The Generalized Gauss Reduction Algorithm

MICHAEL KAIB* and CLAUS P. SCHNORR

Fachbereich Mathematik / Informatik, Universität Frankfurt
Postfach 11 19 32, 60054 Frankfurt a.M., Germany

January 28, 1994

Abstract

We generalize the Gauss algorithm for the reduction of two–dimensional lattices from the l_2-norm to arbitrary norms and extend Vallée's analysis [J. Algorithms 12 (1991), 556-572] to the generalized algorithm.

1 Introduction

Gauss [Ga1801] gave, in the language of qaudratic forms, an algorithm which reduces a basis a, b of a two–dimensional lattice and finds the two successive minima of the lattice. Vallée [Va91] shows that the Gauss reduction algorithm performs at most $\log_{1+\sqrt{2}}(\frac{2\sqrt{2}}{3} \parallel a \parallel_2^2 + \parallel b \parallel_2^2) + 2$ many iterations. This bound is optimal up to an additive constant. Vallée also characterizes for the lattice $\mathbb{Z}^2$ the minimal size input bases for which the Gauss algorithm performs exactly k iterations. The bit complexity of the Gauss algorithm has been studied by Schönhage [Sch91] in the language of quadratic forms.

While these results are all for the l_2-norm, other norms are important, too. The l_∞-norm is the natural norm for integer programming problems. Schnorr [Sch93] reduces the problem of factoring integers to a closest lattice vector problem in the l_1-norm. Lovász and Scarf [LS92] propose a generalized basis reduction algorithm that extends the L^3-algorithm of A.K. Lenstra, H.W. Lenstra Jr. and L. Lovasz [LLL82] to an arbitrary norm.

*e-mail: kaib@informatik.uni-frankfurt.de

Our results. We extend the Gauss reduction algorithm from the l_2-norm to an arbitrary norm. This generalized Gauss algorithm (gGA) essentially coincides with the Lovász–Scarf algorithm for two–dimensional lattice bases. The gGA finds for any norm the two successive minima of the lattice. Given a reduced basis we exhibit minimal size input bases requiring a given number of iterations. These minimal size input bases represent the worst case inputs. They are *universally* worst case for all norms for which the given output basis is reduced. They satisfy the same recursion which holds true for the worst case inputs of the centered Euclidean algorithm according to Dupré [Du1846].

We show that the generalized Gauss algorithm terminates for any norm after at most $\log_{1+\sqrt{2}}(2\sqrt{2}\,B/\lambda_2) + o(1)$ many iterations, where B is the maximum of the norms of the two input vectors and λ_2 is the second successive minimum of the lattice with respect to the given norm.

The paper is organized as follows. In Section 2 we introduce reduced lattice bases. In section 3 we present the generalized Gauss algorithm and its analysis. Section 4 gives complexity bounds for the RAM–model. A preliminary version of this paper has been published by Kaib [Ka91].

2 Geometrical preliminaries

We generalize the concept of reduced lattice bases for lattices of rank 2 to an arbitrary norm $\| \, . \, \|$ on $\mathbb{R}^n$. We use the following three elementary lemmata:

Lemma 1. *Let $a, b \in \mathbb{R}^n, a \neq 0$, let $F : \mathbb{R} \to \mathbb{R}^n : \xi \mapsto \xi a + b$ describe a line in $\mathbb{R}^n$ and $f(\xi) = \| F(\xi) \|$. Then f is a convex function.*

Lemma 2. *Let $F : \mathbb{R} \to \mathbb{R}^n$ be a line in $\mathbb{R}^n$ and $\xi_1, \xi_2, \eta_1, \eta_2$ be four reals with $\xi_1 < \xi_2$, $\eta_1 < \eta_2$, $\xi_1 \leq \eta_1$, $\xi_2 \leq \eta_2$. Then $\| F(\xi_1) \| \leq \| F(\xi_2) \|$ implies $\| F(\eta_1) \| \leq \| F(\eta_2) \|$, and $\| F(\xi_1) \| < \| F(\xi_2) \|$ implies $\| F(\eta_1) \| < \| F(\eta_2) \|$.*

We will usually apply Lemma 2 in the case $\xi_1 = 0$ and $\xi_2 = 1$.

Lemma 3. *Let M be a closed set in $\mathbb{R}^n$ and $0 \notin M$. Then every point in M with minimal norm lies on the boundary of M.*

Throughout the paper let $(a, b) \in \mathbb{R}^n \times \mathbb{R}^n$ be basis of the two–dimensional lattice $L = \mathbb{Z}a + \mathbb{Z}b$. We define reduced and well–ordered lattice bases. The reduction algorithm in the next section recurs on well–ordered bases until a reduced basis is found.

Definition. *A lattice basis* (a, b) *is called*

reduced *if* $\quad \| a \|, \| b \| \leq \| a - b \| \leq \| a + b \|$ *and*

well–ordered *if* $\quad \| a \| \leq \| a - b \| < \| b \|$.

By Lemma 2 $\| a - b \| < \| b \|$ implies

$$\| b \| < \| \eta a + b \| \quad \forall \eta > 0 . \tag{1}$$

Thus (a, b) is well–ordered iff $\| a \| \leq \| a - b \| < \| b \| < \| a + b \|$.

The *i–th successive minimum* λ_i of a lattice L with respect to the norm $\| . \|$ is defined as the minimal real ρ such that there are at least i linearly independent lattice vectors of norm at most ρ.

Theorem 4. *If* (a, b) *is a reduced basis then* $\| a \|, \| b \|$ *are the two successive minima of the lattice* $L = \mathbb{Z}a + \mathbb{Z}b$.

Proof. W.l.o.g. let $\| a \| \leq \| b \|$. The theorem claims the following:

$$\| a \| \leq \| ra + sb \| \quad \text{for all } (r, s) \in \mathbb{Z}^2 - \{(0, 0)\} ,$$

$$\| b \| \leq \| ra + sb \| \quad \text{for all } r \in \mathbb{Z}, s \in \mathbb{Z} - \{0\} .$$

These inequalities follow from the inequalities

$$\begin{aligned} \| a \| &\leq \| b \| , \\ \| a \| &\leq \| ra \| \quad \text{for all } r \in \mathbb{Z} - \{0\} , \\ \| b \| &\leq \| \xi a + \eta b \| \quad \text{for all } \xi, \eta \in \mathbb{R} \text{ with } |\xi|, |\eta| \geq 1 . \end{aligned} \tag{2}$$

It is therefore sufficient to prove Inequality 2. For this we show the following

Claim. *Consider the four dotted areas in Figure 1. The norm takes its minimum in each of the four dotted areas in the points* $\pm a \pm b$.

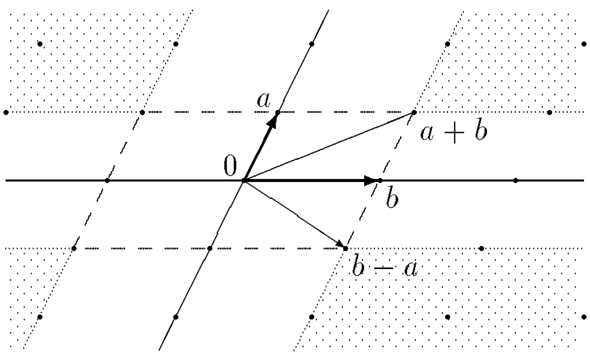

Figure 1: Reduced basis (a, b)

Inequality 2 is an immediate consequence of the claim and the reduction conditions $\| b \| \leq \| a \pm b \|$.

Proof of the claim. Each dashed line in the figure contains three lattice points where the middle point has minimal norm, i.e. we have

$$\| \pm a - b \| \;\geq\; \| \pm a \| \;\leq\; \| \pm a + b \| \,,$$

$$\| -a \pm b \| \;\geq\; \| \pm b \| \;\leq\; \| a \pm b \| \,.$$

Lemma 2 yields

$$\| \pm a \pm \xi b \| \;\geq\; \| \pm a \pm b \| \;\geq\; \| \pm a \| \,,$$

$$\| \pm \xi a \pm b \| \;\geq\; \| \pm a \pm b \| \;\geq\; \| \pm b \|$$

for $\xi \geq 1$. This proves that the points $\pm a \pm b$ have minimal norm for the dotted lines. By Lemma 3 the norm takes its minimum for each of the four dotted areas on the boundary, i.e. on the dotted lines. This proves the claim. □

3 Analysis of the generalized Gauss algorithm

We extend the Gauss basis reduction algorithm from the l_2-norm to an arbitrary norm. We choose the sign of the basis vectors in the algorithm so that the algorithm recurs on well–ordered bases. As a consequence all occurring integral reduction coefficients μ are positive.

The generalized Gauss algorithm (gGA).

INPUT a well–ordered lattice basis (a, b).

WHILE $\| b \| \;>\; \| a - b \|$ **DO**

1. $b := b - \mu a$,
   where the integer μ is chosen to minimize the norm $\| b - \mu a \|$.
2. **IF** $\| a + b \| \;<\; \| a - b \|$ **THEN** $b := -b$.
3. Swap a and b.

END WHILE

OUTPUT (a, b).

Comments.

1. The exchange in Step 3 produces either a well–ordered or a reduced basis. The algorithm traverses, upon exit of Step 3 (resp. entry of Step 1), a sequence of well–ordered bases until a reduced basis is produced.

2. The algorithm terminates after finitely many steps because the norm of the basis vectors decreases in every, except the last, iteration.

3. To have a well defined algorithm we require to choose in Step 1 the smallest μ that minimizes $\| b - \mu a \|$.

We associate with an input basis the sequence of lattice bases occurring in the algorithm upon exit of Step 3. The bases of this sequence are all well–ordered, except that the final basis is reduced. If (b, c), (a, b) are two consecutive bases in any of these sequences we call (a, b) the *successor basis* of (b, c) and (b, c) a *predecessor basis* of (a, b). A well–ordered basis has at most one successor basis but may have infinitely many predecessor bases corresponding to runs of the algorithm with various input bases. If (b, c) is a predecessor basis of (a, b) we call the vector c a *predecessor* of (a, b). The transition of a well–ordered basis (b, c) via Steps 1-3 to its successor basis (a, b) is of the form

$$\begin{aligned} (a, b) \;&:=\; (b, c)\begin{pmatrix} 1 & -\mu \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ &=\; (b,\, c)\begin{pmatrix} -\varepsilon\mu & 1 \\ \varepsilon & 0 \end{pmatrix} \;=\; (\,\varepsilon(c - \mu b),\, b\,) \end{aligned} \tag{3}$$

where $\varepsilon = \pm 1$ denotes the possible change of sign in Step 2. We see that the predecessor c is of the form $c = \varepsilon a + \mu b$. The following lemma characterizes the predecessors of a well–ordered basis (a, b). It generalizes Lemma 1 of Vallée [Va91].

Lemma 5. *If (a, b) is a well–ordered lattice basis then a vector $c = \varepsilon a + \mu b$ is a predecessor of (a, b) if and only if either $\varepsilon = 1,\ \mu \geq 2$ or $\varepsilon = -1,\ \mu \geq 3$.*

Lemma 5 shows that the set of predecessor bases of (a, b) does not depend on the norm, i.e. if (a, b) is well–ordered for two distinct norms then its two sets of predecessors coincide.

Proof. Since (a, b) is well–ordered we have

$$\| a \| \;\leq\; \| a - b \| \;<\; \| b \| \,. \tag{4}$$

The predecessor basis (b, c) is well–ordered iff $\| b \| \;\leq\; \| b - c \| \;<\; \| c \|$.
We consider the lines

$$\begin{aligned} F(\xi) &= (1-\xi)\ (b-a) + \xi\ b \\ G(\xi) &= (1-\xi)\ a + \xi\ b \\ H^{+}(\xi) &= (1-\xi)\ a + \xi\ (a+b) \\ H^{-}(\xi) &= (1-\xi)\ {-a} + \xi\ (b-a) \end{aligned}$$

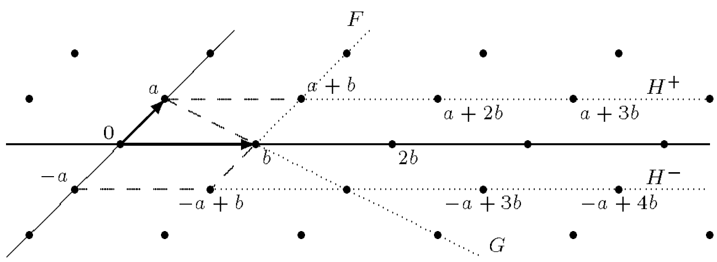

Figure 2: Well-ordered basis (a, b)

Inequality 4 implies:

$$\begin{array}{rcccccc} \| F(0) \| & = & \| b-a \| & < & \| b \| & = & \| F(1) \| \\ \| G(0) \| & = & \| a \| & < & \| b \| & = & \| G(1) \| \\ \| H^-(0) \| & = & \| a \| & \leq & \| b-a \| & = & \| H^-(1) \| . \end{array}$$

Thus by Lemma 2 $\| F(\xi) \|$ and $\| G(\xi) \|$ is strictly increasing and $\| H^-(\xi) \|$ is increasing (i.e. non–decreasing) for $\xi \geq 1$. This yields a corresponding inequality for H^+:

$$\begin{array}{rcl} \| H^+(0) \| & = & \| G(0) \| \\ & < & \| G(1) \| = \| F(1) \| \\ & < & \| F(2) \| = \| H^+(1) \| . \end{array}$$

We decide for all possible cases of μ and $\varepsilon = \pm 1$ whether (b, c) is well–ordered.

$\varepsilon = 1,\ \mu \leq -1$: Then $\| b-c \| = \| (1-\mu)b - a \| = \| H^-(1-\mu) \| \geq \| H^-(2) \| = \| 2b-a \| = \| G(2) \| > \| G(1) \| = \| b \|$. Thus (b, c) is not well–ordered.

$\varepsilon = -1,\ \mu \leq 0$: Then $\| b-c \| = \| (1-\mu)b + a \| = \| H^+(1-\mu) \| \geq \| H^+(1) \| = \| F(2) \| > \| F(1) \| = \| b \|$. Thus (b, c) is not well–ordered.

$\mu = 0$: Then $\| c \| = \| a \| < \| b \|$ and (b, c) is not well–ordered.

$\mu = 1$: Then $\| b-c \| = \| a \| < \| b \|$ and (b, c) is not well–ordered.

$\varepsilon = -1,\ \mu = 2$: Then $\| b-c \| = \| a-b \| < \| b \|$ and (b, c) is not well–ordered.

$\varepsilon = 1,\ \mu \geq 2$: Then $\| c \| = \| a + \mu b \| = \| H^+(\mu) \| > \| H^+(\mu-1) \| = \| a + (\mu-1)b \| = \| b-c \| \geq \| H^+(1) \| = \| F(2) \| > \| F(1) \| = \| b \|$. Thus (b, c) is well–ordered.

$\varepsilon = -1,\ \mu \geq 3$: Then $\| c \| = \| -a + \mu b \| = \| H^-(\mu) \| \geq \| H^-(\mu-1) \| = \| b-c \| \geq \| H^-(2) \| = \| G(2) \| > \| G(1) \| = \| b \|$. Thus (b, c) is well–ordered. □

For the analysis of the algorithm we consider the sequence of well–ordered bases that is traversed upon exit of Step 3. We index this sequence in reverse order $(b_0, b_1), \ldots, (b_k, b_{k+1})$ so that (b_k, b_{k+1}) is the input basis, (b_0, b_1) is the last well–ordered basis and (b_{-1}, b_0) is the reduced output basis. Let ε_i, μ_i be the coefficients which, according to Equation 3, transform (b_i, b_{i+1}) into the successor basis (b_{i-1}, b_i). We have

$$(b_{i-1}, b_i) = (b_i, b_{i+1}) \begin{pmatrix} -\varepsilon_i \mu_i & 1 \\ \varepsilon_i & 0 \end{pmatrix}$$

$$(b_k, b_{k+1}) = (b_0, b_1) \begin{pmatrix} 0 & \varepsilon_1 \\ 1 & \mu_1 \end{pmatrix} \begin{pmatrix} 0 & \varepsilon_2 \\ 1 & \mu_2 \end{pmatrix} \cdots \begin{pmatrix} 0 & \varepsilon_k \\ 1 & \mu_k \end{pmatrix}$$

The latter matrix product can be expressed by the *generalized continuants* which Rieger [R78] has introduced for the analysis of the centered Euclidean algorithm. These polynomials

$$\begin{bmatrix} x_2 \dots x_n \\ y_1 \; \dots \; y_n \end{bmatrix}_n \in \mathbb{Z}[x_2, \dots, x_n, y_1, \dots, y_n]$$

are recursively defined as

$$\begin{bmatrix} \; \end{bmatrix}_{-1} = 0, \quad \begin{bmatrix} \; \end{bmatrix}_0 = 1, \quad \begin{bmatrix} \\ y_1 \end{bmatrix}_1 = y_1,$$

$$\begin{bmatrix} x_2 \dots x_n \\ y_1 \; \dots \; y_n \end{bmatrix}_n = y_1 \begin{bmatrix} x_3 \dots x_n \\ y_2 \; \dots \; y_n \end{bmatrix}_{n-1} + x_2 \begin{bmatrix} x_4 \dots x_n \\ y_3 \; \dots \; y_n \end{bmatrix}_{n-2} . \tag{5}$$

An easy induction shows that

$$\begin{pmatrix} 0 & \varepsilon_1 \\ 1 & \mu_1 \end{pmatrix} \cdots \begin{pmatrix} 0 & \varepsilon_k \\ 1 & \mu_k \end{pmatrix} = \begin{pmatrix} \varepsilon_1 \begin{bmatrix} \varepsilon_3 \dots \varepsilon_{k-1} \\ \mu_2 \dots \mu_{k-1} \end{bmatrix}_{k-2} & \varepsilon_1 \begin{bmatrix} \varepsilon_3 \dots \varepsilon_k \\ \mu_2 \dots \mu_k \end{bmatrix}_{k-1} \\ \begin{bmatrix} \varepsilon_2 \dots \varepsilon_{k-1} \\ \mu_1 \dots \mu_{k-1} \end{bmatrix}_{k-1} & \begin{bmatrix} \varepsilon_2 \dots \varepsilon_k \\ \mu_1 \dots \mu_k \end{bmatrix}_k \end{pmatrix} .$$

Hence

$$b_{k+1} = \varepsilon_1 \begin{bmatrix} \varepsilon_3 \dots \varepsilon_k \\ \mu_2 \; \dots \; \mu_k \end{bmatrix}_{k-1} b_0 + \begin{bmatrix} \varepsilon_2 \dots \varepsilon_k \\ \mu_1 \; \dots \; \mu_k \end{bmatrix}_k b_1 . \tag{6}$$

There is a simple formula for the continuants with $\mu_j = 2, \; \varepsilon_j = 1$:

$$T_i := \begin{bmatrix} 1 \dots 1 \\ 2 \; \dots \; 2 \end{bmatrix}_i = \frac{1}{2\sqrt{2}} \, [(1+\sqrt{2})^{i+1} - (1-\sqrt{2})^{i+1}] . \tag{7}$$

Simultaneous induction on i, via Equation 5, yields the following inequalities:

Lemma 6. *Let $\mu_j \geq 2$ for $\varepsilon_j = 1$, and $\mu_j \geq 3$ for $\varepsilon_j = -1$. Then*

$$\begin{bmatrix} \varepsilon_2 \dots \varepsilon_i \\ \mu_1 \; \dots \; \mu_i \end{bmatrix}_i \geq T_i , \quad \begin{bmatrix} \varepsilon_2 \dots \varepsilon_i \\ \mu_1 \; \dots \; \mu_i \end{bmatrix}_i \geq 2 \begin{bmatrix} \varepsilon_3 \dots \varepsilon_i \\ \mu_2 \; \dots \; \mu_i \end{bmatrix}_{i-1} .$$

Lemma 7. *Every sequence of successive well–ordered bases $(b_0, b_1), \dots, (b_k, b_{k+1})$ satisfies $\| b_{k+1} \| \geq T_k \| b_1 \|$.*

Proof. We see from Lemma 6 that the coefficient of b_1 in Equation 6 is positive and the coefficient of b_0 has sign ε_1. We distinguish two cases:

Case 1. $\varepsilon_1 = 1$. We have

$$\| b_{k+1} \| \geq \begin{bmatrix} \varepsilon_2 \dots \varepsilon_k \\ \mu_1 \; \dots \; \mu_k \end{bmatrix}_k \| b_1 \| \geq T_k \| b_1 \| .$$

The first inequality follows from Equation 6 by Inequality 1 since (b_0, b_1) is well–ordered. The second inequality holds by Lemma 6.

Case 2. $\varepsilon_1 = -1$. We have $\| b_0 \| < \| b_1 \|$ since (b_0, b_1) is well–ordered. Therefore Equation 6 and the triangular inequality yields

$$\| b_{k+1} \| \ \geq \ \left(\begin{bmatrix} \varepsilon_2 \ldots \varepsilon_k \\ \mu_1 \ \ldots \ \mu_k \end{bmatrix}_k - \begin{bmatrix} \varepsilon_3 \ldots \varepsilon_k \\ \mu_2 \ \ldots \ \mu_k \end{bmatrix}_{k-1} \right) \| b_1 \| \ .$$

The right-hand factor can be simplified to

$$\begin{bmatrix} \varepsilon_2 \ldots \varepsilon_k \\ \mu_1 \ \ldots \ \mu_k \end{bmatrix}_k - \begin{bmatrix} \varepsilon_3 \ldots \varepsilon_k \\ \mu_2 \ \ldots \ \mu_k \end{bmatrix}_{k-1} = \begin{bmatrix} \varepsilon_2 \ldots \varepsilon_k \\ (\mu_1 - 1) \ \ldots \ \mu_k \end{bmatrix}_k \ .$$

To verify this equation develop its first and last term via Equation 5. Finally the claim follows from Lemma 6 since $\mu_1 \geq 3$. □

We consider the number of *iterations* of the gGA or equivalently the number of traversed well–ordered bases. We bound this number as a function of $B := \| b \| / \lambda_2$ where b is the largest input vector.

Theorem 8. *The gGA performs on input (a, b), for $B = \| b \| / \lambda_2 \to \infty$, at most $\log_{1+\sqrt{2}}(2\sqrt{2}\, B) + o(1)$ many iterations where λ_2 is the second successive minimum of the lattice.*

Remark. If the input basis is not well–ordered there may be an extra iteration. The o–term is at most $2 - \log_{1+\sqrt{2}}(4\sqrt{2}) \approx 0.0339$ where the maximum occurs in case of a single iteration.

Proof. Let (b_k, b_{k+1}) be a well–ordered input basis and (b_{-1}, b_0) the output basis. There are $k+1$ iterations. Lemma 7 tells us that

$$T_k \ \leq \ \frac{\| b_{k+1} \|}{\| b_1 \|} \ \leq \ B \ .$$

Equation 7 implies

$$\frac{1}{2\sqrt{2}} \left[(1+\sqrt{2})^{k+1} - (1-\sqrt{2})^{k+1} \right] \ \leq \ B \ ,$$

and thus we have

$$k + 1 \ \leq \ \log_{1+\sqrt{2}}(2\sqrt{2}\, B) \ + \ o(1)$$

for $B \to \infty$. □

The bound of Theorem 8 is optimal for all norms:

Theorem 9. *Let (b_{-1}, b_0) be a reduced basis with $\| b_{-1} \| \ \leq \ \| b_0 \|$ and $b_{i+1} = b_{i-1} + 2b_i$ for $i = 0, \ldots, k$. Then the gGA performs on input (b_k, b_{k+1}) exactly $k+1$ iterations where $k + 1 \ \geq \ \log_{1+\sqrt{2}}(2\, B) \ - 1 \ + \ o(1)$ and $B = \| b_{k+1} \| / \lambda_2$.*

Remarks.

- The difference of the upper and lower bound in the above two theorems is $1 + \log_{1+\sqrt{2}}(\sqrt{2}) + o(1) \;\approx\; 1.393 + o(1)$.

- In the particular case that b_{-1}, b_0 are the integers $b_{-1} = 0,\ b_0 = 1$ the recursion $b_{i+1} = b_{i-1} + 2b_i$ for $i = 0, \ldots, k$ yields, according to Dupré [Du1846], the minimal integers b_k, b_{k+1} for which the centered Euclidean algorithm performs exactly $k+1$ divisions. Vallée [Va91] has extended this minimality, in the case of the l_2-norm, to bases of the lattice $\mathbb{Z}^2$.

- The novelty in Theorem 9 is that the recursion $b_{i+1} = b_{i-1} + 2b_i$ for $0 = 1, \ldots, k$ is valid for all norms, all lattices and all reduced output bases (b_{-1}, b_0).

For the proof we characterize the well–ordered predecessor bases $(b, c),\ c = \varepsilon a + \mu b$, of reduced bases (a, b). This extends Lemma 5.

Lemma 10. *Let (a, b) be a reduced basis and $c = \varepsilon a + \mu b$ where $\varepsilon = \pm 1$ and $\mu \in \mathbb{Z}$.*

1. *If $\| a \| \;\le\; \| b \|$ and $(\varepsilon, \mu) \ne (-1, 2)$ then (b, c) is well–ordered iff $\mu \ge 2$.*
2. *If $\| b \| \;\le\; \| a \|$ then (b, c) is not well–ordered for $\mu \le 0$ and well–ordered for $\varepsilon = 1,\ \mu > \frac{2\lambda_2}{\lambda_1} - 1$ and for $\varepsilon = -1,\ \mu > \frac{2\lambda_2}{\lambda_1}$ where $\lambda_i = \lambda_i(\mathbb{Z}a + \mathbb{Z}b)$.*

Proof of Theorem 9. The bases (b_i, b_{i+1}) for $i = 0, \ldots, k$ are all well–ordered. This holds by Lemma 10 for $i = 0$ and by Lemma 5 for $i > 0$. Hence the gGA performs on input b_k, b_{k+1} exactly $k + 1$ iterations with all reduction coefficients equal to 2, and then finds the reduced basis (b_{-1}, b_0). Equation 6 implies

$$b_{k+1} = T_{k-1} b_0 + T_k b_1 \ .$$

Thus we see from $\| b_0 \| \;=\; \lambda_2,\ \| b_1 \| \;\le\; 3\lambda_2$ and Equation 7 that

$$\begin{aligned} \| b_{k+1} \| &\le (T_{k-1} + 3T_k)\lambda_2 \\ &= (1+\sqrt{2})^k \, \frac{1 + 3(1+\sqrt{2})}{2\sqrt{2}} \, \lambda_2 \, (1 + o(1)) \ , \end{aligned}$$

hence $k \;\ge\; \log_{1+\sqrt{2}}(2\,B) - 2 + o(1)$. □

Proof of Lemma 10. We collect facts that cover all the claims. The basis (b, c) is well–ordered iff

$$\| b \| \;\le\; \| \varepsilon a + (\mu - 1) b \| \;<\; \| \varepsilon a + \mu b \| \ . \tag{8}$$

If $\| a \| \;\le\; \| b \|$ the left–hand inequality holds by Theorem 4 iff $\mu \ne 1$. This inequality is trivial for $\| b \| \;\le\; \| a \|$.

For the right–hand inequality we consider the line $H(\xi) = \varepsilon a + \xi b$, assuming that (a, b) is reduced. We have $\| H(-1) \| = \| \varepsilon a - b \| \geq \| a \| = \| H(0) \| \leq \| \varepsilon a + b \| = \| H(1) \|$. For $\mu \leq 0$ Lemma 2 implies

$$\| \varepsilon a + (\mu - 1)b \| = H(\mu - 1) \geq H(\mu) = \| \varepsilon a + \mu b \| .$$

Hence (b, c) is not well–ordered. This proves all claims in the case $\mu \leq 0$.

Now let $\mu \geq 1$. We have

$$\| \varepsilon a + (\mu - 1)b \| = H(\mu - 1) \leq H(\mu) = \| \varepsilon a + \mu b \| . \tag{9}$$

If $\varepsilon = -1$ we have $\| a \| \leq \lambda_2 \leq \| c \|$ which shows that

$$\| b \| = \frac{1}{\mu} \| a + c \| \leq \frac{2}{\mu} \| c \| . \tag{10}$$

If $\varepsilon = 1$ the inequality $\| b - a \| \leq \| a + b \| = H(1) \leq \| c \|$ implies

$$\| b \| = \frac{1}{\mu + 1} \| (b - a) + c \| \leq \frac{2}{\mu + 1} \| c \| . \tag{11}$$

Now assume that the left–hand Inequality 8 holds but (b, c) is not well–ordered. In this case equality must hold in Inequality 9. This implies that (b, c) is reduced and thus $\| b \| = \lambda_1$, $\| c \| = \lambda_2$. Moreover if $\| a \| \leq \| b \|$, i.e. the right–hand Inequality 8 does not hold, we have $\| b \| = \lambda_2 = \| c \|$, and thus the inequalities above imply $\mu \leq 1$ for $\varepsilon = 1$ and $\mu \leq 2$ for $\varepsilon = -1$. On the other hand, if $\| b \| \leq \| a \|$, Inequality 10 yields $\mu \leq 2\frac{\lambda_2}{\lambda_1}$ and Inequality 11 yields $\mu \leq 2\frac{\lambda_2}{\lambda_1} - 1$.

The claims in case $\mu \geq 1$. For those ε, μ where it is claimed that (b, c) is well–ordered we have shown, assuming that (b, c) is not well–ordered, an inequality excluding this μ. Moreover, in the case that $\| a \| \leq \| b \|$, we have shown that (b, c) is not well–ordered if $\mu = 1$. □

Remark. In the indefinite cases of Lemma 10, i.e.,

- $\| a \| \leq \| b \|$, $\varepsilon = -1$, $\mu = 2$,
- $\| b \| \leq \| a \|$, $\varepsilon = -1$, $1 \leq \mu \leq 2\frac{\lambda_2}{\lambda_1}$,
- $\| b \| \leq \| a \|$, $\varepsilon = 1$, $1 \leq \mu \leq 2\frac{\lambda_2}{\lambda_1} - 1$,

the above proof shows that (b, c) is *either* reduced *or* well–ordered. Both reduced and well–ordered bases do actually occur, as the norm and the lattice vary, in each of the three cases. However there is only one indefinite case for the Euclidean norm, namely $\| b \| \leq \| a \|$ and $\varepsilon = -1$, $\mu = 1$.

4 Time bounds

The generalized Gauss algorithm described in the last section needs acces to a norm oracle which for given $a \in \mathbb{R}^n$ outputs $\| a \|$. We give time bounds for the RAM model with the arithmetic operations multiplication, division, addition, subtraction, comparison and next integer computation at unit costs. We count for *steps* arithmetic steps and oracle calls. In this section we prove the following

Theorem 11. *Given an oracle for an arbitrary norm* $\| \,.\, \|$ *there is an algorithm which* $\| \,.\, \|$*-reduces a given basis* $a, b \in \mathbb{R}^n$ *using* $O(n \log (n + \lambda_2/\lambda_1) + \log B)$ *many steps where* $B = \max(\| a \|, \| b \|)/\lambda_2$.

Efficient $\| \,.\, \|$-reduction. For an efficient reduction of a basis $a, b \in \mathbb{R}^n$ in an arbitrary norm $\| \,.\, \|$ we first reduce a, b in the norm corresponding to a suitable inner product $<,>$ and we subsequently reduce the resulting basis in the $\| \,.\, \|$-norm. We initially perform a $<,>$-reduction since it only costs $O(1)$ arithmetic steps per iteration.

The inner product $<,>$ is chosen so that $\{x \in \mathbb{R}^n | \; \| x \| \;\leq\; 1\}$ is *spherical* in the sense that $\max_{x,y \in \mathbb{R}^n} \frac{\|x\| <y,y>^{1/2}}{\|y\| <x,x>^{1/2}} = O(n^{1.5})$. The existence of $<,>$ follows from [Le83], see construction of τ in Section 2, pp. 542 ff. We assume that the inner product is given, we do not count the steps for producing it. The constant $B_{<,>} = \max(<a,a>^{1/2}, \; <b,b>^{1/2})/\lambda_{2,<,>}$ satisfies $B_{<,>} = O(n^{1.5}\, B)$.

The initial $<,>$-reduction in $O(n + \log B)$ arithmetic steps. In each iteration we transform the Gram matrix $G = \begin{pmatrix} <a,a> & <a,b> \\ <b,a> & <b,b> \end{pmatrix}$ and the transformation matrix $H \in GL_n(\mathbb{Z})$ satisfying $(a_{current}, b_{current}) = (a_{input}, b_{input})H$ as $G := S^\top G S$, $H := HS$ where $S = \begin{pmatrix} -\varepsilon\mu & 1 \\ \varepsilon & 0 \end{pmatrix}$ and μ is the integer closest to $\frac{<a,b>}{<a,a>}$. Each iteration requires 6 multiplications, 6 subtractions, 1 division and 1 next integer computation. The initial (resp. final) transformations of (a, b) into G (resp. back from H into (a, b)) require $7n$ multiplications and $5n - 3$ additions. According to Theorem 8 the entire $<,>$-reduction of a, b is done in $O(\log B_{<,>}) = O(\log (B + n))$ iterations.

Computing μ in $O(1)$ resp. $O(\log \lambda_2/\lambda_1)$ oracle steps. Let $\mu : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{Z}$ denote the function that minimizes $\| b - \mu(a, b)\, a \|$. For a given well–ordered basis a, b we have to compute $\mu = \| b \pm x \| / \| a \|$ where (x, a) is the successor basis of (a, b). By the inequality $|\mu - \| b \| / \| a \|| \;\leq\; \| x \| / \| a \|$ we can compute μ, via the bisection method, using $O(\log \| x \| / \| a \|)$ oracle steps. Except for the final iteration we always have $\| x \| \;<\; \| a \|$ and the step bound is $O(1)$. For the final iteration we have $\| x \| / \| a \| \leq \lambda_2/\lambda_1$.

The final $\| \,.\, \|$-reduction in $O(n \log (n+\lambda_2/\lambda_1))$ steps. It follows from Theorem 8 that the final $\| \,.\, \|$-reduction requires at most $\log_{1+\sqrt{2}}(2\sqrt{2} \max_{x,y\in\mathbb{R}^n} \frac{\|x\| <y,y>^{1/2}}{\|y\| <x,x>^{1/2}}) + 1 + o(1) = O(\log n)$ many iterations. Every iteration, except the final one, requires $O(1)$ norm computations and $O(n)$ arithmetic steps. The final iteration costs $O(\log \lambda_2/\lambda_1)$ norm computations and $O(n \log \lambda_2/\lambda_1)$ arithmetic steps.

The case of the $l_1(l_\infty)$-norm. There are particularly efficient algorithms to compute μ for the l_1- and l_∞-norm. For the l_1-norm the real t minimizing $\| b - ta \|_1$ is the generalized median, with weights $|a_i|$, of the component fractions b_i/a_i for $i = 1, \ldots, n$ which can be computed using $O(n)$ arithmetic steps. For the l_∞-norm the graph of the function $\| b - ta \|_\infty$ with real indeterminate t is the maximum polygon of the $2n$ lines $\pm(b_i - ta_i)$. We sort, using $O(n \log n)$ arithmetic steps, these lines in order of descending gradient. A subsequent scan of the lines computes, using $O(n)$ arithmetic steps, the vertices of the polygon and in particular its minimal point which yields the real t that minimizes $\| b - t\,a \|_\infty$. Details can be found in [KS93].

Hence the $l_1(l_\infty)$-norm reduction of a, b takes at most $O(\log B + n \log n)$ arithmetic steps where $B = \max(\| a \|, \| b \|)/\lambda_2$ and $\| \,.\, \|$ is the $l_1(l_\infty)$-norm.

Acknowledgments

We are grateful to Joos Heintz for encouragement and critical comments.

References

[Da93] H. DAUDÉ: Des fractions continues a la réduction des réseaux: Analyse en moyenne. Thèse de doctorat, Université de Caen 1993.

[Du1846] A. DUPRÉ: Sur le nombre de divisions à effectuer pour obtenir le plus grand commun diviseur entre deux nombres entiers. J. de Math., vol. 11 (1846), pp. 41-64.

[Ga1801] C.F. GAUSS: Disquisitiones Arithmeticae. Leipzig 1801. German translation: Untersuchungen über die höhere Arithmetik. Springer, Berlin 1889. (reprint: Chelsea, New York, 1981.)

[Ka91] M. KAIB: The Gauß Lattice Basis Reduction Algorithm Succeeds With Any Norm. Proceedings of the FCT'91, Springer Lecture Notes on Computer Science, vol. 529 (1991), pp. 275-286.

[KS93] M. KAIB, C.P. SCHNORR: The Generalized Gauss Reduction Algorithm. Technical Report. Universität Frankfurt 1993.

[Le83] H.W. Lenstra, Jr: Integer programming with a fixed number of variables. Mathematics of Operations Research, Vol. 8, No. 4, (1983), pp. 538-548.

[LLL82] A.K. Lenstra, H.W. Lenstra, Jr. and L. Lovász: Factoring polynomials with rational coefficients. Math. Annalen 261 (1982), pp. 515-534.

[LS92] L. Lovász, H. Scarf: The Generalized Basis Reduction Algorithm. Mathematics of Operations Research, vol. 17, No. 3 (1992), pp. 754-764.

[R78] G.J. Rieger: Über die mittlere Schrittzahl bei Divisionsalgorithmen. Math. Nachr. 82 (1978), pp. 157-180.

[Sch93] C.P. Schnorr: Factoring Integers and Computing Discrete Logarithms via Diophantine Approximation. In: Advances in Computational Complexity, Ed. Jim–Yi Cai, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, AMS (1993), pp. 171-181.

[Sch91] A. Schönhage: Fast Reduction and Composition of Binary Quadratic Forms. In: Proc. ISSAC 1991, Ed. S.M. Watt, ACM 1991, pp. 128-133.

[Va91] B. Vallée: Gauss' Algorithm Revisited. Journal of Algorithms 12 (1991), pp. 556-572.

[VF90] B. Vallée, Ph. Flajolet: The Lattice Reduction Algorithm of Gauss: An Average Case Analysis. Proc. 31st IEEE Symposium on Foundations of Computer Science, 1990, pp. 830-842.