Short Vectors of Planar Lattices
Via Continued Fractions

Friedrich Eisenbrand

MPI–I–2000–2-001 May 2000

Author's Address

Friedrich Eisenbrand
Max-Planck-Institut für Informatik
Im Stadtwald
66123 Saarbrücken, Germany
eisen@mpi-sb.mpg.de

Abstract

We describe how a shortest vector of a 2-dimensional integral lattice corresponds to a best approximation of a unique rational number defined by the lattice. This rational number and its best approximations can be computed with the euclidean algorithm and its speedup by Schönhage (1971) from any basis of the lattice. The described correspondence allows, on the one hand, to reduce a basis of a 2-dimensional integral lattice with the euclidean algorithm, up to a single normalization step. On the other hand, one can use the classical result of Schönhage (1971) to obtain a shortest vector of a 2-dimensional integral lattice with respect to the ℓ_∞-norm. It follows that in two dimensions, a fast basis-reduction algorithm can be solely based on Schönhage's algorithm and the reduction algorithm of Gauß (1801).

1 Introduction

Lattice basis-reduction is an important technique in computer science. Well known applications are integer programming in fixed dimension (Lenstra 1983), factorization of rational polynomials (Lenstra, Lenstra & Lovász 1982) or the development of strongly polynomial algorithms in combinatorial optimization (Frank & Tardos 1987), among others.

Gauß (1801) invented an algorithm that finds a "short" or reduced basis of a 2-dimensional integral lattice. Such a basis consists of two integral vectors $b_1, b_2 \in \mathbf{Z}^2$ that generate the lattice, with the additional property that the enclosed angle between b_1 and b_2 is in the range $90^\circ \pm 30^\circ$. A shortest vector of a reduced basis is then a shortest vector of the lattice. The algorithm mimics the euclidean algorithm by subtracting integral multiples of the shorter vector from the larger vector thereby reducing its length. This *normalization step* is analogous to the division with remainder in the euclidean algorithm for integers.

Algorithm. $\textsc{Gauss}(b_1, b_2)$

repeat
  arrange that b_1 is the shorter vector of b_1 and b_2
  find $k \in \mathbf{Z}$ such that $b_2 - kb_1$ is of minimal euclidean length
  $b_2 \leftarrow (b_2 - kb_1)$ (*normalization step*)
until $k = 0$
return (b_1, b_2)

The integer k in the repeat-loop of algorithm GAUSS is the nearest integer to the number $(b_1^T b_2)/(b_1^T b_1)$. Figure 1 shows the effect of a normalization step. The length of the second basis vector b_2 has been reduced by subtracting integral multiples of b_1. Lagarias (1980) showed that the Gaussian algorithm has worst-case complexity $O(n^3)$, where n is the size of the binary encoding of the input. (Rote 1997) showed that the 2-dimensional $\bmod m$ shortest vector problem can be reduced to the classical case. See, e.g., (Yap 1999) for a thorough treatment of the Gaussian reduction algorithm.

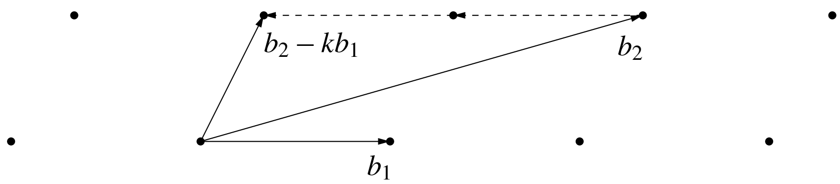

Figure 1: The effect of a normalization step.

A reduced basis of a 2-dimensional integral lattice can actually be computed much faster. Schönhage (1991) and independently Yap (1992) invented basis-reduction algorithms for 2-dimensional lattices with worst-case complexity $O(M(n) \log n)$, where $M(n)$ is the time needed to multiply two n-bit integers. This is in contrast to the fastest known algorithm for the shortest vector problem in arbitrary fixed dimension

(Kannan 1987), which runs in time $O(M(n)\,n)$. In fact Schönhage (1991) solves the closely related but more general problem of reducing an integral not necessarily definite binary quadratic form. The algorithm of Yap (1992) is in the setting of 2-dimensional integral lattices. Both algorithms are based on new techniques and are fairly involved compared to the classical algorithm of Schönhage (1971), that computes the common convergent of two rationals in time $O(M(n)\log n)$.

We show in this paper that a shortest vector of a 2-dimensional integral lattice corresponds to a best approximation of a rational number α, which is uniquely defined by the lattice. This number α can be obtained from any basis of the lattice with the extended euclidean algorithm for integers. The best approximations of α are convergents of α and can again be obtained with the extended euclidean algorithm for integers. This shows that the extended euclidean algorithm can be used to reduce a lattice basis, up to a single normalization step.

On the other hand, this also implies that there is no need for a special algorithm for the fast basis-reduction of a 2-dimensional integral lattice, since the classical result of Schönhage (1971) can be directly applied to find a shortest vector w.r.t. the ℓ_∞-norm and thus to find an "almost reduced" basis. A reduced basis can then be obtained by applying a constant number of Gaussian normalization steps.

It is known that the Gaussian reduction algorithm and the euclidean algorithm are related. Vallée (1991) provided an "a posteriori" connection when one already knows a reduced basis. Daudé, Flajolet & Vallée (1997) showed that the Gaussian algorithm translates into a complex continued fraction expansion and used this for an average-case analysis of the algorithm GAUSS.

In this paper we do not investigate relationships between the Gaussian algorithm and the euclidean algorithm. Instead we show that the classical euclidean algorithm and its speedup by Schönhage (1971) can be used to find short vectors of 2-dimensional lattices.

The Gaussian reduction algorithm is often considered as a 2-dimensional generalization of the euclidean algorithm. Our research implies that the euclidean algorithm is general enough to solve the shortest vector problem in 2-dimensions.

2 Preliminaries

The letters $\mathbf{Z}, \mathbf{Q}$, and $\mathbf{R}$ denote the integers, rationals and reals respectively. The symbol $\mathbf{N}_+$ denotes the positive natural numbers whereas $\mathbf{N}_0$ denotes the natural numbers including 0. In this paper, the running times of algorithms are always given in terms of the binary encoding length n of the input data. The function $M(n)$ denotes the time needed to multiply two integers. All *basic arithmetic operations* `+, -, *, /` can be done in time $O(M(n))$ (Aho, Hopcroft & Ullman 1974). The ℓ_∞, ℓ_1, and ℓ_2-*norm* of a vector $c = (c_1, c_2)^T \in \mathbf{R}^2$ are the numbers $\|c\|_\infty = \max\{|c_1|, |c_2|\}$, $\|c\|_1 = |c_1| + |c_2|$, and $\|c\|_2 = (c_1^2 + c_2^2)^{1/2}$, respectively. One has $\|c\|_\infty \leqslant \|c\|_2 \leqslant \sqrt{2}\,\|c\|_\infty$.

A *2-dimensional* or *planar integral lattice* Λ is a set of the form $\Lambda(A) = \{Ax \mid x \in \mathbf{Z}^2\}$, where $A \in \mathbf{Z}^{2\times 2}$ is a nonsingular integral matrix. The matrix A is called *basis* of Λ. One has $\Lambda(A) = \Lambda(B)$ for $B \in \mathbf{Z}^{2\times 2}$ if and only if $B = AU$ with some *unimodular matrix* $U \in \mathbf{Z}^{2\times 2}$, i.e., $\det(U) = \pm 1$. Denote by $a^{(i)}$, $i = 1, 2$, the i-th column of A. The basis A of Λ is called *reduced* if

$$2|{a^{(1)}}^T a^{(2)}| \leqslant {a^{(1)}}^T a^{(1)} \leqslant {a^{(2)}}^T a^{(2)}. \tag{1}$$

A *shortest vector* of Λ w.r.t. $\|\cdot\|$ is a nonzero member $0 \neq v$ of Λ whose norm $\|v\|$ is minimal. Here $\|\cdot\|$ stands for the ℓ_∞, ℓ_1 or ℓ_2-norm. The first column of a reduced basis of Λ is a shortest vector of Λ w.r.t. the ℓ_2-norm.

2.1 The euclidean algorithm

The *extended euclidean algorithm* takes as input a pair of integers (a,b) and computes $d = \gcd(a,b)$ and a pair of integers (x,y) with $xa + yb = d$ (see, e.g., (Bach & Shallit 1996, p. 71)).

Algorithm. EXGCD(a,b)

$M \leftarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
$n \leftarrow 0$
while $(b \neq 0)$ **do**
    $q \leftarrow \lfloor a/b \rfloor$
    $M \leftarrow M \begin{pmatrix} q & 1 \\ 1 & 0 \end{pmatrix}$
    $(a,b) \leftarrow (b, a - qb)$
    $n \leftarrow n + 1$
return $(d = a, x = (-1)^n M_{2,2}, y = (-1)^{n+1} M_{1,2})$

Let $M^{(k)}$, $k \geqslant 0$, denote the matrix M after the $k+1$-st iteration of the while-loop in EXGCD. The running time of the extended euclidean algorithm is quadratic (see, e.g., (Bach & Shallit 1996)).

2.2 Continued fractions

Continued fractions are a classic in mathematics, see, e.g., the books of Perron (1954) and Khintchine (1963). A very nice and short treatment can also be found in (Grötschel, Lovász & Schrijver 1988, p. 134-137). Let $a_0, \ldots, a_t$ be integers, all positive, except perhaps a_0. The *continued fraction* $\langle a_0, \ldots, a_t \rangle$ is inductively defined as a_0, if $t = 0$ and as $a_0 + 1/\langle a_1, \ldots, a_t \rangle$ if $t > 0$. The function $f_k(x) = \langle a_0, \ldots, a_{k-1}, x \rangle$, $0 \leqslant k \leqslant t$ is increasing for $x > 0$ if k is even and decreasing for $x > 0$ if k is odd. Consider the two sequences g_k and h_k that are inductively defined as

$$\begin{pmatrix} g_{-1} & g_{-2} \\ h_{-1} & h_{-2} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} g_k & g_{k-1} \\ h_k & h_{k-1} \end{pmatrix} = \begin{pmatrix} g_{k-1} & g_{k-2} \\ h_{k-1} & h_{k-2} \end{pmatrix} \begin{pmatrix} a_k & 1 \\ 1 & 0 \end{pmatrix}, k \geqslant 0. \tag{2}$$

Let $\beta_k = g_k/h_k$, then one has $\langle a_0, \ldots, a_k \rangle = \beta_k$ for $0 \leqslant k \leqslant t$. Note that h_k is increasing in k.

The *continued-fraction expansion* of a number $\alpha \in \mathbf{Q}$ is inductively defined as the sequence α if $\alpha \in \mathbf{Z}$, and as $\lfloor \alpha \rfloor, a_1, \ldots, a_t$ if $\alpha \notin \mathbf{Z}$ and where $a_1, \ldots, a_t$ is the continued fraction expansion of $1/(\alpha - \lfloor \alpha \rfloor)$. If k is even, then a_k is maximal with

$\langle a_0,\dots,a_k\rangle \leqslant \alpha$ and if k is odd, then a_k is maximal with $\alpha \leqslant \langle a_0,\dots,a_k\rangle$. For $0 \leqslant k \leqslant t$, the number $\langle a_0,\dots,a_k\rangle = \beta_k$ is called the *k-th convergent* of α, and we have $\beta_0 < \beta_2 < \cdots < \beta_t = \alpha < \cdots < \beta_3 < \beta_1$. It is easy to see that the continued fraction expansion of a rational number $\alpha = u/v \neq 0$ is the sequence of q's which are computed in the while-loop of the algorithm EXGCD on input (u,v). Let $R^{(k)}$ denote the matrix

$$R^{(k)} = \begin{pmatrix} a_0 & 1 \\ 1 & 0 \end{pmatrix} \cdots \begin{pmatrix} a_k & 1 \\ 1 & 0 \end{pmatrix}.$$

Then $R^{(k)} = M^{(k)}$, when EXGCD is run on (u,v) and $u/v = \alpha$.

A *fraction* is a representation x/y, $y > 0$ of a rational number, where x and y are integers. The fraction is *reduced* if $\gcd(x,y) = 1$. A fraction x/y is a *good approximation* to the number $\alpha \in \mathbf{Q}$, if one has $|\alpha - x/y| \leqslant |\alpha - x'/y'|$ for all other fractions x'/y' with $0 < y' \leqslant y$. Each convergent β_k, $0 \leqslant k \leqslant t$, of $\alpha \in \mathbf{Q}$ is a good approximation to α. A fraction x/y is a *best approximation of the second kind* to the number $\alpha \in \mathbf{Q}$, if one has $|y\alpha - x| < |y'\alpha - x'|$ for all other fractions x'/y' with $0 < y' \leqslant y$, see (Khintchine 1963, p. 28). A best approximation of the second kind to $\alpha \in \mathbf{Q}$ is a convergent of α.

The *common convergent* of two rational numbers $\alpha_1, \alpha_2 \in \mathbf{Q}$ is the convergent $\langle a_0,\dots,a_k\rangle$ of α_1 and α_2 that corresponds to the longest common prefix of the continued fraction expansions of α_1 and α_2. Thus k is maximal such that the k-th convergent of α_1 and the k-th convergent of α_2 are equal. If $\alpha_1 \leqslant \alpha_2$, then this is the common convergent of all rationals in the interval $[\alpha_1, \alpha_2]$. Schönhage (1971) showed how to compute the common convergent β_k and the corresponding matrix $R^{(k)}$ of two rationals $\alpha_1, \alpha_2 \in \mathbf{Q}$ in time $O(M(n)\log n)$. Schönhage's result yields an algorithm that computes in time $O(M(n)\log n)$ the greatest common divisor, $\gcd(a,b)$, of two n-bit integers a and b as well as two n-bit integers x and y that represent it, i.e., $\gcd(a,b) = xa + yb$.

3 The Hermite normal form

Before we establish the connection between best approximations and shortest vectors of planar lattices we perform some preprocessing on the lattice basis $A \in \mathbf{Z}^{2\times 2}$. Let A be of the form $A = \left(\begin{smallmatrix} a_1 & a_2 \\ a_3 & a_4 \end{smallmatrix}\right) \in \mathbf{Z}^{2\times 2}$. First we compute integers x and y that represent the greatest common divisor d of a_3 and a_4, i.e., $d = xa_3 + ya_4$. By multiplying the basis A with the unimodular matrix $\left(\begin{smallmatrix} a_4/d & x \\ -a_3/d & y \end{smallmatrix}\right)$ one obtains an upper triangular matrix

$$\begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} a_4/d & x \\ -a_3/d & y \end{pmatrix} = \begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \in \mathbf{Z}^{2\times 2}.$$

After some unimodular column operations, i.e., multiplying the first and second column with ± 1 and adding integral multiples of the first column to the second column, we can assure that $c > 0$ and $a > b \geqslant 0$ holds. This is the *Hermite normal form*, or *HNF*, of A (see, e.g., (Schrijver 1986, p. 45)). The HNF of an integral lattice is unique and its computation can be carried out in time $O(M(n)\log n)$ with the algorithm of Schönhage (1971).

4 Best approximations and shortest vectors

Here we establish the connection between shortest vectors and best approximations. Interestingly, our observation holds for any norm $\|\cdot\|$ which is invariant under the replacement of components by their absolute value. The ℓ_1, ℓ_2 and ℓ_∞-norms have this property.

Let Λ be given by its HNF $\left(\begin{smallmatrix} a & b \\ 0 & c \end{smallmatrix}\right) \in \mathbf{Z}^{2\times 2}$, where $c > 0$ and $a > b \geqslant 0$. If $a \leqslant c$, then $\left(\begin{smallmatrix} a \\ 0 \end{smallmatrix}\right)$ is a shortest vector of Λ. Therefore assume that $a > c$. If a shortest vector has a negative second component, then it yields a shortest vector with a positive second component by multiplying it with -1. Therefore we can assume that a shortest vector is of the form $\left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right)$, where $x \in \mathbf{N}_0$, $y \in \mathbf{N}_+$.

Lemma 1. *There exists a shortest vector* $\left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right)$, $x \in \mathbf{N}_0$, $y \in \mathbf{N}_+$ *of* Λ *such that at least one of the following conditions is satisfied.*

i. The fraction x/y is a best approximation of the second kind to the number b/a.

ii. If the fraction p/q is the reduced representation of b/a, then p is odd, q is even, $x \in \{\lfloor p/2 \rfloor, \lceil p/2 \rceil\}$ and $y = q/2$.

Proof. Let $\left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right)$, $x \in \mathbf{N}_0$, $y \in \mathbf{N}_+$ be a shortest vector of Λ with minimal ℓ_1-norm among all shortest vectors. We show that one of the above conditions holds.

Suppose that x/y is not a best approximation of the second kind of b/a. Then there exists a fraction $x'/y' \neq x/y$ with $y' \leqslant y$ and $|by' - ax'| \leqslant |by - ax|$. If $y' < y$ or $|by' - ax'| < |by - ax|$, then $\left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right)$ is not a shortest vector with minimal ℓ_1-norm among the shortest vectors. So we have $y' = y$ and $|by - ax'| = |by - ax|$. Assume without loss of generality that $x < x'$ holds. The numbers x and x' have been chosen such that

$$|yb/a - x| = |yb/a - x'| = \min_{z \in \mathbf{N}_0} |yb/a - z|$$

holds. Thus we conclude that $x' = x + 1$ and that $p/q = b/a = (2x+1)/2y$. Suppose there is a prime $\ell > 2$ dividing both $(2x+1)$ and $2y$. Let $u = (2x+1)/\ell$ and $v = 2y/\ell$. Then

$$\begin{aligned}
\left\| \left(\begin{smallmatrix} -ua+vb \\ vc \end{smallmatrix}\right) \right\| &= 1/\ell \left\| \left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right) + \left(\begin{smallmatrix} -(x+1)a+yb \\ yc \end{smallmatrix}\right) \right\| \\
&\leqslant 1/\ell \left(\left\| \left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right) \right\| + \left\| \left(\begin{smallmatrix} -(x+1)a+yb \\ yc \end{smallmatrix}\right) \right\| \right) \\
&= 2/\ell \left\| \left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right) \right\| < \left\| \left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right) \right\|,
\end{aligned}$$

a contradiction. Thus $\gcd(2x+1, 2y) = 1$ which implies $2x+1 = p$ and $2y = q$ and finishes the proof. □

Lemma 1 reveals that one can find a shortest vector with the classical extended euclidean algorithm.

A naive method would work as follows. First we compute the reduced representation p/q of b/a. Then we compute the vectors $(-\lfloor p/2 \rfloor a + \lfloor q/2 \rfloor b, \lfloor q/2 \rfloor c)^T$ and $(a, 0)^T$. We store the shortest one of the two vectors in a container MIN. Then we compute successively the convergents g_k/h_k of b/a with EXGCD(b, a) and compare the length of the induced vector $(-g_k a + h_k b, h_k c)^T$ with MIN. If it is shorter, we replace MIN by $(-g_k a + h_k b, h_k c)^T$. In the end MIN contains a shortest vector. This algorithm would require a linear search through all convergents of b/a. In the next section we show a substantial improvement.

5 Finding a shortest vector with respect to ℓ_∞

Let Λ be given by its HNF $\left(\begin{smallmatrix} a & b \\ 0 & c \end{smallmatrix}\right) \in \mathbf{Z}^{2\times 2}$, where $c > 0$ and $a > b \geqslant 0$. In this section, we identify two candidate convergents of b/a that come into question to form a shortest vector and we apply the result of Schönhage (1971) to find them. Throughout this section, we consider only shortest vectors w.r.t. the ℓ_∞-norm.

Consider the set of vectors

$$\left\{ \begin{pmatrix} -g_k a + h_k b \\ h_k c \end{pmatrix} \mid k = 0, \dots, t \right\}, \tag{3}$$

where $\beta_k = g_k/h_k$, $0 \leqslant k \leqslant t$ are the convergents of b/a.

Proposition 2. *The shortest vector in* (3) *w.r.t.* ℓ_∞ *is the last convergent of* b/a *that lies outside the interval* $[(b-c)/a, (b+c)/a]$ *or the first convergent of* b/a *that lies inside* $[(b-c)/a, (b+c)/a]$.

Proof. The absolute value of the first component of the vectors $\left(\begin{smallmatrix} -g_k a + h_k b \\ h_k c \end{smallmatrix}\right)$, $k = 0, \dots, t$ is decreasing, since each convergent of b/a is a good approximation of b/a. The absolute value of the second components is increasing for growing k. We have to determine the first k, for which the absolute value of the second component of $\left(\begin{smallmatrix} -g_k a + h_k b \\ h_k c \end{smallmatrix}\right)$ is larger than the absolute value of the first component. Either this, or the previous k, is the k of the shortest vector. But $|-g_k a + h_k b| \leqslant h_k c$ if and only if $|b/a - g_k/h_k| \leqslant c/a$. $\square$

In the next proposition we show that the common convergent of the interval $[(b-c)/a, (b+c)/a]$ is a good starting point for the convergent of b/a which is "shortest" in (3).

Proposition 3. *Let* $\beta_k = g_k/h_k$ *be the common convergent of* $(b-c)/a$ *and* $(b+c)/a$*. Then the* k*-th,* $k+1$*-st or the* $k+2$*-nd convergent of* b/a *is a shortest vector in* (3) *w.r.t. the* ℓ_∞*-norm.*

Proof. Assume that k is even, the proof is analogous for odd k. Then $\beta_k \leqslant (b-c)/a$. If $\beta_k = (b-c)/a$, then $\left(\begin{smallmatrix} -g_k a + h_k b \\ h_k c \end{smallmatrix}\right)$ is a shortest vector in (3) since the absolute values of the first and second components are equal. So assume that $\beta_k < (b-c)/a$.

Let $\beta_{k+1}^{(i)} = g_{k+1}^{(i)}/h_{k+1}^{(i)}$, $i = 1,2,3$ be the $k+1$-st convergent of the numbers $(b-c)/a$, b/a and $(b+c)/a$ respectively. We show now that β_k or $\beta_{k+1}^{(2)}$ is the last convergent of b/a which is not in $[(b-c)/a, (b+c)/a]$. The claim follows then from Proposition 2.

Suppose $\beta_{k+1}^{(2)}$ is not in $[(b-c)/a, (b+c)/a]$. Then one has $(b-c)/a \leqslant \beta_{k+1}^{(1)} < b/a$ and $(b+c)/a \leqslant \beta_{k+1}^{(2)} = \beta_{k+1}^{(3)}$. Let $a_1 > a_2 \in \mathbf{N}_+$ be the numbers in $\mathbf{N}_+$ with

$$h_{k+1}^{(1)} = h_{k-1} + a_1 h_k \text{ and } h_{k+1}^{(2)} = h_{k-1} + a_2 h_k.$$

Since the sequence $\beta(x) = (g_{k-1} + xg_k)/(h_{k-1} + xh_k)$, $x \in \mathbf{N}_+$ is decreasing and since a_2 is maximal with $b/a \leqslant \beta(a_2)$ and since $(b-c)/a \leqslant \beta(a_1) < b/a$ we see that $\beta(a_2 + 1) \in [(b-c)/a, b/a]$. Let $h^{(2)}_{k+2}$ be the denominator of the $k+2$-nd convergent of b/a. One has

$$h^{(2)}_{k+2} \geqslant h_k + h_{k-1} + a_2 h_k = h_{k-1} + (a_2+1)h_k.$$

Since each convergent of b/a is a good approximation to b/a, the $k+2$-nd convergent of b/a has to lie in $[(b-c)/a, (b+c)/a]$. □

These observations show that the classical result of Schönhage (1971) can be used to compute a shortest vector of a lattice.

Corollary 4. *There exists an algorithm that computes in time* $\mathrm{O}(M(n)\log n)$ *a basis* B *of a 2-dimensional integral lattice* Λ *defined by* $A \in \mathbf{Z}^{2\times 2}$*, with the property that the first column of* B *is a shortest vector of* Λ *w.r.t. the* ℓ_∞*-norm.*

Proof. First we compute the HNF $\left(\begin{smallmatrix} a & b \\ 0 & c \end{smallmatrix}\right)$ of A. Next, we compute the reduced representation p/q of b/a. Then we compute the vectors $(a,0)^T$, $(-\lfloor p/2 \rfloor a + \lfloor q/2 \rfloor b, \lfloor q/2 \rfloor c)^T$ and store a shortest nonzero one in a container MIN. Next we compute the common convergent β_k of $[(b-c)/a, (b+c)/a]$ and the corresponding matrix $R^{(k)}$. The next two convergents of b/a can then be computed as follows. We perform two runs through the while-loop of EXGCD on input ${R^{(k)}}^{-1} \left(\begin{smallmatrix} b \\ a \end{smallmatrix}\right)$ and we store the matrix $M^{(2)}$. The next two convergents β_{k+1} and β_{k+2} of b/a are then obtained from the matrix $R^{(k)}M^{(2)}$ according to (2). We replace MIN if one of the convergents yields a shorter vector. Lemma 1 and Proposition 3 imply that MIN then contains a shortest vector w.r.t. the ℓ_∞-norm.

If one has a shortest vector $\left(\begin{smallmatrix} -xa+yb \\ yc \end{smallmatrix}\right)$, then one computes two integers u and v with $\gcd(x,y) = 1 = uy - vx$. The matrix $\left(\begin{smallmatrix} -x & -u \\ y & v \end{smallmatrix}\right)$ is unimodular. Thus

$$\begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \begin{pmatrix} -x & -u \\ y & v \end{pmatrix}$$

is a basis of Λ whose first column vector consists of a shortest vector of Λ w.r.t. the ℓ_∞-norm.

It is easy to see that the described method runs in time $\mathrm{O}(M(n)\log n)$ if the algorithm of Schönhage (1971) is used. □

6 Finding a reduced basis

In this section, $\|\cdot\|$ denotes the ℓ_2-norm. Let $B \in \mathbf{Z}^{2\times 2}$ be a basis of Λ whose first column is a shortest vector of Λ w.r.t. the ℓ_∞-norm. Let C be a reduced basis according to (1). Then the first column of C is a shortest vector of Λ w.r.t. the ℓ_2-norm. Let $b^{(1)}$ and $c^{(1)}$ be the first columns of B and C respectively. It follows that $\sqrt{2}\,\|c^{(1)}\| \geqslant \|b^{(1)}\|$ holds, and thus that the basis B is "almost reduced".

Lagarias (1980, proof of Theorem 4.2) has shown that in this case the algorithm GAUSS requires only a constant number of runs through the repeat-loop to reduce B. We thus have the following corollary.

Corollary 5. *There exists an algorithm that computes in time* $\mathrm{O}(M(n)\log n)$ *a reduced basis* C *of a 2-dimensional integral lattice* Λ *defined by* $A \in \mathbf{Z}^{2\times 2}$.

Acknowledgements

Thanks to Alexander Bockmayr and Johannes Buchmann for valuable discussions. I am grateful to Hans de Nivelle and Volker Priebe for helpful remarks on this paper.

References

Aho, A. V., Hopcroft, J. E. & Ullman, J. D. (1974), *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading.

Bach, E. & Shallit, J. (1996), *Algorithmic number theory*, Vol. 1: efficient algorithms, MIT Press.

Daudé, H., Flajolet, P. & Vallée, B. (1997), 'An average-case analysis of the gaussian algorithm for lattice reduction', *Comb. Probab. Comput.* **6**(4), 397–433.

Frank, A. & Tardos, É. (1987), 'An application of simultaneous Diophantine approximation in combinatorial optimization', *Combinatorica* **7**, 49–65.

Gauß, C. F. (1801), *Disquisitiones arithmeticae*, Gerh. Fleischer Iun.

Grötschel, M., Lovász, L. & Schrijver, A. (1988), *Geometric Algorithms and Combinatorial Optimization*, Vol. 2 of *Algorithms and Combinatorics*, Springer.

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

Khintchine, A. Y. (1963), *Continued Fractions*, Noordhoff, Groningen.

Lagarias, J. C. (1980), 'Worst-case complexity bounds for algorithms in the theory of integral quadratic forms', *Journal of Algorithms* **1**, 142–186.

Lenstra, A. K., Lenstra, H. W. & Lovász, L. (1982), 'Factoring polynomials with rational coefficients', *Math. Annalen* **261**, 515 – 534.

Lenstra, H. W. (1983), 'Integer programming with a fixed number of variables', *Mathematics of Operations Research* **8**(4), 538 – 548.

Perron, O. (1954), *Die Lehre von den Kettenbrüchen*, 3-rd edition edn, Teubner.

Rote, G. (1997), 'Finding a shortest vector in a two-dimensional lattice modulo m', *Theoretical Computer Science* **172**(1–2), 303–308.

Schönhage, A. (1971), 'Schnelle Berechnung von Kettenbruchentwicklungen. (Speedy computation of expansions of continued fractions)', *Acta Informatica* **1**, 139–144.

Schönhage, A. (1991), Fast reduction and composition of binary quadratic forms, *in* 'Proceedings of ISSAC 91, Bonn, Germany', ACM Press, pp. 128–133.

Schrijver, A. (1986), *Theory of Linear and Integer Programming*, John Wiley.

Vallée, B. (1991), 'Gauss' algorithm revisited', *Journal of Algorithms* **12**(4), 556–572.

Yap, C. K. (1992), Fast unimodular reduction: Planar integer lattices, *in* 'Proceedings of the 33rd Annual Symposium on Foundations of Computer Science', IEEE Computer Society Press, Pittsburgh, pp. 437–446.

Yap, C. K. (1999), *Fundamental Problems of Algorithmic Algebra*, Oxford University Press.

INFORMATIK

Below you find a list of the most recent technical reports of the Max-Planck-Institut für Informatik. They are available by anonymous ftp from `ftp.mpi-sb.mpg.de` under the directory `pub/papers/reports`. Most of the reports are also accessible via WWW using the URL `http://www.mpi-sb.mpg.de`. If you have any questions concerning ftp or WWW access, please contact `reports@mpi-sb.mpg.de`. Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max-Planck-Institut für Informatik
Library
attn. Anja Becker
Im Stadtwald
66123 Saarbrücken
GERMANY
e-mail: `library@mpi-sb.mpg.de`

MPI-I-2000-4-001	J. Kautz, W. Heidrich, K. Daubert	Bump Map Shadows for OpenGL Rendering
MPI-I-2000-2-001	F. Eisenbrand	Short Vectors of Planar Lattices Via Continued Fractions
MPI-I-2000-1-002	R. Beier, J. Sibeyn	A Powerful Heuristic for Telephone Gossiping
MPI-I-2000-1-001	E. Althaus, O. Kohlbacher, H. Lenhof, P. Müller	A branch and cut algorithm for the optimal solution of the side-chain placement problem
MPI-I-1999-4-001	J. Haber, H. Seidel	A Framework for Evaluating the Quality of Lossy Image Compression
MPI-I-1999-3-005	T.A. Henzinger, J. Raskin, P. Schobbens	Axioms for Real-Time Logics
MPI-I-1999-3-004	J. Raskin, P. Schobbens	Proving a conjecture of Andreka on temporal logic
MPI-I-1999-3-003	T.A. Henzinger, J. Raskin, P. Schobbens	Fully Decidable Logics, Automata and Classical Theories for Defining Regular Real-Time Languages
MPI-I-1999-3-002	J. Raskin, P. Schobbens	The Logic of Event Clocks
MPI-I-1999-3-001	S. Vorobyov	New Lower Bounds for the Expressiveness and the Higher-Order Matching Problem in the Simply Typed Lambda Calculus
MPI-I-1999-2-008	A. Bockmayr, F. Eisenbrand	Cutting Planes and the Elementary Closure in Fixed Dimension
MPI-I-1999-2-007	G. Delzanno, J. Raskin	Symbolic Representation of Upward-closed Sets
MPI-I-1999-2-006	A. Nonnengart	A Deductive Model Checking Approach for Hybrid Systems
MPI-I-1999-2-005	J. Wu	Symmetries in Logic Programs
MPI-I-1999-2-004	V. Cortier, H. Ganzinger, F. Jacquemard, M. Veanes	Decidable fragments of simultaneous rigid reachability
MPI-I-1999-2-003	U. Waldmann	Cancellative Superposition Decides the Theory of Divisible Torsion-Free Abelian Groups
MPI-I-1999-2-001	W. Charatonik	Automata on DAG Representations of Finite Trees
MPI-I-1999-1-007	C. Burnikel, K. Mehlhorn, M. Seel	A simple way to recognize a correct Voronoi diagram of line segments
MPI-I-1999-1-006	M. Nissen	Integration of Graph Iterators into LEDA
MPI-I-1999-1-005	J.F. Sibeyn	Ultimate Parallel List Ranking ?
MPI-I-1999-1-004	M. Nissen, K. Weihe	How generic language extensions enable “open-world” desing in Java
MPI-I-1999-1-003	P. Sanders, S. Egner, J. Korst	Fast Concurrent Access to Parallel Disks
MPI-I-1999-1-002	N.P. Boghossian, O. Kohlbacher, H.-. Lenhof	BALL: Biochemical Algorithms Library
MPI-I-1999-1-001	A. Crauser, P. Ferragina	A Theoretical and Experimental Study on the Construction of Suffix Arrays in External Memory
MPI-I-98-2-018	F. Eisenbrand	A Note on the Membership Problem for the First Elementary Closure of a Polyhedron

MPI-I-98-2-017	M. Tzakova, P. Blackburn	Hybridizing Concept Languages
MPI-I-98-2-014	Y. Gurevich, M. Veanes	Partisan Corroboration, and Shifted Pairing
MPI-I-98-2-013	H. Ganzinger, F. Jacquemard, M. Veanes	Rigid Reachability
MPI-I-98-2-012	G. Delzanno, A. Podelski	Model Checking Infinite-state Systems in CLP
MPI-I-98-2-011	A. Degtyarev, A. Voronkov	Equality Reasoning in Sequent-Based Calculi
MPI-I-98-2-010	S. Ramangalahy	Strategies for Conformance Testing
MPI-I-98-2-009	S. Vorobyov	The Undecidability of the First-Order Theories of One Step Rewriting in Linear Canonical Systems
MPI-I-98-2-008	S. Vorobyov	AE-Equational theory of context unification is Co-RE-Hard
MPI-I-98-2-007	S. Vorobyov	The Most Nonelementary Theory (A Direct Lower Bound Proof)
MPI-I-98-2-006	P. Blackburn, M. Tzakova	Hybrid Languages and Temporal Logic
MPI-I-98-2-005	M. Veanes	The Relation Between Second-Order Unification and Simultaneous Rigid *E*-Unification
MPI-I-98-2-004	S. Vorobyov	Satisfiability of Functional+Record Subtype Constraints is NP-Hard
MPI-I-98-2-003	R.A. Schmidt	E-Unification for Subsystems of S4
MPI-I-98-2-002	F. Jacquemard, C. Meyer, C. Weidenbach	Unification in Extensions of Shallow Equational Theories
MPI-I-98-1-031	G.W. Klau, P. Mutzel	Optimal Compaction of Orthogonal Grid Drawings
MPI-I-98-1-030	H. Brönniman, L. Kettner, S. Schirra, R. Veltkamp	Applications of the Generic Programming Paradigm in the Design of CGAL
MPI-I-98-1-029	P. Mutzel, R. Weiskircher	Optimizing Over All Combinatorial Embeddings of a Planar Graph
MPI-I-98-1-028	A. Crauser, K. Mehlhorn, E. Althaus, K. Brengel, T. Buchheit, J. Keller, H. Krone, O. Lambert, R. Schulte, S. Thiel, M. Westphal, R. Wirth	On the performance of LEDA-SM
MPI-I-98-1-027	C. Burnikel	Delaunay Graphs by Divide and Conquer
MPI-I-98-1-026	K. Jansen, L. Porkolab	Improved Approximation Schemes for Scheduling Unrelated Parallel Machines
MPI-I-98-1-025	K. Jansen, L. Porkolab	Linear-time Approximation Schemes for Scheduling Malleable Parallel Tasks
MPI-I-98-1-024	S. Burkhardt, A. Crauser, P. Ferragina, H. Lenhof, E. Rivals, M. Vingron	q-gram Based Database Searching Using a Suffix Array (QUASAR)
MPI-I-98-1-023	C. Burnikel	Rational Points on Circles
MPI-I-98-1-022	C. Burnikel, J. Ziegler	Fast Recursive Division
MPI-I-98-1-021	S. Albers, G. Schmidt	Scheduling with Unexpected Machine Breakdowns
MPI-I-98-1-020	C. Rüb	On Wallace's Method for the Generation of Normal Variates
MPI-I-98-1-019		2nd Workshop on Algorithm Engineering WAE '98 - Proceedings
MPI-I-98-1-018	D. Dubhashi, D. Ranjan	On Positive Influence and Negative Dependence
MPI-I-98-1-017	A. Crauser, P. Ferragina, K. Mehlhorn, U. Meyer, E. Ramos	Randomized External-Memory Algorithms for Some Geometric Problems
MPI-I-98-1-016	P. Krysta, K. Loryś	New Approximation Algorithms for the Achromatic Number