y) \le 2^n \cdot \min\{d(x,y') : y' \in L\}$.

12. Enumerating short vectors

In the present section we show how one can enumerate short vectors in a lattice with the help of a reduced basis. The method runs at best in polynomial time for fixed values of $\operatorname{rk} L$. It relies on the following result, provided by R. J. Schoof, which gives upper bounds for the coefficients of a vector when expressed on a reduced basis, in terms of the length of the vector.

Lemma. *Let L be a lattice in a Euclidean vector space, and put $n = \operatorname{rk} L$. Let $b_1, \ldots, b_n$ be a basis for L, and let c be a real number with $c \ge 1$ such that $b_1, \ldots, b_n$ is c-reduced. For each $i = 1, \ldots, n$, denote by b_i^* the unique vector in $b_i + \sum_{j<i} \mathbf{R} b_j$ that is orthogonal to $\sum_{j<i} \mathbf{R} b_j$. Let $r_1, \ldots, r_n \in \mathbf{R}$, and put $x = \sum_{i=1}^n r_i b_i$. Then one has*

$$|r_j| \le (3\sqrt{c}/2)^{n-j} \cdot \frac{\|x\|}{\|b_j^*\|} \le c^{(n-1)/2} \cdot (3/2)^{n-j} \cdot \frac{\|x\|}{\|b_1\|}$$

for $j = 1, \ldots, n$.

Proof. By the definition of b_i^*, we can write $b_i - b_i^* = \sum_{j<i} \mu_{ij} b_j^*$ with $\mu_{ij} \in \mathbf{R}$. The basis $b_1, \ldots, b_n$ being c-reduced is equivalent to the inequalities

$$\|b_j^*\| \le c^{(i-j)/2} \|b_i^*\|, \qquad -\tfrac{1}{2} < \mu_{ij} \le \tfrac{1}{2}$$

being valid for $1 \le j < i \le n$ (see the definition in Section 10). Substituting $b_i = b_i^* + \sum_{j<i} \mu_{ij} b_j^*$ into $x = \sum_{i=1}^n r_i b_i$ we find that we have $x = \sum_j r_j^* b_j^*$ for $r_j^* = r_j + \sum_{i>j} \mu_{ij} r_i$. The orthogonality of the b_j^* implies $\|x\|^2 = \sum_j r_j^{*2} \|b_j^*\|^2$, so for each j we have

$$|r_j^*| \cdot \|b_j^*\| \le \|x\|.$$

We now prove the inequality $|r_j| \cdot \|b_j^*\| \leq (3\sqrt{c}/2)^{n-j} \cdot \|x\|$ by induction on $n-j$. From $r_j = r_j^* - \sum_{i>j} \mu_{ij} r_i$ and $|\mu_{ij}| \leq \frac{1}{2}$ we obtain

$$\begin{aligned} |r_j| \cdot \|b_j^*\| &\leq |r_j^*| \cdot \|b_j^*\| + \sum_{i>j} \tfrac{1}{2}|r_i| \cdot \|b_j^*\| \leq \|x\| + \tfrac{1}{2} \sum_{i>j} c^{(i-j)/2} \cdot |r_i| \cdot \|b_i^*\| \\ &\leq \Big(1 + \tfrac{1}{2} \sum_{i=j+1}^{n} c^{(i-j)/2} (3\sqrt{c}/2)^{n-i}\Big) \cdot \|x\| \\ &= \Big(1 + c^{(n-j)/2} \cdot \big((3/2)^{n-j} - 1\big)\Big) \cdot \|x\| \leq (3\sqrt{c}/2)^{n-j} \cdot \|x\|, \end{aligned}$$

as required. This proves the first inequality in the Lemma. The second one follows from $\|b_1\| = \|b_1^*\| \leq c^{(j-1)/2} \|b_j^*\|$. This proves the Lemma.

Computing $\lambda(L)$ and finding a shortest non-zero vector. If, in the notation of the Lemma, the r_i range independently over $\mathbf{Z}$, then x ranges over L. If x is a shortest non-zero vector of L, then one has $\|x\| \leq \|b_1\|$, so by the Lemma each $|r_i|$ is bounded by $c^{(n-1)/2} \cdot (3/2)^{n-i}$.

This suggests the following algorithm for computing $\lambda(L)$ for a given lattice L of positive rank n. First, use the LLL algorithm to find a 2-reduced basis $b_1, \ldots, b_n$ for L. Next, compute $q(x)$ for each x of the form $x = \sum_i r_i b_i$, where the r_i range independently over all integers that are at most $2^{(n-1)/2} \cdot (3/2)^{n-i}$ in absolute value. Now $\lambda(L)$ is equal to the minimal non-zero value of $q(x)$ that is found. The algorithm can also be used to compute all shortest non-zero vectors of L; these are the vectors x encountered that achieve the minimum.

Evidently, the number of systems of integers r_i to be tried by the algorithm is bounded by a function of n alone. Therefore, if L is specified in one of the standard formats of Section 4, the algorithm just described runs in polynomial time for any fixed value of $n = \operatorname{rk} L$.

Enumerating all short vectors. Suppose one is given a lattice L of positive rank n, as well as a positive real number r, and one is interested in listing all $x \in L$ with $q(x) \leq r$. Then one can proceed in a similar fashion: apply the LLL algorithm with $c = 2$ (say), and try all x of the form $\sum_i r_i b_i$, where each r_i is an integer satisfying $|r_i| \leq 2^{(n-1)/2} \cdot (3/2)^{n-i} \cdot \sqrt{r}/\|b_1\|$. For 'small' values of r—for example, no larger than $\lambda(L)$ multiplied by a function of n alone—the resulting algorithm will for fixed n run in polynomial time, as in the previous case.

In the case that r is 'large', there is a special advantage in using the sharper upper bound $|r_i| \leq (3/\sqrt{2})^{n-i} \cdot \sqrt{r}/\|b_i^*\|$ from the Lemma. Namely, the number of vectors to be

tried is in that case bounded by

$$\frac{r^{n/2}}{\prod_i \|b_i^*\|} = \frac{r^{n/2}}{d(L)}$$

multiplied by a function of n alone. By what we saw in Section 5, this is a good approximation to the number of vectors $x \in L$ with $q(x) \leq r$ to be enumerated, again up to a factor depending on n alone. In other words, for large enough r, the run time of the resulting algorithm is for fixed n bounded by the length of the output of the algorithm multiplied by a polynomial function of the length of the input. This will in fact be true if r is at least $1/\lambda(L^\dagger)$ times a suitable function of n.

The nearest vector problem. There is a similar enumeration algorithm for solving the nearest vector problem, which for any fixed value of $n = \operatorname{rk} L$ runs in polynomial time. To see how this works, let L be a lattice in a Euclidean vector space E with $n = \dim E = \operatorname{rk} L$, and let $x \in E$. We are interested in finding $y \in L$ with $q(x - y)$ minimal. One starts by applying the LLL algorithm, with any fixed $c > 4/3$. This gives rise to a c-reduced basis $b_1, \ldots, b_n$ for L, with Gram-Schmidt orthogonalization $b_1^*, \ldots, b_n^*$ as in the Lemma. In Section 10 we saw how to use this basis in order to find $y_0 \in L$ such that

$$q(x - y_0) \leq \tfrac{1}{4} \cdot \sum_{i=1}^{n} q(b_i^*) \leq \tfrac{1}{4} \cdot (c^{n-1} + \ldots + c + 1) \cdot q(b_n^*).$$

Write $x = \sum_i r_i b_i$ with $r_i \in \mathbf{R}$, and let the vector $y \in L$ one is looking for be written $y = \sum_i m_i b_i$ with $m_i \in \mathbf{Z}$. Then one has

$$(r_n - m_n)^2 \cdot q(b_n^*) \leq q(x - y) \leq q(x - y_0).$$

In view of our bound for $q(x - y_0)$, this leaves a number of possibilities for the integer m_n that is bounded by a function of n alone. For each $m \in \mathbf{Z}$ satisfying $(r_n - m)^2 \cdot q(b_n^*) \leq q(x - y_0)$, one now solves recursively the nearest vector problem for the lattice $L' = \sum_{i<n} \mathbf{Z} b_i$ of rank $n - 1$ and the element $x - mb_n - (r_n - m)b_n^*$ obtained by projecting $x - mb_n$ orthogonally to the subspace of E spanned by L'; for each value of m, this gives rise to a nearest vector $y_m \in L'$, and one finds the solution to the nearest vector problem for L and x by putting $y = y_m + mb_n$, the value for m being chosen so as to minimize $q(x - y_m - mb_n)$. One checks in a straightforward way that this correctly solves the nearest vector problem, and that for any fixed value of n it does so in polynomial time. Its practical performance can be enhanced by a branch-and-bound technique.

13. Factoring polynomials

The present section is devoted to the earliest published application of the LLL algorithm, namely the construction of a polynomial-time algorithm for the problem of factoring non-zero polynomials in $\mathbf{Q}[X]$ into irreducible factors (see [21]).

Summary description of the algorithm. Let $f \in \mathbf{Q}[X]$ be a given non-constant polynomial, and write $n = \deg f$. One starts by choosing a 'prime' p of the field $\mathbf{Q}$, and by finding an approximation β to a zero α of f in a finite extension of the completion $\mathbf{Q}_p$ of $\mathbf{Q}$ at p; for example, if one chooses $p = \infty$, then β will be a complex number close to a complex zero α of f, and one can compute β by means of techniques from numerical analysis. If f is reducible, then α is a zero of a non-zero polynomial in $\mathbf{Q}[X]$ of degree smaller than n, so 1, α, ..., α^{n-1} are linearly dependent over $\mathbf{Q}$, and 1, β, ..., β^{n-1} are approximately linearly dependent. As we saw in Section 7, one can formulate the problem of finding an approximate linear dependence relation among the β^i in lattice terms, and solve it by means of the LLL algorithm. If the vector found by LLL is short enough, then it will give rise to a non-trivial factor g of f, and otherwise f is irreducible. In the former case, one recursively applies the algorithm to g and f/g, which leads to the full factorization of f into irreducible factors in $\mathbf{Q}[X]$.

Intermezzo on Berlekamp's algorithm. In the more detailed description of the algorithm to be given below, we shall, instead of choosing $p = \infty$, take for p a prime number depending on f. The role of the numerical analysis is then played by a combination of Berlekamp's algorithm and Hensel's algorithm. For the latter, see [5; 30, Section 15.4]; to the former we devote the present intermezzo.

Berlekamp's algorithm takes as input a prime number p and a non-zero polynomial $f \in \mathbf{F}_p[X]$, and its output is the full factorization of f into irreducible factors in $\mathbf{F}_p[X]$. The algorithm is deterministic, and its run time is $O(p \cdot (\log p + \deg f)^c)$ for a positive constant c.

For simplicity of description, we shall make the assumptions that the discriminant of f is non-zero, that f has positive degree, and that f is *monic* in the sense of having leading coefficient 1; and in addition, instead of factoring f completely, we shall find a single irreducible factor. It would be easy to remove these restrictions, but for the purposes of our application there is no need to do so.

Our assumptions imply that $f = \prod_i f_i$ for certain pairwise distinct monic irreducible polynomials $f_1, \ldots, f_t \in \mathbf{F}_p[X]$. There is a ring isomorphism $\mathbf{F}_p[X]/(f) \cong$

$\prod_{i=1}^{t} \mathbf{F}_p[X]/(f_i)$, where each $\mathbf{F}_p[X]/(f_i)$ is a field. For each i, the subring $\{y \in \mathbf{F}_p[X]/(f_i) : y^p = y\}$ equals the prime field $\mathbf{F}_p$, so one has $\{y \in \mathbf{F}_p[X]/(f) : y^p = y\} \cong \prod_{i=1}^{t} \mathbf{F}_p$. In particular, f is irreducible if and only if $\{y \in \mathbf{F}_p[X]/(f) : y^p = y\}$ has dimension 1 as a vector space over $\mathbf{F}_p$; more generally, if h is a non-constant factor of f, then h is irreducible if and only if all $y \in \mathbf{F}_p[X]/(f)$ with $y^p = y$ reduce to a constant mod h.

To exploit these facts, Berlekamp's algorithm starts by finding a basis g_1, g_2, ..., g_t of the $\mathbf{F}_p$-vector space $\{y \in \mathbf{F}_p[X]/(f) : y^p = y\}$. The latter space is the null-space of the linear map $\mathbf{F}_p[X]/(f) \to \mathbf{F}_p[X]/(f)$ sending y to $y^p - y$, and a basis of this null-space can be computed by means of linear algebra. Next, the algorithm keeps track of a non-constant factor h of f, starting with $h = f$, stopping when h is irreducible, and replacing h by a proper factor otherwise. This is done in the following manner.

If all g_i are congruent to a constant modulo h, then h is irreducible, and one stops. Otherwise, choose i such that g_i is not congruent to a constant modulo h. Then h divides $g^p - g$, which equals the product $\prod_{j \in \mathbf{F}_p}(g_i - j)$, but h does not divide any of the factors $g_i - j$. Hence, computing at most $p-1$ greatest common divisors by means of the Euclidean algorithm, one finds $j \in \mathbf{F}_p$ with $0 < \deg \gcd(h, g_i - j) < \deg h$. Now replace h by $\gcd(h, g_i - j)$, and iterate. This finishes the description of Berlekamp's algorithm. One checks in a straightforward way that it has the properties claimed.

For more information on factoring polynomials over finite fields, including the description of a probabilistic algorithm with polynomial expected run time, one may consult [30, Chapter 14].

An auxiliary result. We prove a result that will be useful in proving the correctness of the factoring algorithm to be described. For a polynomial $e = \sum_i a_i X^i \in \mathbf{Z}[X]$, write $q(e) = \sum_i a_i^2$ and $\|e\| = q(e)^{1/2}$. For each positive integer n, write $\mathbf{Z}[X]_n$ for the set of polynomials in $\mathbf{Z}[X]$ of degree smaller than n; each $\mathbf{Z}[X]_n$ is, with the function q, a lattice of rank n and determinant 1.

Proposition. *Let m be a positive integer, and let $h \in \mathbf{Z}[X]$ be a monic polynomial. Let f, g be non-zero elements of the $\mathbf{Z}[X]$-ideal (m, h) generated by m and h, and suppose that we have*

$$\|f\|^{\deg g} \cdot \|g\|^{\deg f} < m^{\deg h}, \qquad \deg f + \deg g \geq \deg h.$$

Then f and g have a common factor of positive degree in $\mathbf{Z}[X]$.

Proof. First suppose that the only pair of polynomials $\lambda \in \mathbf{Z}[X]_{\deg g}$, $\mu \in \mathbf{Z}[X]_{\deg f}$ with $\lambda f + \mu g = 0$ is given by $\lambda = \mu = 0$. Then the set $M = \{\lambda f + \mu g : \lambda \in \mathbf{Z}[X]_{\deg g},$

$\mu \in \mathbf{Z}[X]_{\deg f}\}$ is a sublattice of $\mathbf{Z}[X]_{\deg f+\deg g}$ of rank $\deg f + \deg g$, with basis f, Xf, $\ldots$, $X^{\deg g-1}f$, g, Xg, $\ldots$, $X^{\deg f-1}g$. By Hadamard's inequality, one has $d(M) \leq \|f\|^{\deg g} \cdot \|g\|^{\deg f}$. From f, $g \in (m,h)$ it follows that M is contained in $L = (m,h) \cap \mathbf{Z}[X]_{\deg f+\deg g}$, which is also a sublattice of $\mathbf{Z}[X]_{\deg f+\deg g}$. From $\deg f + \deg g \geq \deg h$ it follows that $(m,h) + \mathbf{Z}[X]_{\deg f+\deg g} = \mathbf{Z}[X]$, and therefore

$$d(L) = \#\mathbf{Z}[X]_{\deg f+\deg g}/L = \#\mathbf{Z}[X]/(m,h) = m^{\deg h}.$$

Altogether we obtain

$$\|f\|^{\deg g} \cdot \|g\|^{\deg f} \geq d(M) = (L : M) \cdot d(L) \geq m^{\deg h},$$

contradicting our hypothesis. Thus, there do exist non-zero polynomials $\lambda \in \mathbf{Z}[X]_{\deg g}$ and $\mu \in \mathbf{Z}[X]_{\deg f}$ with $\lambda f = -\mu g$. This implies that f and g have a common factor of positive degree in $\mathbf{Z}[X]$, as required.

Factoring polynomials. We describe a polynomial-time algorithm that, given a non-constant polynomial $f \in \mathbf{Q}[X]$, finds the factorization of f into irreducible factors in $\mathbf{Q}[X]$. Our description assumes that the discriminant $\Delta(f)$ of f is non-zero, and that the coefficients of f are in $\mathbf{Z}$; to achieve the first, one replaces f by $f/\gcd(f, \mathrm{d}f/\mathrm{d}X)$, and to achieve the second one multiplies the coefficients by a common denominator. We let $n = \deg f$.

(a) *Choose an auxiliary prime number.* Compute the least prime number p not dividing the resultant $R(f, \mathrm{d}f/\mathrm{d}X)$ of f and its derivative. Since $\pm R(f, \mathrm{d}f/\mathrm{d}X)$ equals the product of the leading coefficient and the discriminant of f, the polynomial $(f \bmod p) \in \mathbf{F}_p[X]$ has degree n and non-zero discriminant.

(b) *Find an irreducible factor mod p.* Apply Berlekamp's algorithm, as described above, to $(f \bmod p)$ divided by its leading coefficient. This leads to a monic irreducible factor $h_0 \in \mathbf{F}_p[X]$ of $(f \bmod p)$. If $\deg h_0 = \deg f$, then f is irreducible in $\mathbf{Q}[X]$, and the algorithm stops. Assume now $\deg h_0 < \deg f$.

(c) *Determine the p-adic precision needed.* Compute the least integer μ with

$$p^{2\mu \deg h_0} > 2^{n(n-1)} \cdot \binom{2(n-1)}{n-1}^n \cdot q(f)^{2n-1}.$$

(d) *Find an approximate p-adic factor of f.* Use Hensel's algorithm, as described in [5], to find a monic polynomial $h \in \mathbf{Z}[X]$ such that $h_0 = (h \bmod p)$ and such that $(h \bmod p^\mu)$ divides $(f \bmod p^\mu)$ in $(\mathbf{Z}/p^\mu\mathbf{Z})[X]$; by Hensel's lemma and the fact that $\Delta(f) \not\equiv 0 \bmod p$,

the polynomial h exists and is unique modulo p^μ. (*Note.* A formal zero of h may be viewed as an approximate p-adic zero of f; so the computation of h corresponds to the computation of β in the summary description provided earlier.)

(e) *Apply lattice basis reduction.* Define L to be the additive subgroup of $\mathbf{Z}[X]$ that has basis

$$p^\mu,\ p^\mu \cdot X,\ \ldots,\ p^\mu \cdot X^{(\deg h)-1},\ h,\ X \cdot h,\ \ldots,\ X^{n-1-\deg h} \cdot h.$$

Viewing L as a sublattice of the lattice $\mathbf{Z}[X]_n$ defined above, apply the LLL algorithm to find a 2-reduced basis $b_1, \ldots, b_n$ for L. (*Note.* The elements of L are the polynomials of degree smaller than n that assume p-adically small values at a zero β of h, so they provide approximate linear dependencies among $1, \beta, \ldots, \beta^{n-1}$.)

(f) *Decide irreducibility or find a factor.* If

$$q(b_1) > 2^{n-1} \cdot \binom{2(n-1)}{n-1} \cdot q(f),$$

declare f irreducible and stop. Otherwise, compute $g = \gcd(b_1, f)$ using the Euclidean algorithm in $\mathbf{Q}[X]$. Multiplying g by a suitable scalar, we may assume that the coefficients of g are in $\mathbf{Z}$ and generate the unit ideal of $\mathbf{Z}$. Factor g and f/g recursively into irreducible factors in $\mathbf{Q}[X]$, and combine their factorizations into the factorization of f.

Correctness of the algorithm. The proof that the algorithm, as described, runs in polynomial time, is largely routine. The only point worth emphasizing is that, by a very weak form of the prime number theorem, the prime number p chosen in (a) is small enough for Berlekamp's algorithm to run in time polynomial in the length of the input data for our factoring algorithm. For more details on the run time analysis one may consult the original article [21].

The correctness of the algorithm, in particular of step (f), follows from the equivalence of the following statements: (i) f is reducible; (ii) we have

$$q(b_1) \le 2^{n-1} \cdot \binom{2(n-1)}{n-1} \cdot q(f);$$

(iii) f and b_1 have a common factor of positive degree in $\mathbf{Z}[X]$. The implication (iii)⇒(i) follows from $\deg b_1 < n = \deg f$. To prove (i)⇒(ii), denote by g the irreducible factor of f in $\mathbf{Z}[X]$ for which h_0 divides $(g \bmod p)$; from $\Delta(f) \not\equiv 0 \bmod p$ it follows that g exists and is unique up to sign. By Hensel's lemma, $(h \bmod p^\mu)$ divides $(g \bmod p^\mu)$ in $(\mathbf{Z}/p^\mu\mathbf{Z})[X]$. Also, if we assume (i), then we have $\deg g < n$, and therefore $g \in L$. A very general inequality of

Mignotte [25] on factors of polynomials implies $q(g) \leq \binom{2 \deg g}{\deg g} \cdot q(f) \leq \binom{2(n-1)}{n-1} \cdot q(f)$. Since $b_1, \ldots, b_n$ is a 2-reduced basis for L (see the end of Section 11), we have $q(b_1) \leq 2^{n-1} \cdot q(g)$, which leads to (ii). Finally, the inequalities in (ii) and (c) imply that the conditions of the Proposition are satisfied for $m = p^\mu$ and $g = b_1$, and this leads to a proof of (ii)$\Rightarrow$(iii).

Global fields. The factoring algorithm in $\mathbf{Q}[X]$ described above admits a generalization to $K[X_1, \ldots, X_t]$, for any global field K and any positive integer t. A significant special case is treated in [20] by means of a different notion of lattice, as defined in Section 16 below. For a good general discussion with references, see [30, Chapters 15 and 16].

Van Hoeij's algorithm. The reader may have noticed that, for practical purposes, the factoring algorithm as described allows many improvements. There is no need to care about these, since in virtually all practical situations there are other algorithms with a better performance. The chief one among these is *Van Hoeij's algorithm*, which applies lattice basis reduction in an altogether different manner. We sketch the basic idea, without paying attention to refinements of practical value.

Let $f \in \mathbf{Z}[X]$ be the monic polynomial to be factored in irreducible factors in $\mathbf{Q}[X]$ or, equivalently, in $\mathbf{Z}[X]$. Put $n = \deg f$. As in the previous algorithm, one starts by choosing a prime p of $\mathbf{Q}$, but next, instead of finding a good approximation to a single p-adic zero α of f, one finds good approximations $\beta_1, \ldots, \beta_n$ to *all* zeroes $\alpha_1, \ldots, \alpha_n$ of f in a suitable finite extension K of the completion $\mathbf{Q}_p$ of $\mathbf{Q}$ at p. These approximations are found by means of techniques from classical or p-adic numerical analysis. Every monic factor g of f is of the form $\prod_{i \in I}(X - \alpha_i)$ for some subset $I \subset \{1, 2, \ldots, n\}$, and for g to have coefficients in $\mathbf{Z}$ it is necessary that $\sum_{i \in I} \alpha_i$, $\sum_{i \in I} \alpha_i^2$, $\ldots$, are in $\mathbf{Z}$, and hence that $\sum_{i \in I} \beta_i$, $\sum_{i \in I} \beta_i^2$, $\ldots$, are p-adically very close to elements of $\mathbf{Z}$. Thus, Van Hoeij's algorithm proceeds by choosing a positive integer m and searching for an integer vector $(k_i)_{i=1}^n$ with the property that each of $\sum_{i=1}^n k_i \beta_i$, $\sum_{i=1}^n k_i \beta_i^2$, $\ldots$, $\sum_{i=1}^n k_i \beta_i^m$ is very close to an integer. This can be done by means of lattice basis reduction, the construction of the lattice being similar to the constructions shown in Section 7. If the only vectors that one finds have all k_i equal, then one declares f to be irreducible; if not all k_i are equal, then for each k that occurs among the k_i one computes $\prod_{i,\, k_i = k}(X - \beta_i)$, and one hopes to be able to round its coefficients to integers and obtain a non-trivial factor of f. Using different vectors $(k_i)_{i=1}^n$ one may even hope to find the full factorization of f into irreducible factors in $\mathbf{Z}[X]$ in this way. This strategy often works for very small values of m, such as $m = 1$ or 2. If it doesn't work, then one increases the value of m and tries again.

Van Hoeij's algorithm presents a number of interesting mathematical problems. The first is to give a version that can be rigorously analyzed and that runs in polynomial time. The second is to extend the algorithm from $\mathbf{Q}[X]$ to $K[X]$, for any global field K, including the case of positive characteristic. Neither of these problems is trivial, but they do admit solutions, see [2]. The solution to the first problem uses an unrealistically large value for m, namely $m = n - 1$. One may wonder whether smaller values of m can be proved to work in all cases.

14. Linear algebra over the ring of integers

Lattice basis reduction is useful in solving linear algebra problems over $\mathbf{Z}$. Examples of such problems are: given an $m \times n$ matrix $\mathbf{F}$ with integral entries, find bases both for the kernel and for the image of the group homomorphism $\mathbf{Z}^n \to \mathbf{Z}^m$ mapping $x \in \mathbf{Z}^n$ to $\mathbf{F} \cdot x \in \mathbf{Z}^m$; and given such a matrix $\mathbf{F}$, and $b \in \mathbf{Z}^m$, determine all $x \in \mathbf{Z}^n$ with $\mathbf{F} \cdot x = b$.

The problems that we shall consider are purely linear, and their formulation does not refer to a lattice structure. Lattices are nevertheless useful in their solution, because they provide a natural way of coping with a difficulty that the more traditional approach, which depends on the *Hermite normal form* of an integer matrix (see [7, Section 2.4]) runs into. The straightforward algorithm for computing the Hermite normal form (see [7, Algorithm 2.4.4]) suffers from serious coefficient blow-up, and is therefore not expected to run in polynomial time. Preventing coefficient blow-up is tantamount to controlling the Euclidean length of the vectors that one works with, and that is what lattice algorithms are designed to do.

We shall in this section have occasion to endow groups of the form $\mathbf{Z}^k$, with k a non-negative integer, with several different lattice structures; the notation $\| \ \|^2$ will always be reserved for the standard lattice structure, defined by $\|x\|^2 = \sum_{i=1}^{k} x_i^2$ for $x = (x_i)_{i=1}^{k} \in \mathbf{Z}^k$.

Kernels, images, and reduced bases. Let n and m be non-negative integers, and let $f\colon \mathbf{Z}^n \to \mathbf{Z}^m$ be a group homomorphism. Denote by $\mathbf{F}$ the $m \times n$ matrix over $\mathbf{Z}$ with the property that for all $x \in \mathbf{Z}^n$ one has $f(x) = \mathbf{F} \cdot x$; so the columns of $\mathbf{F}$ are the images of the standard basis vectors of $\mathbf{Z}^n$ under f. The following result shows how one can define a lattice with the property that bases for the kernel and the image of f can be read off from a reduced basis for the lattice.

Proposition. *Let n, m, f, $\mathbf{F}$ be as above, and write r for the rank of $\mathbf{F}$. Let F be a real number such that the absolute value of any entry of $\mathbf{F}$ is at most F, and let c and N be*

real numbers with

$$c \geq 4/3, \qquad N > c^{n-1} \cdot (r+1) \cdot r^r \cdot F^{2r}.$$

Let the lattice L, q be defined by $L = \mathbf{Z}^n$ and

$$q(x) = \|x\|^2 + N \cdot \|f(x)\|^2 \qquad \text{for } x \in \mathbf{Z}^n,$$

and let $b_1, \ldots, b_n$ be a c-reduced basis for this lattice. Then we have:

(a) $q(b_i) < N$ *for* $1 \leq i \leq n - r$;

(b) $b_1, \ldots, b_{n-r}$ *form a basis for* $\ker f$ *over* $\mathbf{Z}$;

(c) $q(b_i) \geq N$ *for* $n - r < i \leq n$;

(d) $f(b_{n-r+1}), \ldots, f(b_n)$ *form a basis for* $f(\mathbf{Z}^n)$ *over* $\mathbf{Z}$.

Proof. For notational convenience we may assume that the standard basis vectors of $\mathbf{Z}^n$ are numbered in such a way that the first r columns of $\mathbf{F}$ are linearly independent. Let $r < h \leq n$. By Cramer's rule, there is a non-trivial linear dependency among the first r columns and the hth column of $\mathbf{F}$, with coefficients that are $r \times r$ minors of $\mathbf{F}$. This dependency gives rise to an element $x = (x_i)_{i=1}^n$ of $\ker f$ with $x_h \neq 0$ and $x_i = 0$ for all $i > r$ with $i \neq h$. By Hadamard's inequality we have $|x_i| \leq r^{r/2} F^r$ for all i, and therefore $q(x) = \|x\|^2 \leq (r+1) \cdot r^r \cdot F^{2r}$. The $n - r$ vectors obtained in this way for $h = r+1$, $\ldots$, n are linearly independent, so for each $i \leq n - r$ the ith successive minimum $\lambda_i(L)$, as defined in Section 10, satisfies $\lambda_i(L) \leq (r+1) \cdot r^r \cdot F^{2r}$. By the Proposition in Section 10, we now have

$$q(b_i) \leq c^{n-1} \cdot \lambda_i(L) \leq c^{n-1} \cdot (r+1) \cdot r^r \cdot F^{2r} < N \quad \text{for } i \leq n - r.$$

This proves (a). The definition of q implies that every $x \in L$ with $q(x) < N$ belongs to $\ker f$. Thus, from (a) we see that $\ker f$ contains the linearly independent vectors $b_1, \ldots, b_{n-r}$. By linear algebra, the null space of $\mathbf{F}$ on $\mathbf{Q}^n$ has $\mathbf{Q}$-dimension equal to $n - r$ and is therefore spanned by $b_1, \ldots, b_{n-r}$. Consequently, inside $\mathbf{Q}^n$ we have

$$\ker f = \Big(\sum_{i=1}^{n-r} \mathbf{Q} b_i \Big) \cap \mathbf{Z}^n = \sum_{i=1}^{n-r} \mathbf{Z} b_i,$$

the latter equality because $b_1, \ldots, b_n$ form a basis for $\mathbf{Z}^n$ over $\mathbf{Z}$. This proves (b). It follows that for each $i > n - r$ we have $b_i \notin \ker f$ and therefore $q(b_i) \geq N$, which is (c). Finally, (d) follows from (b) and the homomorphism theorem from elementary group theory. This proves the Proposition.

The kernel and image algorithm. We describe an algorithm that, given non-negative integers n and m and a group homomorphism $f\colon \mathbf{Z}^n \to \mathbf{Z}^m$, determines the kernel and the image of f. Here f is specified by an $m \times n$ matrix $\mathbf{F}$ over $\mathbf{Z}$, as above. The kernel of f is required to be specified by a sequence of vectors in $\mathbf{Z}^n$ that form a basis for $\ker f$ over $\mathbf{Z}$, and likewise for the image of f in $\mathbf{Z}^m$.

One starts by defining F to be the maximum of the absolute values of the entries of $\mathbf{F}$, with $F = 0$ if $nm = 0$. One chooses $c = 2$, and one chooses N to be an integer exceeding $2^{n-1} \cdot (r+1) \cdot r^r \cdot F^{2r}$, where r denotes the rank of $\mathbf{F}$; if the value of r is not known, one just uses the upper bound $r \leq \min\{n, m\}$. Next, one applies the LLL algorithm to find a c-reduced basis $b_1, \ldots, b_n$ for L. By the Proposition, the b_i with $q(b_i) < N$ form a basis for $\ker f$, and the images of the other b_i under f form a basis for the image of f. This completes the description of the algorithm. With a proper choice of N, this algorithm is readily shown to run in polynomial time.

Ordered vector spaces. We discuss a modification of the algorithm just described that both improves its practical performance and has theoretical interest. The modification consists of not choosing an actual value for N, but viewing it as an 'indefinitely large' symbol. More rigorously, one redefines the function q on L by $q(x) = (\|x\|^2, \|f(x)\|^2)$; its values are not in $\mathbf{R}$, but in the real vector space $\mathbf{R} \times \mathbf{R}$, which one endows with a total ordering by putting $(r_1, r_2) > (s_1, s_2)$ if and only if either $r_2 > s_2$, or $r_2 = s_2$ and $r_1 > s_1$ (the *anti-lexicographic ordering*). To capture the structure L, q defined in this manner in a theoretical framework, one is led to define a generalized notion of Euclidean vector space, in which the real-valued inner product $\langle\ ,\ \rangle$ defined on $E \times E$, as considered in Section 2, is replaced by one that takes values in a totally ordered real vector space; in addition to the axioms from Section 2, one requires that for any x, $y \in E$ there exists $r \in \mathbf{R}$ with $\langle x, y\rangle \leq r\langle x, x\rangle$. It appears to be both worthwhile and feasible to define a correspondingly generalized notion of lattice, and to formulate conditions under which a natural extension of the LLL algorithm terminates in polynomial time. This theory, yet to be developed, should confirm that the modified kernel and image algorithm, and similar algorithms to be discussed below, run in polynomial time. The implications for diophantine approximation, where large weights N are also encountered (see Section 7), are worth exploring as well.

Solving a system of linear equations over $\mathbf{Z}$. Let m and n be non-negative integers, let $\mathbf{F}$ be an $m \times n$ matrix over $\mathbf{Z}$, and let $b \in \mathbf{Z}^m$. We are interested in finding all $x \in \mathbf{Z}^n$ with $\mathbf{F} \cdot x = b$.

Define the group homomorphisms $g\colon \mathbf{Z}^n \times \mathbf{Z} = \mathbf{Z}^{n+1} \to \mathbf{Z}^m$ and $h\colon \mathbf{Z}^n \times \mathbf{Z} \to \mathbf{Z}$ by $g(x,z) = \mathbf{F} \cdot x - z \cdot b$ and $h(x,z) = z$, for $x \in \mathbf{Z}^n$, $z \in \mathbf{Z}$. Clearly, there exists $x \in \mathbf{Z}^n$ with $\mathbf{F} \cdot x = b$ if and only if 1 belongs to the image under h of the kernel of g. Thus, one can decide whether the equation $\mathbf{F} \cdot x = b$ is solvable with $x \in \mathbf{Z}^n$ by performing the kernel and image algorithm twice. Actually, a single application of the LLL algorithm suffices, and the resulting algorithm does not only decide solvability, but in fact describes the set of all solutions. It runs as follows.

Let N and M be suitably chosen large integers with $N \gg M$, and make the group $L = \mathbf{Z}^n \times \mathbf{Z}$ into a lattice by putting

$$q(x,z) = \|x\|^2 + M \cdot z^2 + N \cdot \|\mathbf{F} \cdot x - z \cdot b\|^2 \qquad \text{for } x \in \mathbf{Z}^n,\ z \in \mathbf{Z}.$$

Use the LLL algorithm to determine a 2-reduced basis $b_1, \ldots, b_{n+1}$ for L. Then $\mathbf{F} \cdot x = b$ has a solution $x \in \mathbf{Z}^n$ if and only if there exists an index j with $M \leq q(b_j) < 4M$; moreover, if such an index exists, then it is unique, and the following is valid: each b_i with $i < j$ is of the form $(b_i', 0)$ with $b_i' \in \mathbf{Z}^n$, the z-coordinate of b_j equals ± 1, and if $x_0 \in \mathbf{Z}^n$ is defined by $\pm b_j = (x_0, 1)$, then $x = x_0$ is a solution to $\mathbf{F} \cdot x = b$, whereas the general solution is given by $x = x_0 + \sum_{i=1}^{j-1} k_i b_i'$ with $k_1, \ldots, k_{j-1} \in \mathbf{Z}$.

One can show that the assertions just made are correct if $M > 2^n \cdot (r+1) \cdot r^r \cdot F^{2r}$ and $N > 2^n \cdot (r+M) \cdot r^r \cdot F^{2r}$, where r equals the rank of $\mathbf{F}$ and $F \in \mathbf{Z}$ is an upper bound for the absolute values of all entries of $\mathbf{F}$ and b. As a consequence, one obtains a polynomial-time algorithm for solving $\mathbf{F} \cdot x = b$ over $\mathbf{Z}$. Alternatively, one may redefine q to take values in the anti-lexicographically ordered real vector space $\mathbf{R} \times \mathbf{R} \times \mathbf{R}$, by putting $q(x,z) = \big(\|x\|^2, \|z\|^2, \|\mathbf{F} \cdot x - z \cdot b\|^2\big)$, and invoke the generalized algorithmic theory of lattices alluded to above.

The Chinese remainder theorem. Suppose one is given a positive integer k, a sequence $m_1, \ldots, m_k$ of pairwise coprime positive integers, as well as a sequence $r_1, \ldots, r_k$ of integers, and that one is interested in finding an integer x satisfying the k congruences $x \equiv r_i \bmod m_i$ $(1 \leq i \leq k)$. The problem is equivalent to finding a vector $(x, y_1, \ldots, y_k) \in \mathbf{Z}^{k+1}$ satisfying the system of linear equations $x - y_i m_i = r_i$ $(1 \leq i \leq k)$, and can thus be solved in polynomial time by the linear algebra algorithm just explained. There is also a more direct approach (see [18, Section 4.3.2]), and the reader is invited to make a comparison of run times.

The generalized extended Euclidean algorithm. We revisit a problem considered earlier. Let, slightly more generally than in Section 8, a non-negative integer k as well as integers a_1,

$\ldots$, a_k be given; we want to compute an integer d with $\sum_{i=1}^{k} \mathbf{Z}a_i = \mathbf{Z}d$, as well as 'small' integers x_1, $\ldots$, x_k with $\sum_{i=1}^{k} x_i a_i = d$.

As in the linear algebra problem just considered, let N and M be suitably large positive integers with $N \gg M$, and make the group $\mathbf{Z}^{k+1}$ into a lattice by putting

$$q(x_1, \ldots, x_{k+1}) = \Big(\sum_{i=1}^{k} x_i^2\Big) + M \cdot x_{k+1}^2 + N \cdot \Big(x_{k+1} - \sum_{i=1}^{k} x_i a_i\Big)^2.$$

Let $b_1, \ldots, b_{k+1}$ be a 2-reduced basis for this lattice. If there is an index j with $M \leq q(b_j) < N$, and $b_j = (x_i)_{i=1}^{k+1}$, then for $d = x_{k+1}$ one has $\sum_{i=1}^{k} \mathbf{Z}a_i = \mathbf{Z}d$ and $\sum_{i=1}^{k} x_i a_i = d$. If no such index j exists, then all a_i are 0, and one can take d and all x_i to be 0 as well. The details, and the proof that the resulting algorithm runs in polynomial time, may again be left to the reader.

The nearest vector problem. The problem that we just discussed, was in Section 8 identified as a special case of the nearest vector problem. The general nearest vector problem admits a similarly direct solution by means of lattice basis reduction. Namely, suppose one is given a lattice L in a Euclidean vector space E, as well as an element $x \in E$, and that one wants to find $y \in L$ with $q(x - y)$ small. Define a lattice L', q' by putting $L' = L \times \mathbf{Z}$ and $q'(y, z) = q(y - zx) + N \cdot z^2$ for $y \in L$, $z \in \mathbf{Z}$, where again N is chosen large enough or indefinitely large. Only the last basis vector of a c-reduced basis b_1, $\ldots$, $b_{\mathrm{rk}\, L'}$ for L' will then have a non-zero z-coordinate, and that z-coordinate will be ± 1; if $\pm b_{\mathrm{rk}\, L'} = (y, 1)$, with $y \in L$, then y is a 'good' solution to the nearest vector problem. This solution is essentially the same as the one constructed in Section 10.

Operations on subgroups. Let n be a non-negative integer. The kernel and image algorithm can be used to perform several operations on subgroups of $\mathbf{Z}^n$. We give a number of examples; it is always assumed that, for algorithmic purposes, a subgroup $H \subset \mathbf{Z}^n$ is specified by means of a sequence of elements of $\mathbf{Z}^n$ that is a basis for H over $\mathbf{Z}$. All algorithms to be described run in polynomial time, n being viewed as part of the input.

Let H_1 and H_2 be two subgroups of $\mathbf{Z}^n$, and consider the group homomorphism $H_1 \times H_2 \to \mathbf{Z}^n$ sending (x, y) to $x - y$. Its image is the subgroup $H_1 + H_2$ of $\mathbf{Z}^n$, and its kernel can in an obvious manner be identified with $H_1 \cap H_2$. Thus, from the kernel and image algorithm one obtains bases for both $H_1 + H_2$ and $H_1 \cap H_2$ over $\mathbf{Z}$. In fact, in the case of $H_1 \cap H_2$, one obtains *three* expressions for the same basis: one in terms of the given basis for H_1, one in terms of the given basis for H_2, and one in terms of the standard basis for $\mathbf{Z}^n$.

Let H be a subgroup of $\mathbf{Z}^n$, and let $\mathbf{F}$ be a $n \times (\operatorname{rk} H)$ matrix over $\mathbf{Z}$ of which the columns form a basis for H over $\mathbf{Z}$. The *transpose* of $\mathbf{F}$ may be viewed as the matrix that describes the map $\varphi\colon \mathbf{Z}^n \to \operatorname{Hom}(H, \mathbf{Z})$ defined by $\varphi(x)(y) = \langle x, y\rangle$ for $x \in \mathbf{Z}^n$, $y \in H$, where $\langle\ ,\ \rangle$ denotes the standard inner product on $\mathbf{Z}^n$. Applying the kernel and image algorithm, one obtains a basis for $H^\perp = \ker \varphi = \{x \in \mathbf{Z}^n : \langle x, y\rangle = 0 \text{ for all } y \in H\}$. Doing this again, one obtains a basis for $H^{\perp\perp}$, which equals the subgroup $(\mathbf{Q} \cdot H) \cap \mathbf{Z}^n$ of $\mathbf{Z}^n$. Simultaneously, one obtains a basis for $\mathbf{Z}^n/H^{\perp\perp}$, which may be identified with the group $\mathbf{Z}^n/H$ modulo its torsion subgroup.

Define the *degree* $\deg x$ of a non-zero vector $x = (x_i)_{i=1}^n \in \mathbf{Z}^n$ to be $\max\{i : x_i \neq 0\}$. It is well-known that any subgroup $H \subset \mathbf{Z}^n$ has a basis $b_1, \ldots, b_{\operatorname{rk} H}$ with the property that $\deg b_i$ is strictly increasing as a function of i. To compute such a basis from a given basis for H, it suffices to apply lattice basis reduction to the lattice H, q, where q is defined by

$$q(x_1, \ldots, x_n) = \sum_{i=1}^{n} N_i x_i^2,$$

for suitable integers N_i with $N_n \gg N_{n-1} \gg \ldots \gg N_2 \gg N_1 = 1$; again, the formalism involving ordered vector spaces would be applicable here. The same technique can be used to compute the Hermite normal form of an integer matrix by means of lattice basis reduction.

I do not know whether lattice basis reduction algorithms may assist in computing the *Smith normal form* of an integer matrix (see [7, Section 2.4.4]), or how useful they are in doing computations with finitely generated abelian groups that are allowed to have torsion.

15. Nonlinear problems

In Section 13 we saw that lattices can be used to solve the nonlinear problem of factoring in the ring $\mathbf{Q}[X]$. There is in fact a surprisingly large class of nonlinear problems that can be solved by means of lattices. In the present section we describe a general technique, and we illustrate it with three examples. Related methods are well-known in the area of diophantine approximation, where they are used to prove upper bounds for the number of integral solutions to certain systems of equations that satisfy certain inequalities (see [17]). It is a more recent insight that in many cases these solutions can be efficiently enumerated by means of lattice basis reduction. One may consult [3] for a different perspective, for references, and for a historical discussion, and [10] for an account of a very similar technique, with additional applications.

Let V be an affine algebraic set defined over $\mathbf{R}$, embedded in affine t-space $\mathbf{A}^t_{\mathbf{R}}$, for some non-negative integer t; so the coordinate ring $\mathbf{R}[V]$ equals $\mathbf{R}[X_1, \ldots, X_t]/I$ for some ideal I of the polynomial ring $\mathbf{R}[X_1, \ldots, X_t]$. The set $V(\mathbf{R})$ of real points of V is defined by $\{x \in \mathbf{R}^t : f(x) = 0 \text{ for all } f \in I\}$. By abuse of notation, we write $V(\mathbf{Z}) = V(\mathbf{R}) \cap \mathbf{Z}^t$. Suppose in addition that B is a subset of $\mathbf{R}^t$ for which $B \cap V(\mathbf{R})$ is bounded. Then the set $S = B \cap V(\mathbf{Z})$ is finite. We assume that one is interested in determining upper bounds for $\#S$ and, if I and B are given in some explicit manner, in algorithms for listing all elements of S.

The lattice-based technique that applies in this context, produces a non-zero element $g \in \mathbf{R}[V]$ that vanishes on S, so that S remains unchanged if V is replaced by the affine algebraic set W defined by $\mathbf{R}[W] = \mathbf{R}[V]/(g)$, which can in principle be dealt with recursively.

In many situations of interest, the variety V is an irreducible curve. In that case, the zero set of g on V, which contains S, is finite; the lattice method gives an upper bound for its cardinality, and in algorithmic circumstances it is usually easy to first compute all zeroes of g in $V(\mathbf{Z})$ and next check them one by one for membership of S.

Examples. Rather than attempting to formulate general conditions under which the technique is useful, we describe three problems from algorithmic number theory to which it has been successfully applied. In each case, the efficiency of the resulting algorithm is contingent upon inequalities satisfied by the problem parameters.

(a) *Zeroes of polynomials modulo n.* Suppose one is given integers a, b, and n with $a < b$ and $n > 0$, as well as a monic polynomial $p \in \mathbf{Z}[X]$, and that one is interested in the set of all $x \in \mathbf{Z}$ with $a \leq x \leq b$ and $p(x) \equiv 0 \bmod n$. Then one can take $t = 2$, and V to be the algebraic subset of real affine 2-space defined by the equation $p(x) = n \cdot y$; that is, one has $\mathbf{R}[V] = \mathbf{R}[X, Y]/(p - nY)$. Note that the natural map $\mathbf{R}[X] \to \mathbf{R}[V]$ is a ring isomorphism, so that V is actually isomorphic to the affine line over $\mathbf{R}$, which is an irreducible curve. With $B = \{(x, y) \in \mathbf{R}^2 : a \leq x \leq b\}$, the set $S = B \cap V(\mathbf{Z})$ defined above maps bijectively to the set $\{x \in \mathbf{Z} : a \leq x \leq b,\ p(x) \equiv 0 \bmod n\}$ that one is interested in, by the projection map $(x, y) \mapsto x$.

(b) *Divisors in residue classes.* Suppose one is given positive integers u, v, and n with $\gcd(u, v) = 1$, and that one is interested in the set of divisors x of n that satisfy $x \equiv u \bmod v$. In this case, one can take $t = 3$ and define V by $xy = n$, $x = u + vz$. Then one has $\mathbf{R}[V] = \mathbf{R}[X, Y, Z]/(XY - n, X - u - vZ)$, and there is an $\mathbf{R}$-algebra isomorphism from the ring $\mathbf{R}[X, X^{-1}]$ of Laurent polynomials in X over $\mathbf{R}$ to the ring $\mathbf{R}[V]$ that maps

X to X and X^{-1} to Y/n. Hence, V is isomorphic to the affine line with a single point removed, which is again an irreducible curve. With $B = \{(x, y, z) \in \mathbf{R}^3 : 1 \leq x \leq n\}$, the set $S = B \cap V(\mathbf{Z})$ may again be identified with the set one is interested in.

(c) *Diophantine approximation with restricted denominators.* Let α be a real number and let n be a positive integer. We suppose that one is interested in 'good' rational approximations y/z to α, with y, $z \in \mathbf{Z}$, $z > 0$, of which the denominator z is 'small' and satisfies the additional restriction that it divide n. Denote by $[a/n, b/n]$ the interval around α that one wishes y/z to belong to, with the endpoints properly rounded to integer multiples of $1/n$, so that a, $b \in \mathbf{Z}$, $a < b$. We shall always assume $b - a < n$, since otherwise the interval $[a/n, b/n]$ contains rational umbers with any given denominator. Write m for the desired upper bound on z. We can now take $t = 3$, define the surface V by $xz = ny$, and put $B = \{(x, y, z) \in \mathbf{R}^3 : a \leq x \leq b,\ 1 \leq z \leq m\}$. One has $\mathbf{R}[V] = \mathbf{R}[X, Y, Z]/(XZ - nY)$, and the natural map $\mathbf{R}[X, Z] \to \mathbf{R}[V]$ is an isomorphism. The set $S = B \cap V(\mathbf{Z})$ maps bijectively to the set one is interested in, by $(x, y, z) \mapsto y/z$.

If two distinct rational numbers in $[a/n, b/n]$ each have denominator at most m, then their difference is a non-zero rational number of absolute value at most $(b - a)/n$ with denominator at most m^2, so that $(b-a)/n \geq 1/m^2$. Thus, for $m < \sqrt{n/(b-a)}$ the number y/z is unique if it exists. One can find it using continued fractions or two-dimensional lattice basis reduction, as in Section 7. This approach, however, disregards the requirement that z divide n. The approach of the present section does take that requirement into account, and it allows larger values for m to be taken. More specifically, if ϵ is such that $b - a = n^{\epsilon}$, then Proposition C below shows that instead of $m < \sqrt{n/(b-a)} = n^{(1-\epsilon)/2}$ we can allow $m < n^{\eta}$ for any $\eta < 1 - \sqrt{\epsilon}$; note that one has $(1 - \epsilon)/2 < 1 - \sqrt{\epsilon}$.

The equation $xz = ny$ defining V is homogeneous in y and z, so it may also be thought of as defining a curve V' in the product of the affine line $\mathbf{A}^1_{\mathbf{R}}$ parametrized by x and the projective line $\mathbf{P}^1_{\mathbf{R}}$ parametrized by $y : z$. One may then view V as a 'cone' over V', the 'top' of the cone being the line in $\mathbf{A}^3_{\mathbf{R}}$ defined by $y = z = 0$. We will be careful to construct the non-zero element $g \in \mathbf{R}[V]$ in such a way that it will likewise be homogeneous in Y and Z, so that $g = 0$ defines a finite set of points in V'.

The following result shows the relevance of lattices for the type of problem we are considering. Let the notations V, $\mathbf{R}[V]$, $V(\mathbf{R})$, $V(\mathbf{Z})$, B, S be as introduced at the beginning of this section.

Lemma. *Let L, q be a non-zero lattice and let c be a positive real number such that:*

(i) *the group L is a subgroup of the additive group of $\mathbf{R}[V]$ with the property that each $f \in L$ is integral-valued on $V(\mathbf{Z})$,*

(ii) *for each $x \in B \cap V(\mathbf{R})$ and each f in the $\mathbf{R}$-linear span of L, one has $|f(x)| \leq c \cdot q(f)^{1/2}$,*

(iii) *one has $c \cdot \sqrt{\operatorname{rk} L} \cdot d(L)^{1/\operatorname{rk} L} < 1$.*

Then there exists a non-zero element $g \in L$ such that for all $x \in S$ one has $g(x) = 0$.

Proof. By the theorem of Minkowski (Section 6), we can choose a non-zero element $g \in L$ with $q(g) \leq (\operatorname{rk} L) \cdot d(L)^{2/\operatorname{rk} L}$. Let $x \in S$. Applying (ii) to $f = g$ we obtain $|g(x)| \leq c \cdot \sqrt{\operatorname{rk} L} \cdot d(L)^{1/\operatorname{rk} L}$, so by (iii) we have $|g(x)| < 1$. Since by (i) we have $g(x) \in \mathbf{Z}$, we obtain $g(x) = 0$. This proves the Lemma.

In algorithmic circumstances, one replaces the theorem of Minkowski by a lattice basis reduction algorithm. This allows the actual construction of a non-zero element $g \in L$ that vanishes on S, provided that the condition (iii) is replaced by a slightly stronger one. Specifically, if one makes use of 2-reduced bases, then the factor $\sqrt{\operatorname{rk} L}$ in (iii) should be replaced by $2^{(\operatorname{rk} L - 1)/4}$.

The integrality condition (i) of the Lemma is satisfied if L is chosen inside the image of the ring $\mathbf{Z}[X_1, \ldots, X_t]$ in $\mathbf{R}[X_1, \ldots, X_t]/I = \mathbf{R}[V]$. (One can also use the ring of integral-valued polynomials, generated by $\{\binom{X_i}{j} : 1 \leq i \leq t,\ j \in \mathbf{Z}_{\geq 0}\}$.) Condition (ii) is, under weak conditions, probably automatic for *some* value of c; to keep c small, with an eye on (iii), one adapts the choice of q to the set B, as illustrated in the examples below. The inequality in (iii) expresses the condition under which the technique under discussion is useful.

Several strategies are available if (iii) is not satisfied. One strategy, which we shall follow in the proof of Proposition B below, is to cut up B into several pieces, each piece having its own L, q and a smaller value for c. Alternatively, one may decide to be satisfied with an element $g \in L$ with the weaker property that the zeroes of $g - i$ cover all of S when i ranges over all integers with $|i|$ below a certain bound; to avoid the possibility that one of these $g - i$ is identically zero (that is, $g = i$ in $\mathbf{R}[V]$), one may have to find a non-zero element in the lattice $L/(L \cap \mathbf{Z})$ instead of in L itself.

We return to our examples and illustrate how suitable lattices may be constructed.

Proposition A. *There is a function $\alpha\colon \mathbf{Z}_{>0} \to \mathbf{R}_{>0}$ with $\lim_{m\to\infty} \alpha(m) = 1/\log 2$ such that for any integers a, b, n and any polynomial $p \in \mathbf{Z}[X]$ with*

$$p \notin \mathbf{Z}, \qquad p \text{ monic}, \qquad n > 1, \qquad 0 < b - a \leq n^{1/\deg p},$$

the number of integers x with $a \le x \le b$ and $p(x) \equiv 0 \bmod n$ is at most $\deg p + \alpha(n) \cdot \log n$. In addition, there is a polynomial-time algorithm that given such a, b, n, and p, determines all those x.

Proof. We write $d = \deg p$, and we let h be the least positive integer satisfying the inequality $2^{dh-1} > (dh)^2 \cdot n^{1-1/d}$. One readily checks that one has $dh < \deg p + \alpha(n) \cdot \log n$ for a function α as in the Proposition, so to prove the first statement it suffices to show that the number of desired values for x is smaller than dh.

Define L to be the additive group of polynomials in the subring $\mathbf{Z}[X, p/n]$ of $\mathbf{R}[X]$ that have degree smaller than dh. Then L is a free abelian group of rank dh, with basis $X^i(p/n)^j$, $0 \le i < d$, $0 \le j < h$, and it contains $\sum_{i=0}^{dh-1} \mathbf{Z} \cdot X^i$ as a subgroup of index $n^{dh(h-1)/2}$. To endow L with a lattice structure, write any polynomial $f \in \mathbf{R}[X]$ with $\deg f < dh$ in the form $f = \sum_{i=0}^{dh-1} c_i (X - \frac{b+a}{2})^i$ with $c_i \in \mathbf{R}$, and put $q(f) = \sum_i c_i^2 (\frac{b-a}{2})^{2i}$. This makes L into a lattice, and a straightforward calculation gives

$$d(L) = \Big(\frac{b-a}{2}\Big)^{dh(dh-1)/2} \cdot n^{-dh(h-1)/2}.$$

For any real number x with $a \le x \le b$ one has $\big|x - (b+a)/2\big| / \big((b-a)/2\big) \le 1$, so the Cauchy-Schwarz inequality implies $|f(x)| \le \big(dh \cdot q(f)\big)^{1/2}$ for any $f \in \mathbf{R}[X]$ with $\deg f < dh$. We can now apply the Lemma with $c = \sqrt{dh}$. Condition (iii) is

$$dh \cdot \Big(\frac{b-a}{2}\Big)^{(dh-1)/2} \cdot n^{-(h-1)/2} < 1.$$

From $b - a \le n^{1/d}$ and the choice of h it follows that this condition is satisfied. The Lemma now implies that there is a non-zero polynomial $g \in \mathbf{Q}[X]$ of degree smaller than dh that has all $x \in \mathbf{Z}$ with $a \le x \le b$ and $p(x) \equiv 0 \bmod n$ among its zeroes. It follows that the number of those x is smaller than dh, as desired.

It is straightforward to convert the proof just given into a polynomial-time algorithm finding all desired values of x. Instead of the version of the Lemma that depends on Minkowski's theorem, one uses the algorithmic version, in which (iii) is replaced by a stronger condition. Thus, h needs to be chosen somewhat larger, but one can still assure that dh is small enough for the algorithm to run in polynomial time. Basis reduction yields a polynomial g of degree smaller than dh as above. All of its integral zeroes can be determined by the method of Section 13, and these can be checked one by one. This proves Proposition A.

The exponent $1/\deg p$ in Proposition A is best possible as a function of $\deg p$. Namely, for any integer $d > 1$ and any real number $\eta > 1/d$, the number of $x \in \mathbf{Z}$ with $0 \le x \le n^\eta$ that are zeroes of $p = X^d$ modulo an integer n that is a dth power, grows exponentially with $\log n$; thus, there does not exist a polynomial-time algorithm for enumerating all those x.

Proposition B. *There is a positive real number β such that for any three integers u, v, n with*

$$\gcd(u, v) = 1, \qquad n > 1, \qquad v \ge n^{1/4},$$

the number of positive divisors x of n with $x \equiv u \bmod v$ is at most $\beta \cdot (\log n)^2$. In addition, there is a polynomial-time algorithm that given such u, v, n, determines all those x.

Proof. Any divisor of n that is congruent to u mod v is coprime to v. Hence, replacing n by the largest divisor of n that is coprime to v (and dealing separately with the case in which this divisor equals 1), we may assume $\gcd(n, v) = 1$. We shall do this throughout the proof.

Let a, b, h be positive integers with $b > a$. We start by establishing, under suitable conditions, an upper bound for the number of divisors x of n with $a \le x \le b$ and $x \equiv u \bmod v$, the number h being an auxiliary parameter.

The lattice to be used is of full rank in the $2h+1$-dimensional subspace $\sum_{i=-h}^{h} \mathbf{R} \cdot X^i$ of the ring $\mathbf{R}[X, X^{-1}]$ of Laurent polynomials over $\mathbf{R}$. On this vector space, we define a positive definite quadratic form q by

$$q(f) = \sum_{i=0}^{h} c_i^2 \cdot \Big(\frac{b-a}{2}\Big)^{2i} + \sum_{i=1}^{h} d_i^2 \cdot \Big(\frac{a^{-1}-b^{-1}}{2}\Big)^{2i}$$

if

$$f = \sum_{i=0}^{h} c_i \cdot \Big(X - \frac{b+a}{2}\Big)^i + \sum_{i=1}^{h} d_i \cdot \Big(X^{-1} - \frac{a^{-1}+b^{-1}}{2}\Big)^i, \qquad c_i, d_i \in \mathbf{R}.$$

As in the previous proof, for any such f and any $x \in \mathbf{R}$ with $a \le x \le b$ one has $|f(x)| \le \big((2h+1) \cdot q(f)\big)^{1/2}$, so that condition (ii) of the Lemma will be satisfied with $c = \sqrt{2h+1}$.

One checks that the lattice $L_0 = \sum_{i=-h}^{h} \mathbf{Z} \cdot X^i$ in $\sum_{i=-h}^{h} \mathbf{R} \cdot X^i$ has determinant

$$d(L_0) = \Big(\frac{b-a}{2}\Big)^{h(h+1)/2} \cdot \Big(\frac{a^{-1}-b^{-1}}{2}\Big)^{h(h+1)/2}.$$

Write $Y = n \cdot X^{-1}$. Then the elements of the sublattice $L_1 = \sum_{i=0}^{h} \mathbf{Z} \cdot X^i + \sum_{i=1}^{h} \mathbf{Z} \cdot Y^i$ of L_0 are integral-valued on the set of divisors of n. One has $(L_0 : L_1) = n^{h(h+1)/2}$ and

therefore

$$d(L_1) = \Big(\frac{b-a}{2}\Big)^{h(h+1)/2} \cdot \Big(\frac{n/a - n/b}{2}\Big)^{h(h+1)/2}.$$

Write $Z = (X-u)/v$. Then all elements of the lattice $L = L_1 + \sum_{i=0}^{2h} \mathbf{Z} \cdot Y^h Z^i \subset \sum_{i=-h}^{h} \mathbf{R} \cdot X^i$ are integral-valued on the set of divisors x of n with $x \equiv u \bmod v$. From $\gcd(n, v) = 1$ one deduces $(L : L_1) = v^{h(2h+1)}$, so

$$d(L) = \Big(\frac{b-a}{2}\Big)^{h(h+1)/2} \cdot \Big(\frac{n/a - n/b}{2}\Big)^{h(h+1)/2} \cdot v^{-h(2h+1)}.$$

Now the Lemma shows: if h satisfies the inequality

$$(2h+1)^2 \cdot \Big(\frac{b-a}{2}\Big)^{h(h+1)/(2h+1)} \cdot \Big(\frac{n/a - n/b}{2}\Big)^{h(h+1)/(2h+1)} \cdot v^{-2h} < 1,$$

then there exists a non-zero element $g \in L$ that has all divisors x of n with $x \equiv u \bmod v$ and $a \le x \le b$ among its zeroes, so that the number of such x is at most $2h$.

To investigate which values of h satisfy the inequality, we restrict to the case $b = 2a$. Then one has $\big((b-a)/2\big) \cdot (n/a - n/b)/2 = n/8$. From $v \ge n^{1/4}$ one now deduces that the inequality for h is satisfied if

$$(2h+1)^{2(2h+1)} \cdot n^{h/2} < 8^{h(h+1)}.$$

Such a value for h can be chosen to satisfy $h \le \delta \cdot \log n$ for some positive constant δ. Thus, we have shown that for any positive integer a, the number of divisors x of n with $x \equiv u \bmod v$ and $a \le x \le 2a$ is at most $2\delta \log n$. We apply this to $a = 1, 2, \ldots, 2^t$, where t is maximal with $2^t < n$. It follows that the number of positive divisors x of n with $x \equiv u \bmod v$ is at most $(1 + (\log n)/\log 2) \cdot 2\delta \log n$. This implies the first statement of Proposition B.

The conversion of the proof just given into a polynomial-time algorithm follows exactly the same lines as in the case of Proposition A. This proves Proposition B.

The lattice L used in the proof just given equals the intersection of $\sum_{i=-h}^{h} \mathbf{R} \cdot X^i$ with the subring $\mathbf{Z}[X, Y, Z]$ of $\mathbf{R}[X, X^{-1}]$. The reader may verify that use of the lattice $L_1 + \sum_{i=0}^{2h} \mathbf{Z} \cdot Y^h \binom{Z}{i}$ leads to a notably better result if n has no small prime factors.

Choosing a different partition of $[1, n]$ into intervals $[a, b]$, and using the lattice $L = \mathbf{Z}[X, Y, Z] \cap \sum_{i=-h}^{k} \mathbf{R} \cdot X^i$ for suitable h, k depending on a, b, one can improve the bound $\beta \cdot (\log n)^2$ given in Proposition B to $\beta \cdot (\log n)^{3/2}$. This result is due to D. J. Bernstein [3, Theorem 6.4].

Pollard [26] exhibited in 1974 a deterministic and fully proved algorithm for factoring integers that runs in time $n^{1/4+o(1)}$ when the number n to be factored tends to infinity. His result is still the best that is known. Pollard's algorithm depends on fast multiplication techniques. A different algorithm that proves the same result, and that has excellent parallelization properties, is obtained from Proposition B, as follows.

Corollary. *There exists, for some positive real number c, an algorithm that given a positive integer n, determines the complete prime factorization of n in time at most $n^{1/4} \cdot (2 + \log n)^c$.*

Proof. We give a brief sketch of the algorithm. First, reduce to the case n is odd. Next, let v be the least power of 2 with $v > n^{1/4}$, and apply the algorithm from Proposition B to all odd values of u with $0 < u < v$. This gives rise to a complete list of divisors of n, from which one easily assembles the prime factorization of n. This proves the Corollary.

Proposition C. (a) *Let a, b, n be integers with $0 < b - a < n$, let ϵ be the real number with $b - a = n^\epsilon$, and let $\eta \in \mathbf{R}$ satisfy $\eta < 1 - \sqrt{\epsilon}$. Then there are at most $3/(1 - \sqrt{\epsilon} - \eta)$ integers x with $a \le x \le b$ for which the denominator of x/n is at most n^η.*

(b) *There is an algorithm that, given integers a, b, n, k, h with $h > k > 0$ and $0 < b - a \le n^{k^2/h^2}$, determines, in time bounded by a polynomial function of $\log(|a| + |b|)$, $\log n$, and h, all integers x with $a \le x \le b$ for which the denominator of x/n is at most $n^{1-k/h-1/(2h)}$.*

Proof. Let a, b, n be as in (a). We let m be a positive integer, to be thought of as an upper bound for the denominator of x/n. Further we let h, k be integers satisfying $h > k > 0$; these are auxiliary parameters.

We shall consider lattices of full rank in the h-dimensional subspace $\sum_{i=0}^{h-1} \mathbf{R} \cdot X^i Z^k$ of the polynomial ring $\mathbf{R}[X, Z]$. For $f = \sum_{i=0}^{h-1} c_i \big(X - (b+a)/2\big)^i Z^k$ (with $c_i \in \mathbf{R}$) in that space, we write $q(f) = \sum_{i=0}^{h-1} c_i^2 \big((b-a)/2\big)^{2i} m^{2k}$; as in the earlier proofs in this section, we have $|f(x)| \le \big(h \cdot q(f)\big)^{1/2}$ for all x, $z \in \mathbf{R}$ with $a \le x \le b$, $1 \le z \le m$.

The lattice $L_0 = \sum_{i=0}^{h-1} \mathbf{Z} \cdot X^i Z^k$ has rank h and $d(L_0) = \big((b-a)/2\big)^{h(h-1)/2} \cdot m^{kh}$. Write $Y = XZ/n$. Then the lattice $L = \sum_{i=0}^{k} \mathbf{Z} \cdot Y^i Z^{k-i} + \sum_{j=1}^{h-k-1} \mathbf{Z} \cdot X^j Y^k$ in $\sum_{i=0}^{h-1} \mathbf{R} \cdot X^i Z^k$ contains L_0 as a sublattice of index $n^{kh-k(k+1)/2}$, so one has $d(L) = \big((b-a)/2\big)^{h(h-1)/2} \cdot m^{kh} \cdot n^{-kh+k(k+1)/2}$. All $f \in L$ are integral-valued on the set of pairs of integers (x, z) for which x/n has denominator dividing z.

Now the Lemma implies: if m, h, k satisfy the inequality

$$\frac{h^2}{2^{h-1}} \cdot (b-a)^{h-1} \cdot m^{2k} \cdot n^{-2k+k(k+1)/h} < 1,$$

then there is a non-zero polynomial $g \in \mathbf{Q}[X]$ with $\deg g < h$ that has among its zeroes all integers x with $a \leq x \leq b$ for which x/n has denominator at most m, so that the number of such x is at most $h-1$. For example, with $h=2$, $k=1$ this shows that x is unique (if it exists) whenever $m < \sqrt{n/(b-a)}/\sqrt{2}$, which is slightly weaker than what we saw earlier.

To prove (a), put $\epsilon = \big(\log(b-a)\big)/\log n$ as in (a), and let $\eta < 1-\sqrt{\epsilon}$. Since we know that there is at most one x as in (a) if $\eta < (1-\epsilon)/2$, we may assume $\eta \geq (1-\epsilon)/2$. Then we have $1-\sqrt{\epsilon}-\eta < 1/2$. Choose h to be the unique integer with $1/h < (1-\sqrt{\epsilon}-\eta)/3 \leq 1/(h-1)$ and k to be the least integer with $k \geq h\sqrt{\epsilon}$. Then one verifies that we have $0 < k < h$ and

$$h \geq 7, \qquad \frac{1}{2} \cdot \Big(\frac{h-1}{k} \cdot \epsilon + \frac{k+1}{h}\Big) < 1-\eta.$$

This implies that h, k, and $m = \lfloor n^\eta \rfloor$ satisfy the inequality above, so the number of x is at most $h-1$, which by the choice of h is at most $3/(1-\sqrt{\epsilon}-\eta)$. This proves (a).

The proof of (b) follows the same lines as before. It depends on the inequality

$$\frac{1}{2} \cdot \Big(\frac{h-1}{k} \cdot \frac{k^2}{h^2} + \frac{k+1}{h}\Big) < \frac{k}{h} + \frac{1}{2h}.$$

Note that replacing k, h by $4k$, $4h$, if necessary, one may assume $h \geq 7$. This proves Proposition C.

Remark. No particular effort has been spent on optimizing the constant 3 in the bound $3/(1-\sqrt{\epsilon}-\eta)$ in (a). A more pressing issue is to decide whether the number of x in (a) may be bounded above by a continuous function of ϵ alone.

Error correction in $\mathbf{Z}/n\mathbf{Z}$. The result just proved admits an attractive reformulation in the terminology of coding theory. Let n be an integer with $n > 1$. We define an 'n-adic' metric d on the underlying set of the ring $\mathbf{Z}/n\mathbf{Z}$ by putting $d(r,s) = (\log \#J)/\log n$, where J is the ideal of $\mathbf{Z}/n\mathbf{Z}$ generated by $r-s$; the reader may verify that d is indeed a metric, and that the maximal value assumed by d equals 1. This metric is closely related to the *Hamming metric* from coding theory (see [29]). To see this, assume momentarily that n is squarefree, write P for the set of prime factors of n, and identify $\mathbf{Z}/n\mathbf{Z}$ with $\prod_{p\in P} \mathbf{Z}/p\mathbf{Z}$ through the ring isomorphism sending r to $(r \bmod p)_{p\in P}$. Two 'vectors' $(r_p)_{p\in P}$, $(s_p)_{p\in P}$ in

$\prod_{p\in P}\mathbf{Z}/p\mathbf{Z}$ have Hamming distance $\#\{p : r_p \neq s_p\}$, whereas their newly defined distance equals $\big(\sum_{p,\, r_p\neq s_p}\log p\big)/\sum_{p\in P}\log p$; thus, d is a weighted version of the Hamming distance, the weights having been normalized such that the maximum distance equals 1.

Note that, for general n and all x, $x' \in \mathbf{Z}$, the denominator of $(x-x')/n$ equals $n^{d(x \bmod n,\, x' \bmod n)}$.

Next let, in addition to an integer $n > 1$, two integers a, b with $0 < b-a < n$ be given, and write

$$C = \{(x \bmod n) : x \in \mathbf{Z},\, a \leq x \leq b\}, \qquad \delta = 1 - \frac{\log(b-a)}{\log n}.$$

We think of the subset C of $\mathbf{Z}/n\mathbf{Z}$ as a *code*, and, as in coding theory, we refer to δ as the *designed distance* of C. To justify this terminology, suppose that x, x' are integers with $a \leq x < x' \leq b$. Then we have $d(x \bmod n, x' \bmod n) = 1 - \big(\log\gcd(x'-x, n)\big)/\log n \geq 1 - \big(\log(b-a)/\log n\big) = \delta$, so the 'distance' $\min\{d(v,w) : v,\, w \in C,\, v \neq w\}$ of C is at least δ. From $\delta > 0$ we also see that no two distinct integers x, $x' \in [a,b]$ are congruent modulo n, so we have $\#C = b-a+1$.

For given $r \in \mathbf{Z}/n\mathbf{Z}$, one is now interested in the set of all $v \in C$ for which $d(v,r)$ is small; say, $d(v,r) \leq \eta$, where η is a given real number. For v, $w \in C$, $v \neq w$, one has $d(v,r) + d(w,r) \geq d(v,w) \geq \delta$, so at most one $v \in C$ satisfies $d(v,r) < \delta/2$. If $u \in \mathbf{Z}$ is such that $r = (u \bmod n)$, then the set of all $v \in C$ with $d(v,r) \leq \eta$ is the same as the set of all $(x \bmod n) + r$, where x ranges over those integers with $a-u \leq x \leq b-u$ for which x/n has denominator at most n^{η}. Thus, the results of Proposition C can be transposed to the present setting. From (a) one sees that, for any $\eta < 1 - \sqrt{1-\delta}$, the number of $v \in C$ with $d(v,r) \leq \eta$ is at most $3/(1 - \sqrt{1-\delta} - \eta)$; note that $\delta/2 < 1 - \sqrt{1-\delta}$. Similarly, (b) gives rise to an efficient 'decoding algorithm' past half the designed distance.

The analogue of Proposition C in non-zero characteristic, which may be based on the theory from Section 16 below, has applications to decoding Reed-Solomon and algebraic geometry codes from conventional coding theory, see [14; 3, Section 7].

16. Lattices over polynomial rings

There is an analogue of the notion of lattice in which the role of the ring $\mathbf{Z}$ of integers is played by the ring $k[t]$ of polynomials in one variable t over a field k. The theory, to which we alluded in earlier sections, is in substance due to Mahler [24]. Some of the main points are presented below, but we have good reasons to forgo a detailed treatment: from an algorithmic point of view, the theory has little to offer that one cannot obtain from linear algebra over k; and from a theoretical point of view the almost equivalent language of *vector bundles over the projective line* is more common.

Let k and $k[t]$ be as above, and let $\deg\colon k[t] \to \{-\infty\} \cup \mathbf{R}$ map each non-zero polynomial to its degree and 0 to $-\infty$. By a $k[t]$-*lattice* we mean a pair consisting of a finitely generated $k[t]$-module L and a function $q\colon L \to \{-\infty\} \cup \mathbf{R}$ with the following properties:

$$q(x+y) \leq \max\{q(x), q(y)\} \text{ for all } x, y \in L,$$
$$q(cx) = \deg c + q(x) \text{ for all } c \in k[t],\ x \in L,$$
$$q(x) \neq -\infty \text{ for all } x \in L,\ x \neq 0,$$
$$\dim_k\{x \in L : q(x) \leq r\} < \infty \text{ for each } r \in \mathbf{R}.$$

The first two properties imply that $\{x \in L : q(x) \leq r\}$ is a k-vector space for each $r \in \mathbf{R}$, so the dimension referred to in the last property is well-defined. To improve the resemblance to the definition given in Section 2, one may replace q by the function $L \to \mathbf{R}$ sending x to $\exp(q(x))$. One often restricts to lattices that are *integral-valued* in the sense that the image of q is contained in $\{-\infty\} \cup \mathbf{Z}$.

Examples. (a) For each $\lambda \in \mathbf{R}$, an example of a $k[t]$-lattice is given by $L = k[t]$, $q(f) = \lambda + \deg f$; this lattice is denoted by $\mathcal{O}(-\lambda)$. If L_1, q_1 and L_2, q_2 are $k[t]$-lattices, then their *orthogonal sum* is the $k[t]$-lattice $L = L_1 \oplus L_2$ with $q(x_1, x_2) = \max\{q_1(x_1), q_2(x_2)\}$, for $x_1 \in L_1$, $x_2 \in L_2$. Somewhat surprisingly, there exists for every $k[t]$-lattice a finite sequence $\lambda_1, \ldots, \lambda_n$ of real numbers such that the lattice is, in an obvious sense, isomorphic to the orthogonal sum of the n lattices $\mathcal{O}(-\lambda_i)$, $1 \leq i \leq n$; if we also require $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$, then the λ_i are uniquely determined as the *successive minima* of the lattice, and all λ_i are in $\mathbf{Z}$ if and only if the lattice is integral-valued. Thus, unlike usual lattices, $k[t]$-lattices admit a satisfactory classification.

(b) The reader acquainted with algebraic geometry (see [15]) can obtain $k[t]$-lattices from vector bundles over the projective line, as follows. Write $\mathbf{A}^1_k$ for $\operatorname{Spec} k[t]$, and let $\mathbf{P}^1_k = \mathbf{A}^1_k \cup \{\infty\}$ be the projective line over k. If $\mathcal{E}$ is a vector bundle over $\mathbf{P}^1_k$, then

$L = \mathcal{E}(\mathbf{A}_k^1)$ is a $k[t]$-lattice, with $q(x) = \min\{m \in \mathbf{Z} : x \in t^m \mathcal{E}_\infty\}$ for $x \in L$, $x \neq 0$, and $q(0) = -\infty$. The $k[t]$-lattices obtained in this way are integral-valued, and conversely, each integral-valued $k[t]$-lattice arises, up to isomorphism, from exactly one vector bundle over $\mathbf{P}_k^1$. The classification just referred to amounts in this case to Grothendieck's theorem describing all vector bundles over the projective line (see [13, Theorem 2.1]).

(c) Just as the ring $k[t]$ plays the role that in previous sections was played by $\mathbf{Z}$, so does the field $k(t)$ of fractions of $k[t]$ play the role of $\mathbf{Q}$. In Section 3 we obtained examples of lattices from algebraic number fields, and in a similar way one can obtain $k[t]$-lattices from fields that are finite extensions of $k(t)$. Let K be such a field, and write A for the integral closure of $k[t]$ in K. Consider the set D of all maps $d\colon K \to \{-\infty\} \cup \mathbf{R}$ satisfying $d(xy) = d(x) + d(y)$ and $d(x+y) \leq \max\{d(x), d(y)\}$ for all x, $y \in K$, as well as $d(x) \neq -\infty$ for all $x \neq 0$ and $d(x) = \deg x$ for all $x \in k[t]$; so the maps $-d$, $d \in D$, are the exponential valuations of K that extend the 'infinite valuation' $-\deg$ of $k(t)$. From valuation theory it is well-known that the set D is finite and non-empty. We may make A into a $k[t]$-lattice by putting $q(x) = \max\{d(x) : d \in D\}$ for $x \in A$. If the infinite valuation is ramified in K, this is an example of a $k[t]$-lattice that is not integral-valued.

(d) The role of Euclidean vector spaces is in the current theory played by certain normed vector spaces over the completion $k(t)_\infty$ of $k(t)$ at the infinite prime. One may identify this completion with the field $k((t^{-1}))$ of formal Laurent series in t^{-1} over k, and define $\deg\colon k(t)_\infty \to \{-\infty\} \cup \mathbf{R}$ by $\deg\bigl(\sum_{i \in \mathbf{Z}, i \leq m} a_i t^i\bigr) = m$ for $a_i \in k$, $a_m \neq 0$, and $\deg 0 = -\infty$. For integral-valued lattices, the only normed vector spaces one needs to consider are of the form $E = k(t)_\infty^n$, with $n \in \mathbf{Z}_{\geq 0}$, the norm $q\colon E \to \{-\infty\} \cup \mathbf{R}$ being defined by $q((c_i)_{i=1}^n) = \max\{\deg c_i : 1 \leq i \leq n\}$ (and $q(E) = \{-\infty\}$ if $n = 0$). For each basis $b_1, \ldots, b_n$ of E over $k(t)_\infty$, the $k[t]$-module $L = \sum_{i=1}^n k[t] \cdot b_i$, together with the restriction of q to L, is an integral-valued $k[t]$-lattice. This is the way in which integral-valued $k[t]$-lattices are often represented numerically. In order to specify the entries of the basis vectors b_i by means of a finite number of elements of k, one may require them to be 'rational' in the sense that they belong to the subfield $k(t)$ of $k(t)_\infty$; in algorithmic circumstances, one will also need to place restrictions on the base field k.

To represent general $k[t]$-lattices in a similar way, it suffices to choose real numbers $\lambda_1, \ldots, \lambda_n$ and to redefine q on E by $q((c_i)_{i=1}^n) = \max\{\lambda_i + \deg c_i : 1 \leq i \leq n\}$.

Basis reduction. Let L, q be a $k[t]$-lattice. Then L has a *basis* as a $k[t]$-module, i.e., a sequence $b_1, \ldots, b_n$ of elements of L such that the map $k[t]^n \to L$ sending $(c_i)_{i=1}^n$ to $\sum_{i=1}^n c_i b_i$ is bijective. A basis $b_1, \ldots, b_n$ is called *reduced* if for each $(c_i)_{i=1}^n \in k[t]^n$ one has

$q\big(\sum_{i=1}^n c_i b_i\big) = \max\{q(c_i b_i) : 1 \le i \le n\}$. The classification theorem stated in Example (a) is readily seen to imply that each $k[t]$-lattice has a reduced basis. One may wonder whether there is an algorithmic version of the classification theorem. In other words, is there an 'algorithm' that, given a $k[t]$-lattice L as in Example (d), produces a reduced basis for L? In the case k is finite and the lattice $L \subset E = k(t)^n_\infty$ from (d) is a sublattice of $k[t]^n$, such an algorithm, running in polynomial time, was exhibited by A. K. Lenstra [20, Section 1]. It is not hard to adapt his algorithm to more general situations.

Linear algebra. The reader may enjoy developing the theory further, defining the determinant of a lattice and finding the analogue of Minkowski's theorem; but it is good to realize that almost anything that one can do with $k[t]$-lattices can also be done by means of linear algebra over k. In many applications, one is interested in the set $\{x \in L : q(x) \le r\}$ for some $k[t]$-lattice L and some $r \in \mathbf{R}$; that set is a finite-dimensional k-vector space, and one can usually compute a k-basis of that vector space using linear algebra over k (see [20, Section 1]). Over infinite fields, such as $k = \mathbf{Q}$, linear algebra has the distinct advantage of offering ready means for controlling coefficient blow-up. For finite k, however, the linear algebra approach is less efficient than the approach through $k[t]$-lattice basis reduction [20, Section 1]. This algorithmic distinction may be the one redeeming feature of the theory of $k[t]$-lattices.

References

1. K. Aardal, F. Eisenbrand, *Integer programming, lattices, and results in fixed dimension*, K. Aardal, G. L. Nemhauser, R. Weismantel (eds), *Discrete optimization*, North-Holland, Amsterdam, to appear, Chapter 4.
2. K. Belabas, M. van Hoeij, J. Klüners, A. Steel, *Factoring polynomials over global fields*, to appear.
3. D. J. Bernstein, *Reducing lattice bases to find small-height values of univariate polynomials*, these proceedings.
4. J. A. Buchmann, H. W. Lenstra, Jr., *Approximating rings of integers in number fields*, J. Théor. Nombres Bordeaux **6** (1995), 221–260.
5. J. P. Buhler, S. Wagon, *Basic algorithms in number theory*, these proceedings.
6. J. W. S. Cassels, *Rational quadratic forms*, Academic Press, London, 1978.
7. H. Cohen, *A course in computational algebraic number theory*, Springer-Verlag, Berlin, 1993.

8. J. H. Conway, N. J. A. Sloane, *Sphere packings, lattices and groups*, Springer-Verlag, New York, 1988.
9. B. de Smit, *Measure characteristics of complexes*, Cahiers Topologie Géom. Différentielle Catég. **37** (1996), 3–20.
10. N. D. Elkies, *Rational points near curves and small nonzero $|x^3 - y^2|$ via lattice reduction*, W. Bosma (ed.), *Algorithmic number theory* (ANTS IV), Lecture Notes in Comput. Sci. **1838**, Springer-Verlag, Berlin, 2000, 33–63.
11. C. F. Gauss, *Disquisitiones arithmeticae*, Fleischer, Leipzig, 1801.
12. O. Goldreich, D. Micciancio, S. Safra, J. P. Seifert, *Approximating shortest lattice vectors is not harder than approximating closest lattice vectors*, Inform. Process. Lett. **71** (1999), 55–61.
13. A. Grothendieck, *Sur la classification des fibrés holomorphes sur la sphère de Riemann*, Amer. J. Math. **79** (1957), 121–138.
14. V. Guruswami, M. Sudan, *Improved decoding of Reed-Solomon and algebraic-geometry codes*, IEEE Trans. Inform. Theory **45** (1999), 1757–1767.
15. R. Hartshorne, *Algebraic geometry*, Springer-Verlag, New York, 1977.
16. G. H. Hardy, E. M. Wright, *An introduction to the theory of numbers*, Oxford University Press, Oxford, 1938.
17. D. R. Heath-Brown, *The density of rational points on curves and surfaces*, Ann. of Math. (2) **155** (2002), 553–598.
18. D. E. Knuth, *The art of computer programming*, Volume 2, *Seminumerical algorithms*, Addison-Wesley, Reading, Mass., second edition, 1981.
19. S. Lang, *Introduction to Arakelov theory*, Springer-Verlag, New York, 1988.
20. A. K. Lenstra, *Factoring multivariate polynomials over finite fields*, J. Comput. System Sci. **30** (1985), 235–248.
21. A. K. Lenstra, H. W. Lenstra, Jr., L. Lovász, *Factoring polynomials with rational coefficients*, Math. Ann. **261** (1982), 515–534.
22. H. W. Lenstra, Jr., *Integer programming with a fixed number of variables*, Math. Oper. Res. **8** (1983), 538–548.
23. H. W. Lenstra, Jr., *Flags and lattice basis reduction*, C. Casacuberta et al. (eds), *European congress of mathematics, Barcelona, July 10–14, 2000*, vol. I, Birkhäuser Verlag, Basel, 2001, 37–51.
24. K. Mahler, *An analogue to Minkowski's geometry of numbers in a field of series*, Ann. of Math. **42** (1941), 488–522.

25. M. Mignotte, *An inequality about factors of polynomials*, Math. Comp. **28** (1974), 1153–1157.
26. J. M. Pollard, *Theorems on factorization and primality testing*, Proc. Cambridge Philos. Soc. **76** (1974), 521–528.
27. M. O. Rabin, J. O. Shallit, *Randomized algorithms in number theory*, Comm. Pure Appl. Math. **39** (1986), no. S, suppl., S239–S256.
28. A. Schrijver, *Theory of linear and integer programming*, John Wiley, Chicester, 1986.
29. J. H. van Lint, *Introduction to coding theory*, Springer-Verlag, New York, 1982.
30. J. von zur Gathen, J. Gerhard, *Modern computer algebra*, Cambridge University Press, 1999.