A More Efficient Algorithm for Lattice Basis Reduction

C. P. SCHNORR*

Fachbereich Mathematik / Informatik, Universität Frankfurt, Frankfurt, West Germany

Received January 6, 1986

The famous lattice basis reduction algorithm of Lovász transforms a given integer lattice basis \( b_1, \ldots, b_n \in \mathbb{Z}^n \) into a reduced basis, and does this by \( O(n^4 \log B) \) arithmetic operations on \( O(n \log B) \)-bit integers. Here \( B \) bounds the euclidean length of the input vectors, i.e., \( \|b_1\|^2, \ldots, \|b_n\|^2 \leq B \). The new algorithm simulates the Lovász algorithm through approximate arithmetic. It uses at most \( O(n^4 \log B) \) arithmetic operations on \( O(n + \log B) \)-bit integers. For most practical cases reduction can be done without very large integer arithmetic but with floating point arithmetic instead. © 1988 Academic Press, Inc.

1. INTRODUCTION

The lattice basis reduction algorithm proposed in Lenstra et al. [8] is a fundamental technique for solving various types of integer programming problems, such as diophantine approximation [7, 8, [14]], breaking knapsack cryptosystems [1, 3, 10–12], finding integer relations among real numbers [5]. In the latter paper it is shown that the Lovász reduction algorithm leads to a multidimensional Euclidean algorithm and in fact is very similar to a variant of the vector Euclidean algorithm of Ferguson and Forcade [2].

The implementation of the Lovász algorithm uses arithmetic operations on large integers; the integers occurring in the algorithm have \( O(n \log B) \) bits, where \( B \) bounds the euclidean length of the input basis vectors \( b_1, \ldots, b_n \in \mathbb{Z}^n \) and \( n \) is the dimension of the lattice. To avoid the overhead of large integer arithmetic it has been proposed to do part of the computation with approximate real numbers in floating point arithmetic.

*This research has been done at the University of Chicago, Department of Computer Science.
The main problem unsolved so far, is to preserve sufficient accuracy throughout the computation. We solve this problem by a method of self-correction that is used to approximate the inverse of a given matrix. The new reduction algorithm operates on integers with $O(n + \log B)$ bits and uses $O(n^4 \log B)$ arithmetic operations on such integers.

A completely different speed up of the Lovász reduction algorithm has been proposed by Schönhage [13]. By combining our algorithm with Schönhage's semi-reduction one obtains a reduction algorithm that uses at most $O(n^{3.5} \log B)$ arithmetic operations on $O(n + \log B)$-bit integers, see Theorem 9.

2. THE LOVÁSZ LATTICE BASIS REDUCTION ALGORITHM

Let $b_1, \ldots, b_n \in \mathbb{R}^n$ be linearly independent vectors. The set of all points

$$u_1 b_1 + u_2 b_2 + \cdots + u_n b_n$$

with integral $u_1, \ldots, u_n$ is called a lattice with basis $b_1, \ldots, b_n$. Let $\langle , \rangle$ be the euclidean, inner product on $\mathbb{R}^n$, and $\|y\| = \langle y, y \rangle^{1/2}$ is the euclidean length of vector $y$. The transpose of matrix $A$ is denoted $A^T$. Vectors $y \in \mathbb{R}^n$ are considered to be $n \times 1$-matrices (column vectors), then $y^T$ is a row vector. Gram-Schmidt orthogonalization associates with an ordered lattice basis $b_1, \ldots, b_n$ the orthogonal basis $b_1^*, \ldots, b_n^*$ and the elimination factors $\mu_{i,j} = \langle b_i, b_j^* \rangle \|b_j^*\|^{-2}$ such that $\mu_{i,i} = 1$ and $\mu_{i,j} = 0$ for $i < j$. The equation $b_i = \sum_{i \leq j} \mu_{i,j} b_j^*$ gives rise to the $QR$ factorization

$$[b_1, \ldots, b_k] = [b_1^*, \ldots, b_k^*] M_k^T$$

with $M_k := [\mu_{i,j}]_{1 \leq i, j \leq k}$. We will also work with the following $k \times k$ matrices for $k = 1, \ldots, n$:

$$N_k = [v_{i,j}]_{1 \leq i, j \leq k} := M_k^{-1}, \quad P_k := [\langle b_i, b_j \rangle]_{1 \leq i, j \leq k},$$

$$Q_k = [q_{i,j}]_{1 \leq i, j \leq k} := P_k^{-1}.$$ 

Since $M_k$ is lower triangular, the matrix $M_k$ is lower triangular too, and $N_{k-1}$ is a submatrix of $N_k$. On the other hand, $Q_{k-1}$ is not a submatrix of $Q_k$, i.e., the coefficients $q_{i,j}$ of $Q_k$ depend on $k$. The $QR$ factorization yields the matrix decomposition

$$P_k = M_k \begin{bmatrix} \|b_1^*\|^2 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \|b_k^*\|^2 \end{bmatrix} M_k^T \quad Q_k = N_k^T \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ \|b_1^*\|^{-2} & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & \|b_k^*\|^{-2} \end{bmatrix} N_k.$$ 

The new algorithm simulates the Lovász reduction algorithm through
approximate arithmetic. We give an outline of the Lovász algorithm omitting the details on how to compute and how to update the numbers \( \mu_{k,j}, \|b_{k-1}^*\|^2 \). Let \([r]\) be the nearest integer to the real number \( r \).

**OUTLINE OF THE LOVÁSZ ALGORITHM.**

1. (Initiation) \( k := 2 \) (\( k \) is the *stage* of the algorithm).
2. \( b_k := b_k - [\mu_{k,k-1}]b_{k-1} \).
   If \( \|b_k^* + \mu_{k,k-1}b_{k-1}^*\|^2 \frac{4}{3} \geq \|b_{k-1}^*\|^2 \) then go to 4.
3. Exchange \( b_k \) and \( b_{k-1} \), if \( k > 2 \) then \( k := k - 1 \), go to 2.
4. For \( j = k - 2, \ldots, 1 \) do \( b_k := b_k - [\mu_{k,j}]b_j \).
5. If \( k < n \) then \( (k := k + 1 \) and go to 2) otherwise terminate.

On termination the output basis \( b_1, \ldots, b_n \) satisfies the following reduction properties:

(i) \( |\mu_{i,j}| \leq 0.5 \) for \( 1 \leq i < j \leq n \).

(ii) \( \|b_i^* + \mu_{i,i-1}b_{i-1}^*\|^2 \frac{4}{3} \geq \|b_i^*\|^2 \) for \( i = 2, \ldots, n \).

**THEOREM** (A. K. Lenstra, H. W. Lenstra, Jr. and L. Lovász [8]). For an integer input basis \( b_1, \ldots, b_n \in \mathbb{Z}^n \) with length bound \( \|b_1\|^2, \ldots, \|b_n\|^2 \leq B \) the Lovász algorithm takes at most \( O(n^4 \log B) \) arithmetic operations on integers with \( O(n \log B) \) bits.

The integers occurring in the computation are the numerators and denominators of the rational numbers \( \mu_{i,j} \) and \( \|b_i^*\|^2 \). The Gramian determinants \( d_i := \det P_i = \prod_{j=1}^i |b_j|^2 \) are integers. The number \( \|b_i^*\|^2 = d_i/d_{i-1} \) has denominator \( d_{i-1} \); we have \( \mu_{i,j}, d_j \in \mathbb{Z} \) and \( b_j^*d_{j-1} \in \mathbb{Z}^n \) as can be seen by induction on \( j \). Initially the numerators and denominators of the numbers \( \mu_{i,j}, \|b_i^*\|^2 \) are bounded by \( B^n \). A detailed discussion in Lenstra et al. [8] shows that, throughout the computation, these integers are bounded by \( n2^nB^n \), thus they have at most \( O(n \log B) \) bits.

In order to avoid large integer arithmetic it has been proposed to use floating point arithmetic for the numbers \( \mu_{k,j}, \|b_k^*\|^2 \) instead. For the purpose of the test in Step 2 of the Lovász algorithm it would be sufficient to have the numbers \( \mu_{k,k-1}, \|b_k^*\|^2, \|b_{k-1}^*\|^2 \) with a numerical error that is somewhat smaller than \( \|b_{k-1}^*\|^2 \). An even more coarse approximation for the numbers \( \mu_{k,j} \) would be sufficient for the other operations of Steps 2 and 4. This approach looks promising. The numbers \( \|b_{k-1}^*\|^2, \|b_k^*\|^2 \) are never large since the basis \( b_1, \ldots, b_{k-1} \) is already reduced. The numbers \( \mu_{k,j} \) are small too; we show in Lemma 1 that \( |\mu_{k,j}| \leq \sqrt{nB2^n} \) holds throughout the computation.
However, it is nontrivial to sustain the required accuracy throughout the computation. Even the initial approximation of all the \( \mu_{i,j} \) seems to require \( O(n^3) \) arithmetic operations with \( O(n \log B) \) precision bits. Note that the step

\[
\mu_k, j := \langle b_k, b_j^* \rangle \|b_j^*\|^2
\]

which computes \( \mu_{k,j} \) from \( b_j^* \) may lose \( \log B \) precision bits if an approximation for \( b_j^* \) is used and \( \|b_k\|^2 \) is as large as \( B \).

We overcome this problem by a novel method of self-correction that improves the accuracy of an initial approximation by one step of an iteration that converges quadratically to the correct values. To this end we use Schulz's method for approximating the inverse of a matrix.

3. The New Basis Reduction Algorithm

The new algorithm simulates the Lovász algorithm and uses numbers \( \nu_{i,j} \) that approximate the entries \( v_{i,j} \) of the lower triangular matrix \( N_n = M_n^{−1} \). At stage \( k \) of the algorithm only a \( (k-1) \times (k-1) \) matrix \( N_{k−1} \) approximating the submatrix \( N_{k−1} \) of \( N_n \) is available. Associated with \( N_{k−1} \) are the following matrix \( Q_{k−1} \) and the vectors \( \bar{b}_1^*, \ldots, \bar{b}_{k−1}^* \) that are stored during stage \( k \):

\[
\begin{bmatrix}
\bar{b}_1^*, \ldots, \bar{b}_{k−1}^*
\end{bmatrix} :=
\begin{bmatrix}
b_1, \ldots, b_{k−1}
\end{bmatrix} N_{k−1}^T,
\]

\[
Q_{k−1} := N_{k−1}^T
\begin{bmatrix}
\|\bar{b}_1^*\|^2 & 0 \\
0 & \|\bar{b}_{k−1}^*\|^2
\end{bmatrix} N_{k−1}.
\] (1)

To demonstrate the correctness of the algorithm we will show that the following conditions always hold when entering stage \( k \).

**Conditions That Hold upon Entry of Stage \( k \)**

\[
|\mu_{i,j}| < 0.55 \quad \text{for } 1 \leq j < i < k.
\] (2)

\[
\|b_i^* + \mu_{i,i-1} b_{i-1}^*\|^2 \leq 1.05 \|b_{i-1}^*\|^2 \quad \text{for } 1 < i < k.
\] (3)

the matrices \( P_{k−1}, N_{k−1}, Q_{k−1} \) and the vectors \( \bar{b}_1^*, \ldots, \bar{b}_{k−1}^* \) are stored according to (1).

\[
\|N_n - N_{k−1}\| \leq c(n, 1.55^{18} B^7.
\] (4)

\[
\text{with } c(n, B) = n^{16} 1.55^{18} B^7.
\] (5)
Here let the matrix norm $\|A\|$ be the maximal absolute value of the matrix elements. The number 0.55 in (2) can be replaced by any number greater than 0.5. We need a number that is larger than 0.5, since we use approximations for the numbers $\mu_{i,j}$. The number 1.05 in (3) corresponds to $\frac{4}{3}$ in the Lovász algorithm. We use the smaller number 1.05 in order to perform a stronger reduction, and this will also reduce the accuracy required for the numbers $\bar{\nu}_{i,j}$. The matrix $\overline{N}_{k-1}$ will be a lower triangular submatrix with 1's in the diagonal. Since $c(n,B)$ does not depend on $k$ we can extend $\overline{N}_{k-1}$ to $\overline{N}_k$ by computing the numbers $\bar{\nu}_{k,1}, \ldots, \bar{\nu}_{k,k-1}$.

Outline of the New Algorithm

Steps 2, 3, and 4 of the Lovász algorithm can easily be simulated using the matrix $\overline{N}_{k-1}$ and the vectors $\bar{b}_{1}^*, \ldots, \bar{b}_{k-1}^*$. For this we approximate $\mu_{k,j} = \langle b_k, b_j^* \rangle \|b_j^*\|^2$ by $\langle b_k, b_j^* \rangle \|b_j^*\|^2$, and $\|b_{k-1}^*\|^2$ by $\|b_{k-1}^*\|^2$. We approximate $\|b_k^* + \mu_{k,k-1}b_{k-1}^*\|^2 = \|b_k\|^2 - \sum_{j<k} \langle b_k, b_j^* \rangle^2 \|b_j^*\|^2 - \sum_{j<k} \langle b_k, b_{k-1}^* \rangle^2 \|b_{k-1}^*\|^2$. We show in Lemmata 1 and 2 that these approximations are sufficiently accurate.

When $k$ increases we extend $\overline{N}_{k-1}$ to $\overline{N}_k$, as is explained subsequently, and then we compute $\overline{b}_k$ and $\overline{Q}_k$ from $\overline{Q}_{k-1}$ and $\overline{N}_k$. The matrix $\overline{Q}_{k-1}$ is only needed to extend $\overline{N}_{k-1}$ to $\overline{N}_k$. The numbers $\nu_{k,1}, \ldots, \nu_{k,k-1}$ are defined by the equations

$$0 = \langle \overline{b}_k^*, \overline{b}_i \rangle = \langle \overline{b}_k, \overline{b}_i \rangle + \sum_{j<k} \nu_{k,j} \langle \overline{b}_j, \overline{b}_i \rangle$$

for $i = 1, \ldots, k-1$.

We obtain a first approximation for $\nu_{k,1}, \ldots, \nu_{k,k-1}$ as

$$\begin{bmatrix} \nu_{k,1} \\ \vdots \\ \nu_{k,k-1} \end{bmatrix} = -\overline{Q}_{k-1} \begin{bmatrix} \langle b_1, b_k \rangle \\ \vdots \\ \langle b_{k-1}, b_k \rangle \end{bmatrix}.$$

We obtain a first approximation for $\nu_{k,1}, \ldots, \nu_{k,k-1}$ as

$$\begin{bmatrix} \bar{\nu}_{k,1} \\ \vdots \\ \bar{\nu}_{k,k-1} \end{bmatrix} := -\overline{Q}_{k-1} \begin{bmatrix} \langle b_1, b_k \rangle \\ \vdots \\ \langle b_{k-1}, b_k \rangle \end{bmatrix}, \quad \bar{\nu}_{j,k} := \begin{cases} 1, & j = k, \\ 0, & j < k. \end{cases}$$

Using $\bar{b}_k^* := \sum_{j \leq k} \bar{\nu}_{k,j} b_j$, we obtain a first approximation $\overline{Q}_k$ for $Q_k$ by evaluating (1):

$$\overline{q}_{k,i} := 0$$

for $i = 1, \ldots, k$,

$$\overline{q}_{i,j} := q_{i,j} + \bar{\nu}_{k,j} \bar{b}_k^* \|\bar{b}_k^*\|^2$$

for $1 \leq i, j \leq k$. 

For the final approximation of \( v_{k,1}, \ldots, v_{k,k-1} \) and \( Q_k \) we use the iteration 
\[ \overline{Q}_k := 2\overline{Q}_k - \overline{Q}_k P_k \overline{Q}_k \]
in which \( \overline{Q}_k \) converges quadratically to \( P_k^{-1} = \overline{Q}_k \). This is known as Schulz's method for computing the inverse matrix. We merely compute the last row of
\[ 2\overline{Q}_k' - \overline{Q}_k' P_k \overline{Q}_k' \]
which can be done in \( O(k^2) \) arithmetic operations. Since the last row of \( Q_k \) is \( [v_{k,1}, \ldots, v_{k,k}]\|b_k^*\|^{-2} \) this new last row of \( \overline{Q}_k \) gives new and better approximate values \( \tilde{v}_{k,1}, \ldots, \tilde{v}_{k,k-1} \). Using the new matrix \( \overline{N}_k \) we again compute \( b_k^* \) and \( \overline{Q}_k \) according to (1).

When \( k \) decreases we restrict \( P_{k-1} \) to \( P_{k-2} \) and \( \overline{N}_{k-1} \) to \( \overline{N}_{k-2} \); recomputing \( \overline{Q}_{k-2} \) from \( \overline{Q}_{k-1} \) and \( \tilde{v}_{k-1,1}, \ldots, \tilde{v}_{k-1,k-1} \|b_{k-1}^*\| \) is easy.

**The New Basis Reduction Algorithm.**

1. (Initiation) \( k := 2, \tilde{v}_{1,1} := 1, \overline{b}_1^* := b_1, \)
\[ \tilde{q}_{i,j} := \begin{cases} 
\langle b_1, b_1 \rangle^{-1}, & \text{if } (i, j) = (1, 1), \\
0, & \text{for } 1 \leq i, j \leq n \text{ with } (i, j) \neq (1, 1).
\end{cases} \]

2. \( r := [\langle b_k, b_{k-1}^* \rangle \|b_{k-1}^*\|^{-2}], b_k := b_k - rb_{k-1}. \)

3. If \( (\|b_k\|^2 - \Sigma_{j<k} (b_k \overline{b}_j^*)^2 \|\overline{b}_j^*\|^2) 1.025 \geq \|b_{k-1}^*\|^2 \) then go to 4. 
Exchange \( b_k \) and \( b_{k-1} \), 
if \( k = 2 \) then \( \overline{b}_1^* := b_1, \tilde{q}_{1,1} := \langle b_1, b_1 \rangle^{-1}, \) go to 2), \( k := k - 1. \)
For \( i = 1, \ldots, k \) do \( \tilde{q}_{i,k} := 0. \) 
For \( i = 1, \ldots, k \) for \( j = 1, \ldots, k \) do \( \tilde{q}_{i,j} := \tilde{q}_{i,j} - \tilde{v}_{i,j} \overline{b}_k^* \|b_k^*\|^2. \)
Go to 2.

4. For \( j = k - 2 \) down to 1 do \( (r := [\langle b_k, \overline{b}_j^* \rangle \|\overline{b}_j^*\|^{-2}], b_k := b_k - rb_j). \)

5. (First approximation of \( v_{k,1}, \ldots, v_{k,k-1}, b_k^* \), \( Q_k \)) 
\[ \begin{bmatrix} \tilde{v}_{k,1} \\
\vdots \\
\tilde{v}_{k,k-1} 
\end{bmatrix} := -\overline{Q}_{k-1} \begin{bmatrix} \langle b_k, b_1 \rangle \\
\vdots \\
\langle b_k, b_{k-1} \rangle 
\end{bmatrix}, \]
\[ \tilde{v}_{k,k} := 1, \overline{b}_k^* := b_k + \Sigma_{j<k} \tilde{v}_{k,j} b_j. \]
For \( i = 1, \ldots, k \) for \( j = 1, \ldots, k \) do \( \tilde{q}_{i,j} := \tilde{q}_{i,j} + \tilde{v}_{i,j} \overline{b}_k^* \|b_k^*\|^{-2}. \)

6. (Compute the \( k \)th row of \( 2\overline{Q}_k' - \overline{Q}_k' P_k \overline{Q}_k' \))
Compute \( \langle b_k, b_1 \rangle, \ldots, \langle b_k, b_{k-1} \rangle. \)
For \( r = 1, \ldots, k \) do \( \tilde{r}_r := \Sigma_{i=1}^k \tilde{q}_{i,k} \langle b_i, b_r \rangle. \)
For \( j = 1, \ldots, k \) do \( \tilde{q}_{i,j} := 2\tilde{q}_{i,j} - \Sigma_{i=1}^k \tilde{q}_{i,j} \tilde{r}_r. \)
7. (Final approximation of \( \nu_{k,1}, \ldots, \nu_{k,k-1}, b^*_k, Q_k \))
   For \( j = 1, \ldots, k - 1 \) do \( \bar{\nu}_{k,j} := \bar{q}_{k,j}/\bar{q}_{k,k} \).
   Reduce the bit length of \( \bar{\nu}_{k,1}, \ldots, \bar{\nu}_{k,k-1} \) to at most \( \log_2 c(n, B) \) bits to the
   right of the point.
   \( \bar{b}^*_k := b_k + \sum_{j < k} \bar{\nu}_{k,j} b_j. \)
   For \( i = 1, \ldots, k \) for \( j = 1, \ldots, k \) do \( \bar{q}_{i,j} := \bar{q}_{i,j} + \bar{\nu}_{k,j} \bar{\nu}_{k,i} \|ar{b}^*_k\|^{-2}. \)
   8. If \( k < n \) then \( (k := k + 1, \text{go to 2}) \) otherwise terminate.

Remarks (i) It is clear that the number of arithmetic operations per
iteration, i.e., before the next return to step 2, is at most \( O(k^2) \). Thus the
number of arithmetic operations per iteration is proportional to that of the
Lovász algorithm.
   (ii) The correction of \( \bar{\nu}_{k,1}, \ldots, \bar{\nu}_{k,k-1} \) by steps 5–7 can be repeated.
   This will reduce the number \( c(n, B) \) of necessary precision bits.

3. The Correctness of the Algorithm

We first derive from the inequalities (2), (3), (5) upper bounds on various
numbers that are relevant for the computation of stage \( k \). We then prove
that the inequalities (2), (3), (5) hold when entering the next stage of the
algorithm, and this finishes the proof of correctness.

The inequality (2) implies via the standard computation of the inverse
matrix the inequality

\[
|\nu_{i,j}| \leq 1.55^{i-j} \quad \text{for } 1 \leq j < i < k. \tag{6}
\]

Therefore the entries \( \bar{\nu}_{i,j} \) of the matrix \( \bar{N}_{k-1} \) in (5) can be taken as rationals in
binary fixed point representation with at most \( n \log_2 1.55 + \log_2 c(n, B) \)
bits.

During the computation of the Lovász algorithm \( \max \|b^*_i\| \) does not
increase, since for every exchange \( b_{k-1} \leftrightarrow b_k \) the new vector \( b^*_{k-1} \) is shorter
than the old one. We will see below that this also holds for our algorithm.
As a consequence \( \|b^*_i\|^2 \leq B \) holds throughout the computation and we have

\[
\|b_i\|^2 = \sum_{j \leq i} \mu_{i,j}^2 \|b^*_j\|^2 \leq (2)^i B \quad \text{for } i < k. \tag{7}
\]

From \( \|b^*_i + \mu_{i,i-1} b^*_{i-1}\|^2 \leq 1.05 \geq (3) \|b^*_{i-1}\|^2 \) and (2) we conclude that \( \|b^*_i\|^2 \geq (1.05^{-1} - 0.55^2) \|b^*_{i-1}\|^2 \) for \( i < k \). Since \((1.05^{-1} - 0.55^2)^{-1} < 1.55 \) this
yields \( \|b_{i-1}\|^2 < 1.55\|b_i\|^2 \) and thus proves
\[
\|b_i^*\|^{-2} < 1.55^{i-1}\|b_1\|^{-2} \leq 1.55^{i-1} \quad \text{for } i < k,
\] (8)
the latter inequality follows from \( \|b_i^*\| = \|b_i\| \geq 1 \). Since \( q_{i,j} = \sum_{r=1}^{k-1} \nu_{r,i} \nu_{r,j} \|b_r^*\|^{i-j} \leq \sum_{r=1}^{k-1} 1.55^{k-3} \leq 1.55^{3k-3}(1 - 1.55^{-3})^{-1}. \)

This proves
\[
|q_{i,j}| < 1.55^{3k-2} \quad \text{for } i, j < k.
\] (9)

Lemma 1, following, improves the upper bounds on \( |\mu_{k,i}|, \|b_k\| \) in [8, formulae (1.31), (1.34)]. The tighter upper bound has also been observed by E. Kaltofen [6].

**Lemma 1.** The inequality (10) holds throughout the computation:
\[
\|b_i\|^2 \leq nB \quad \text{for } i \neq k.
\] (10)

The inequalities (11), (12) hold upon entry of stage \( k \):
\[
|\mu_{i,j}|^2 < nB \ 1.55^{j-1} \quad \text{for } j < k < i,
\] (11)
\[
|\mu_{k,j}|^2 < nB \ 1.6^{2n-k} \quad \text{for } j < k.
\] (12)

The inequalities (13), (14) hold throughout stage \( k \):
\[
|\mu_{k,j}|^2 < nB \ 1.6^{2n} \quad \text{for } j < k,
\] (13)
\[
\|b_k\| < nB \ 1.6^n.
\] (14)

**Proof.** The inequalities are clearly true for the initial stage. We prove each inequality by induction, assuming that it already holds for the previous stage \( k' \).

**Proof of (10).** For \( i < k \) the inequality (10) follows from (2) and \( \max_j \|b_j^*\|^2 \leq B \). For \( i > k \) we see that the inequality (10) follows from the inequality (10) of the previous stage \( k' \). This is obvious for \( i > k' \). If \( i = k' \) then \( k' = k + 1 \), \( b_i \) is the vector \( b_{k'-1} \) of the previous stage \( k' \), and the inequality follows from the inequality \( \|b_{k'-1}\|^2 \leq nB \) of stage \( k' \).

**Proof of (11).** We have for \( j < k < i \):
\[
|\mu_{i,j}|^2 = |\langle b_i, b_j^* \rangle|^2 \|b_j^*\|^{-4} \leq \|b_i\|^2 \|b_j^*\|^{-2} \leq (10)nB \|b_j^*\|^{-2} \leq (8)nB \ 1.55^{j-1}.
\]
Proof of (12). If \( k' = k - 1 \) then \( b_k \) is the vector \( b_k \) of stage \( k' \), so here (12) follows from the inequality (11) of stage \( k' \).

If \( k' = k + 1 \) then \( b_k \) is the vector \( b_k \) of the previous stage \( k' \), and we follow what happened to \( \mu_{k',j} \) during stage \( k' \). It is in step 2 replaced by \( \mu_{k',j} - r \mu_{k',j-1} \) with \( |r - \mu_{k',k'-1}| < ^{(15)}0.55 \) and \( |\mu_{k',j} - \mu_{k',j-1}| < ^{(2)}0.55 \). Thus the inequality (12) of the previous stage \( k' \) implies

\[
|\mu_{k,j}| = |\mu_{k',j} - r \mu_{k',j-1}| \leq |\mu_{k',j}| + |r \mu_{k',j-1}|
\]

\[
\leq ^{(2),(12)}1.55nB \cdot 1.6^{2n-k'} + 0.55^2 \leq nB \cdot 1.6^{2n-k}
\]

(provided that \( n \geq 3 \))

which proves the claim.

Proof of (13). We know from (12) that upon entry of stage \( k \), \( |\mu_{k,j}|^2 < nB \cdot 1.6^{2n-k} \) for \( j < k \). We follow what happens to \( \mu_{k,j} \) during stage \( k \). In Step 2, \( \mu_{k,j} \) is replaced by \( \mu_{k,j} - r \mu_{k,j-1} \) with \( |r - \mu_{k,k-1}| < ^{(15)}0.55 \). We have seen above that this increases \( \max_{j < k} |\mu_{k,j}|^2 \) by at most a factor 1.6. In Step 4 the same thing happens for \( k - 2 \) values \( r_v \) with \( |r_v - \mu_{k,v}| < 0.55 \) for \( v = k - 2, \ldots, 1 \). Therefore \( \max_{j < k} |\mu_{k,j}|^2 \) increases at most by a factor 1.6\(^{k-1} \) during the computation of stage \( k \), and thus (13) follows from (12).

Proof of (14). \( \|b_k\|^2 = \|b_k^*\|^2 + \sum_{j < k} |\mu_{k,j}^2| |b_j^*|^2 \leq knB^2 \cdot 1.6^{2n} \) by (13), (2), and \( \max_{j} |b_j^*|^2 \leq B \).

Q.E.D.

Remark. The proof for the inequalities (13), (14), and (15) below is interrelated so that the inequalities (13), (14) of stage \( k \) follow from inequality (15) of the previous stage \( k' \), and (15) of stage \( k \) will follow from (13), (14) of stage \( k \).

**Lemma 2.** The inequality (15) holds throughout stage \( k \):

\[
|\mu_{k,j} - \langle b_k, \bar{b}_j^* \rangle \|\bar{b}_j^*\|^{-2}| < 0.05 \quad \text{for} \quad j < k.
\]

Proof. Since \( \|b_i\|^2 \leq nB \) for \( i < k \) and \( b_j^* = b_j + \sum_{i < j} v_{i,j} b_i \) the inequality (5) implies

\[
\|b_j^* - \bar{b}_j^*\| < n^{1.5}B^{0.5}c(n, B)^{-1} \quad \text{for} \quad j < k.
\]

Because of (16) the difference vector \( b_j^* - \bar{b}_j^* \) is so short that (8) and the definition of \( c(n, B) \) imply \( \|b_j^*\|/\|\bar{b}_j^*\| \leq 1.4 \). The following inequalities
hold throughout stage $k$ for $j < k$:

$$
|\mu_{k, j} - \langle b_k, \bar{b}_j^* \rangle \|b_j^*\|^{-2}| = |\langle b_k, b_j^* - \bar{b}_j^* \rangle \|b_j^*\|^{-2} |
\leq \|b_k\| \|b_j^* - \bar{b}_j^*\| \|b_j^*\|^{-2}
\leq (14), (16) n B 1.6^n n^{1.5} B^{0.5} c(n, B)^{-1} \|b_j^*\|^{-2}
\leq (8) n^{2.5} B^{1.5} 1.6^{2n} c(n, B)^{-1} < \frac{1}{40} .
$$

$$
\langle b_k, \bar{b}_j^* \rangle (\|b_j^*\|^{-2} - \|\bar{b}_j^*\|^{-2}) \leq \|b_k\| \|b_j^* - \bar{b}_j^*\|
\times (\|b_j^*\| + \|\bar{b}_j^*\|) \|b_j^*\|^{-2} \|\bar{b}_j^*\|^{-1}
\leq (14), (16), (8) 2.4
n B 1.6^n n^{1.5} B^{0.5} c(n, B)^{-1} 1.55^{-1}
< n^{2.5} B^{1.5} 1.6^{2n} c(n, B)^{-1} < \frac{1}{40} .
$$

The two above inequalities imply (15). Q.E.D.

**Remark.** Step 2 of the algorithm, by Lemma 2, achieves $|\mu_{k, k-1}| < 0.55$ and step 4 achieves $|\mu_{k, j}| < 0.55$ for $j = k - 1, \ldots, 1$. Therefore the inequality (2) is preserved throughout the algorithm.

**Lemma 3.** The inequalities (17), (18) hold during stage $k$:

$$
\|b_k\|^2 - \sum_{j < k-1} \langle b_k, \bar{b}_j^* \rangle^2 \|b_j^*\|^{-2} - \|b_k^* + \mu_{k, k-1} b_{k-1}^*\|^2 \leq 0.01 \|b_{k-1}\|^2
\tag{17}
$$

$$
\|\bar{b}_{k-1}\|^2 - \|b_{k-1}\|^2 \leq 0.01 \|b_{k-1}\|^2. \tag{18}
$$

**Proof of (17).** We have $b_k^* = b_k - \sum_{j < k} \mu_{j, k} j b_j^*$ and

$$
b_k^* + \mu_{k, k-1} b_{k-1}^* = b_k - \sum_{j < k-1} \langle b_k, b_j^* \rangle \|b_j^*\|^{-2} b_{j^*}.
$$

This implies $\|b_k^* + \mu_{k, k-1} b_{k-1}^*\|^2 = \|b_k\|^2 - \sum_{j < k-1} \langle b_k, b_j^* \rangle^2 \|b_j^*\|^{-2}$. We approximate this latter number by $\|b_k\|^2 - \sum_{j < k-1} \langle b_k, \bar{b}_j^* \rangle^2 \|b_j^*\|^{-2}$. The numerical error hereby is at most

$$
\sum_{j < k-1} (\|\langle b_k, b_j^* + \bar{b}_j^* \rangle \langle b_k, b_j^* - \bar{b}_j^* \rangle \|b_j^*\|^{-2}
+ \langle b_k, \bar{b}_j^* \rangle^2 \|b_j^*\|^{-2} - \|\bar{b}_j^*\|^{-2})
\leq (14) (k - 2) n B 2^{1.6^{2n} 2} \max \|b_j - \bar{b}_j^*\| (\|b_j^*\| + \|\bar{b}_j^*\|) \|b_j^*\|^{-2},
$$
We have used that
\[
|\|b_j^*\|^2 - \|\bar{b}_j^*\|^2| = (\|b_j^*\| + \|\bar{b}_j^*\|)(\|b_j^*\| - \|\bar{b}_j^*\|)(\|b_j^*\|^2 + \|\bar{b}_j^*\|^2)
\leq (16) (k - 2)n^2 B^2 1.62^n 2n^{1.5} B^{0.5} c(n, B)^{-1} 21.6^n / 2
\leq (5) 0.01 \cdot 1.55^{1 - k} < (8) 0.01 \|b_{k - 1}^*\|^2
\]
which proves (17).

The inequality (18) is a direct consequence of the inequalities (5), (14), (16).

**Remark.** A straightforward calculation using the inequalities (17), (18) shows that the test in Step 3 has the following effect.

If \( \|b_k^* + \mu_{k, k - 1} b_{k - 1}^*\|^2 \cdot 1.05 \leq \|b_{k - 1}^*\|^2 \) then \( b_k, b_{k - 1} \) are exchanged.

If \( \|b_k^* + \mu_{k, k - 1} b_{k - 1}^*\|^2 \cdot 1.01 \geq \|b_{k - 1}^*\|^2 \) then \( b_k, b_{k - 1} \) are not exchanged.

As a consequence, for each exchange \( b_{k - 1} \leftrightarrow b_k \) the new vector \( b_{k - 1}^* \) is shorter than the old one. Therefore \( \max_i \|b_i^*\| \) does not increase throughout the computation. We also see that our exchange conditions aim to establish the inequalities (3). We do not exchange \( b_{k - 1}, b_k \) if the inequality (3) already holds for \( i = k \).

To finish the correctness proof it remains to show that \( \|N_k - \bar{N}_k\| \leq c(n, B)^{-1} \) holds before \( k \) increases. For this we analyze the error propagation during Steps 5–7. It will be important to note that, at the beginning of Step 5, the inequalities (6)–(10) even hold for \( i = k \). This is because the inequalities (2), (3) already hold with \( k \) replaced by \( k + 1 \).

**Lemma 4.** The inequality (19) holds throughout Step 5:
\[
\|Q_{k - 1} - \bar{Q}_{k - 1}\| \leq n^{1.5} 1.55^{3.5(k - 1)} \sqrt{B} c(n, B)^{-1}.
\]

The inequalities (20), (21) hold upon termination of step 5:
\[
\|\nu_{k, j} - \bar{\nu}_{k, j}\| \leq n^{3.5} 1.55^{3.5(k - 1)} B^{1.5} c(n, B)^{-1} \quad \text{for } 1 \leq j \leq k - 1; \quad (20)
\]
\[
\|b_k^* - \bar{b}_k^*\| \leq n^{5.5} 1.55^{3.5(k - 1)} B^2 c(n, B)^{-1}. \quad (21)
\]

**Proof of (19).** By definition of the matrix \( Q_{k - 1} = [q_{i, j}]_{1 \leq i, j \leq k - 1} \) we have
\[
q_{i, j} = \sum_{r=1}^{k-1} \nu_{r, i} \nu_{r, j} \left\|b_r^*\right\|^2 \quad \text{for } 1 \leq i, j \leq k - 1
\]
and by (1),
\[
\tilde{q}_{i,j} = \sum_{r=1}^{k-1} \tilde{v}_{r,i} \bar{v}_{r,j} \| \bar{b}_r^* \|^{-2} \quad \text{for } 1 \leq i, j \leq k - 1.
\]

This yields
\[
\begin{align*}
|q_{i,j} - \tilde{q}_{i,j}| &\leq \sum_{r=1}^{k-1} |v_{r,i} v_{r,j} \| \bar{b}_r^* \|^{-2} - \| \bar{b}_r^* \|^{-2}| \\
&\quad + \sum_{r=1}^{k-1} |v_{r,i} \| \bar{b}_r^* \|^{-2} |v_{r,j} - \bar{v}_{r,j}| \\
&\quad + \sum_{r=1}^{k-1} |\bar{v}_{r,j} \| \bar{b}_r^* \|^{-2} |v_{r,i} - \bar{v}_{r,i}|.
\end{align*}
\]

To bound the right-hand side we use the inequality
\[
\| \|b_r^*\|^{-2} - \| \bar{b}_r^*\|^{-2}\| \leq \|b_r^* - \bar{b}_r^*\| \left( \|b_r^*\| + \| \bar{b}_r^*\| \right) \|b_r^*\|^{-2} \| \bar{b}_r^*\|^{-2}
\leq (16),(8) 2 n^{1.5} B^{0.5} c(n, B)^{-1} 1.55^{1.5(r-1)}.
\]

By (8) and the definition of \(c(n, B)\) this inequality implies \(\| \bar{b}_r^*\|^{-2} \leq 1.55^r\). We conclude from (5), (6), and the definition of \(c(n, B)\) that \(|\bar{v}_{r,i}| \leq 1.55^r\). These inequalities and (5), (6) give for \(n \geq 3\):
\[
|q_{i,j} - \tilde{q}_{i,j}| \leq 2 n^{1.5} B^{0.5} \sum_{r=1}^{k-1} 1.55^{3.5(r-1)} c(n, B)^{-1} + 2 \sum_{r=1}^{k-1} 1.55^{2r} c(n, B)^{-1}
\leq n^{1.5} B^{0.5} 1.55^{3.5(k-1)} c(n, B)^{-1}.
\]

**Proof of (20).** Since
\[
\begin{bmatrix}
v_{k,1} \\
\vdots \\
v_{k,k-1}
\end{bmatrix} - \begin{bmatrix}
\bar{v}_{k,1} \\
\vdots \\
\bar{v}_{k,k-1}
\end{bmatrix}
= -(Q_{k-1} - \bar{Q}_{k-1})
\begin{bmatrix}
\langle b_k, b_1 \rangle \\
\vdots \\
\langle b_k, b_{k-1} \rangle
\end{bmatrix},
\]
the inequality (20) follows from (19) and \(|\langle b_k, b_i \rangle| \leq \|b_k\| \|b_i\| \leq nB\).

**Proof of (21).** The inequality (21) follows from (20), (10), and the equation
\[
b_k^* - \bar{b}_k^* = \sum_{j<k} (v_{k,j} - \bar{v}_{k,j}) b_j.
\]
Q.E.D.
LEMMA 5. The following inequalities hold throughout step 6:

\[ \|\bar{b}_k^*\|^2 - \|\bar{b}_k^\prime\|^2 \leq n^5 1.55^{5n}B^2 c(n, B)^{-1}; \]  
\[ \|Q_k - \bar{Q}_k\| \leq n^5 1.55^{5n}B^2 c(n, B)^{-1}; \]  
\[ \|I - \bar{Q}_k P_k\| \leq n^7 1.55^{7n}B^3 c(n, B)^{-1}. \]  
(22)  
(23)  
(24)

Proof of (22). \[ \|\bar{b}_k^*\|^2 - \|\bar{b}_k^\prime\|^2 \leq \|\bar{b}_k^* - \bar{b}_k^\prime\| (\|\bar{b}_k^*\| + \|\bar{b}_k^\prime\|) \]
\[ \|\bar{b}_k^*\|^2 \|\bar{b}_k^\prime\|^2 \leq n^5 1.55^{5n}B^2 c(n, B)^{-1}. \]  
(21)
For the latter inequality we use \[ \|\bar{b}_k^*\|^2 \leq 1.55^k \] which follows from (8), (5), and (21).

Proof of (23). We have
\[ \|Q_k - \bar{Q}_k\| = \max_{1 \leq i, j \leq k} \left| \sum_{r=1}^{k} \left( v_{r, i} v_{r, j} \|b_r^*\|^2 - \bar{v}_{r, i} \bar{v}_{r, j} \|\bar{b}_r^*\|^2 \right) \right|, \]
\[ \left| \sum_{r=1}^{k-1} v_{r, i} v_{r, j} \|b_r^*\|^2 - \bar{v}_{r, i} \bar{v}_{r, j} \|\bar{b}_r^*\|^2 \right| \leq (n^{1.55} 1.55^{3.5(k-1)} \sqrt{B} c(n, B)^{-1} \]
(19)
and we conclude from \[ |\bar{v}_{r, i}| \leq 1.55^k \] and \[ \|\bar{b}_r^*\|^2 \leq 1.55^k \] that
\[ |v_{r, i} v_{r, j} \|b_r^*\|^2 - \bar{v}_{r, i} \bar{v}_{r, j} \|\bar{b}_r^*\|^2| \]
\[ \leq (6), (8), (20), (22) n^5 1.55^{5n+2k-2} B^2 c(n, B)^{-1}. \]  
(26)

The difference \[ |v_{r, i} - \bar{v}_{r, i}| \] is, by (20), so small that its contribution to (26) is negligible. The inequality (23) now follows by adding the inequalities (25) and (26).

Proof of (24).
\[ \|I - \bar{Q}_k P_k\| = \|Q_k P_k - \bar{Q}_k P_k\| \leq \|Q_k - \bar{Q}_k\| \|P_k\| \leq n^{10} \|Q_k - \bar{Q}_k\| n^2 B, \] which derives (24) from (23).

Q.E.D.

LEMMA 6. On termination of step 6 we have for \[ i \leq k: \]
\[ |q_{k, i} - \bar{q}_{k, i}| < n^{15} 1.55^{17n}B^6 c(n, B)^{-2}. \]  
(27)

Proof. \[ [\bar{q}_{k, 1}, \ldots, \bar{q}_{k, k}] \] is the last row of the matrix
\[ \bar{Q}_k := 2 \bar{Q}_k - \bar{Q}_k P_k \bar{Q}_k = \bar{Q}_k + (I - \bar{Q}_k P_k) \bar{Q}_k. \]

Using the identity \[ I - \bar{Q}_k P_k = (I - \bar{Q}_k P_k)^2, \] we obtain
\[ \|Q_k - \bar{Q}_k\| = \|Q_k - \bar{Q}_k P_k\| \leq k \|Q_k\| \|I - \bar{Q}_k P_k\| \]
\[ \leq k^2 \|Q_k\| \|I - \bar{Q}_k P_k\|^2. \]

Now Lemma 6 follows from (24) and (9).

Q.E.D.
LEMMA 7. Step 7 yields values $\tilde{v}_{k,i}$ so that $|v_{k,i} - \tilde{v}_{k,i}| \leq c(n, B)^{-1}$ for $i = 1, \ldots, k$.

Proof. Step 7 yields $\tilde{v}_{k,i} = \tilde{q}_{k,i} \tilde{q}_{k,k}^{-1}$, whereas $v_{k,i} = q_{k,i}^{-1} q_{k,k}$ with $q_{k,k}^{-1} = \|b_k^*\|^2$. We obtain

$$|v_{k,i} - \tilde{v}_{k,i}| \leq |q_{k,i}| |q_{k,k}^{-1} - \tilde{q}_{k,k}^{-1}| + |\tilde{q}_{k,k}^{-1}| |q_{k,i} - \tilde{q}_{k,i}|$$

$$\leq |q_{k,i}| |q_{k,k}^{-1} - \tilde{q}_{k,k}^{-1}| + |q_{k,k}^{-1}| |q_{k,k} - \tilde{q}_{k,k}| + |\tilde{q}_{k,k}^{-1}| |q_{k,i} - \tilde{q}_{k,i}|$$

$$\leq |v_{k,i}| |\tilde{q}_{k,k}^{-1} - q_{k,k}^{-1}| + |q_{k,k}^{-1}| |q_{k,k} - \tilde{q}_{k,k}| + |\tilde{q}_{k,k}^{-1}| |q_{k,i} - \tilde{q}_{k,i}|.$$

We have $|q_{k,k}^{-1}| = \|b_k^*\|^2 \leq B$ and we can conclude from $q_{k,k}^{-1} = \|b_k^*\|^2 \geq 1.55^{1-k},$ (27), and the definition of $c(n, B)$ that $|\tilde{q}_{k,k}^{-1}| \leq 1.55B$. By (26) and (6) this implies

$$|v_{k,i} - \tilde{v}_{k,i}| \leq 1.55^{-1} n^{16} 1.55^{17n} B^6 c(n, B)^{-2} B$$

$$\leq n^{16} 1.55^{18n} B^7 c(n, B)^{-2} \leq (5)c(n, B)^{-1}.$$ Q.E.D.

This finishes the correctness proof for the new reduction algorithm and we can enunciate our main theorem.

THEOREM 8. The new reduction algorithm transforms an integer lattice basis $b_1, \ldots, b_n \in \mathbb{Z}^n$ with $\|b_1\|^2, \ldots, \|b_n\|^2 \leq B$ into a basis $b_1, \ldots, b_n$ with the reduction properties (i) and (ii). The algorithm uses at most $O(n^4 \log B)$ arithmetic operations on $O(n + \log B)$-bit integers:

(i) $|\mu_{i,j}| \leq 0.55$ for $1 \leq j < i \leq n$.

(ii) $\|b_i^* + \mu_{i,i-1} b_{i-1}^*\|^2 1.05 \geq \|b_{i-1}^*\|^2$ for $i = 2, \ldots, n$.

Remarks (i) To make the new algorithm operate with $O(n + \log B)$ bit numbers, the number $\Sigma_{j < k-1} \langle b_k, \bar{b}_j^* \rangle^2 \|\bar{b}_j^*\|^2$ in the test of Step 2 has to be approximated since this number does not necessarily have a denominator with $O(n + \log B)$ bits. It is sufficient to approximate this number with a numerical error $2^{-7-k}$ which by (8) is less than $0.002 \cdot \|b_{k-1}^*\|^2$.

(ii) As for the Lovász algorithm the time bound of the algorithm follows from the fact that each exchange step $b_k \leftrightarrow b_{k-1}$ reduces the number $D = \prod_{i=1}^n \|b_i^*\|^{2(n-i)}$ by at least a factor 1.01. Since initially $D \leq B^n$ and on termination $D \geq 1$, the number of exchange steps is at most $n^2 \log B / \log 1.01$. Since the number of iterations is at most $n + \text{the number of exchange steps}$ and each iteration costs no more than $O(n^2)$ arithmetical steps there are at most $O(n^4 \log B)$ arithmetical steps in total.

(iii) The improved efficiency of the new lattice basis reduction algorithm carries over to the more powerful reduction algorithms in Schnorr et al.
An entirely different speed-up of the Lovász algorithm has been proposed by A. Schönhage [14] who computes a semi-reduced basis in $O(n^3 \log B)$ arithmetic operations on $O(n \log B)$ bit integers. The Schönhage and our speed-up can be combined to a yet faster worst-case running time bound but it is open whether the two speed-ups can be added without loss. Our method strives to reduce maximal initial segments of the basis (which is necessary to preserve the accuracy of the approximation) whereas Schönhage's method requires us to reduce suitable inner blocks of the basis. A combination of the two speed-up methods to a reduction algorithm which uses $O(n^{3.5} \log B)$ arithmetic operations on $O(n + \log B)$-bit integers works as follows. Partition the basis into $\sqrt{n}$ overlapping blocks of $\sqrt{n} + 1$ basis vectors $b_{i,\sqrt{n}}, b_{i,\sqrt{n}+1}, \ldots, b_{i,\sqrt{n}+\sqrt{n}}$ for $i = 0, \ldots, \sqrt{n} - 1$. Successively reduce the block associated with the smallest $i$ such that $\|b_{i,\sqrt{n}}^*\| > 1.55^{\sqrt{n}}\|b_{(i+1),\sqrt{n}}^*\|$. 

During block reduction we merely transform the corresponding $(\sqrt{n} + 1) \times (\sqrt{n} + 1)$ submatrices of $N_n, P_n, Q_n$. On termination of block reduction the matrix of basis vectors is transformed by one matrix multiplication with a $(\sqrt{n} + 1) \times (\sqrt{n} + 1)$ unimodular matrix. An easy calculation shows that this saves a factor $\sqrt{n}$ for the number of arithmetic operations (whereas Schönhage's more general block choices saves a factor $n$). However, we can still use our method to simulate by approximate arithmetic the reduction of the first block satisfying $\|b_{i,\sqrt{n}}^*\| > 1.55^{\sqrt{n}}\|b_{(i+1),\sqrt{n}}^*\|$. The sketched algorithm yields the following theorem.

**Theorem 9.** Combining Schönhage's semi-reduction (see [14]) and our simulation of the Lovász algorithm yields a reduction algorithm that uses at most $O(n^{3.5} \log B)$ arithmetic operations on $O(n + \log B)$-bit integers. This algorithm finds a basis such that

(i) $|\mu_{i,j}| < 0.55$ for $i < j \leq n$

(ii) $\|b_i^*\|^2 < 1.55^{2(j-i)}\|b_j^*\|^2$ for all $i, j \leq n$ satisfying $j - i > \sqrt{n}$.

**Acknowledgments**

I thank J. Höpfel for reading the manuscript and for correcting several calculations, and the University of Chicago for supporting it.

**References**


2. H. R. P. Ferguson and R. W. Forcade, Generalization of the euclidean algorithm for real numbers to all dimensions higher than two, *Bull. Amer. Math. Soc. (N.S.)* 1, No. 6


