MATHEMATICS OF COMPUTATION, VOLUME 29, NUMBER 131
JULY 1975, PAGES 827–833

How to Calculate Shortest Vectors in a Lattice

By U. Dieter*

Dedicated to W. Fenchel, Copenhagen, on the occasion of his 70th birthday

Abstract. A method for calculating vectors of smallest norm in a given lattice is outlined. The norm is defined by means of a convex, compact, and symmetric subset of the given space. The main tool is the systematic use of the dual lattice. The method generalizes an algorithm presented by Coveyou and MacPherson, and improved by Knuth, for the determination of vectors of smallest Euclidean norm.

1. Formulation of the Problem. Let $\mathbf{G}$ be a lattice in the n-dimensional Euclidean space R^n, generated by n linearly independent vectors $\mathbf{e}_i$:

$$\mathbf{G} = \left\{ \mathbf{x} = \sum_{i=1}^{n} z_i \mathbf{e}_i \,\middle|\, z_i \text{ integers} \right\}. \tag{1}$$

The norm in R^n is defined by a convex, compact set $\mathbf{B}$ which has positive measure and is symmetric about the origin:

$$\|\mathbf{x}\| = \min\{\lambda \in R \mid \mathbf{x} \in \lambda \mathbf{B}\}. \tag{2}$$

Examples of these norms for $\mathbf{x} = (x_1, \ldots, x_n)$ are

(i) The Euclidean norm $\|\mathbf{x}\| = (x_1^2 + \cdots + x_n^2)^{1/2}$.

(ii) The Maximum norm $\|\mathbf{x}\| = \max\{|x_i| \,|\, i = 1, \ldots, n\}$.
Here $\mathbf{B}_\infty = \{(x_1, \ldots, x_n) \,|\, |x_i| \leqslant 1 \text{ for all } i\}$.

(iii) The norm $\|\mathbf{x}\| = |x_1| + \cdots + |x_n|$.
Here $\mathbf{B}_1 = \{(x_1, \ldots, x_n) \,|\, |x_1| + \cdots + |x_n| \leqslant 1\}$.

The problem is to find a nonzero vector of shortest length (norm) in $\mathbf{G}$. The main tool of the presented method is the use of the dual lattice,

$$\mathbf{G}^* = \left\{ \mathbf{x}^* = \sum_{k=1}^{n} z_k^* \mathbf{e}_k^* \,\middle|\, z_k^* \text{ integers} \right\}, \tag{3}$$

where the $\mathbf{e}_k^*$ are defined by $\mathbf{e}_i \mathbf{e}_k^* = \delta_{ik}$; here δ_{ik} is equal to 1 if $i = k$ and equal to 0 if $i \neq k$, and $\mathbf{e}_i \mathbf{e}_k^*$ denotes the scalar product $\Sigma_{j=1}^{n} e_{ij} e_{kj}^*$. The polar of $\mathbf{B}$, namely

$$\mathbf{B}^* = \{\mathbf{b}^* \in R^n \,|\, |\mathbf{b}\mathbf{b}^*| \leqslant 1, \forall\, \mathbf{b} \in \mathbf{B}\}, \tag{4}$$

induces a length or norm in $\mathbf{G}^*$ by

$$\|\mathbf{x}^*\|^* = \min\{\lambda^* \in R \,|\, \mathbf{x}^* \in \lambda^* \mathbf{B}^*\}. \tag{5}$$

It should be noted that the Euclidean norm corresponds to itself, whereas the Maximum norm $\|\mathbf{x}\| = \max_i |x_i|$ corresponds to $\|\mathbf{x}^*\|^* = |x_1^*| + \cdots + |x_n^*|$ and vice versa.

Received May 16, 1974; revised August 6, 1974.

AMS (MOS) subject classifications (1970). Primary 10E05, 10E20, 10E25; Secondary 65C10.

Key words and phrases. Geometry of numbers, lattice theory, minima of forms, random number generation.

* Research supported by National Research Council of Canada and ONR contract N00014-67-A-0112-0051 (NR-042-993).

For the scalar product $\mathbf{x}^*\mathbf{x} = x_1^*x_1 + \cdots + x_n^*x_n$ the following inequality holds

$$|\mathbf{x}^*\mathbf{x}| \leqslant \|\mathbf{x}^*\|^* \, \|\mathbf{x}\| \tag{6}$$

which may be proved as follows: Since $\mathbf{x} = \|\mathbf{x}\|\mathbf{b}$, $\mathbf{x}^* = \|\mathbf{x}^*\|^*\mathbf{b}^*$ for some $\mathbf{b} \in \mathbf{B}$ and $\mathbf{b}^* \in \mathbf{B}^*$ one has $\mathbf{x}^*\mathbf{x} = \|\mathbf{x}^*\|^*\|\mathbf{x}\|\mathbf{b}^*\mathbf{b}$. Since $|\mathbf{b}^*\mathbf{b}| \leqslant 1$ holds for all $\mathbf{b} \in \mathbf{B}$, $\mathbf{b}^* \in \mathbf{B}^*$, the inequality (6) is proved.

2. Presentation of the Method. If $\mathbf{x} = z_1\mathbf{e}_1 + \cdots + z_n\mathbf{e}_n$ is any element of $\mathbf{G}$, inequality (6) implies that

$$|z_i| = |\mathbf{e}_i^*(z_1\mathbf{e}_1 + \cdots + z_n\mathbf{e}_n)| = |\mathbf{e}_i^*\mathbf{x}| \leqslant \|\mathbf{e}_i^*\|^*\|\mathbf{x}\|.$$

Hence, if w is the length of a shortest nonzero vector $\mathbf{x}$ in $\mathbf{G}$, the coordinates satisfy

$$|z_i| \leqslant \|\mathbf{e}_i^*\|^* w \quad \text{for } 1 \leqslant i \leqslant n. \tag{7}$$

This inequality helps to limit the search for a shortest vector. Since

$$w = \operatorname{Min}\{\|\mathbf{x}\| \,|\, \mathbf{x} \in \mathbf{G}, \mathbf{x} \neq 0\}$$

is not known, when the algorithm is started, the minimum value of $\|\mathbf{e}_k\|$ is initially taken. Hence z_i is bounded by

$$|z_i| \leqslant c_i = \left[\|\mathbf{e}_i^*\|^* \operatorname*{Min}_k \|\mathbf{e}_k\|\right], \quad i = 1, \ldots, n; \ ([y] \text{ integral part of } y). \tag{8}$$

If the bounds c_i are reasonably small, a direct search through the

$$P = \prod_{i=1}^{n} (2c_i + 1) \tag{9}$$

possibilities may become feasible. Otherwise, attempts are made to change the bases $\mathbf{e}_i$ and $\mathbf{e}_i^*$ such that the bounds c_i are decreased. The task is to find a transformation with the following properties:

(M) The new $\|\mathbf{e}_i\|$ are smaller than the old ones.

(M*) The new $\|\mathbf{e}_i^*\|^*$ are not larger than the old ones.

Among the unimodular transformations of the $\mathbf{e}_i$ and $\mathbf{e}_i^*$, two special types are considered,

$$T_i: \left.\begin{array}{l} \mathbf{e}_i \leftarrow \mathbf{e}_i \\ \mathbf{e}_i^* \leftarrow \mathbf{e}_i^* + \sum\limits_{k \neq i} m_k\mathbf{e}_k^* \end{array}\right\} \text{ for a fixed } i \qquad \left.\begin{array}{l} \mathbf{e}_k \leftarrow \mathbf{e}_k - m_k\mathbf{e}_i \\ \mathbf{e}_k^* \leftarrow \mathbf{e}_k^* \end{array}\right\} \text{ for } k \neq i,$$

$$T_i^*: \left.\begin{array}{l} \mathbf{e}_i^* \leftarrow \mathbf{e}_i^* \\ \mathbf{e}_i \leftarrow \mathbf{e}_i + \sum\limits_{k \neq i} m_k^*\mathbf{e}_k \end{array}\right\} \text{ for a fixed } i \qquad \left.\begin{array}{l} \mathbf{e}_k^* \leftarrow \mathbf{e}_k^* - m_k^*\mathbf{e}_i^* \\ \mathbf{e}_k \leftarrow \mathbf{e}_k \end{array}\right\} \text{ for } k \neq i.$$

It is easy to see that $\mathbf{e}_i\mathbf{e}_j^* = \delta_{ij}$ also holds for the new $\mathbf{e}_i$ and $\mathbf{e}_j^*$.

In the transformation T_i, the integers m_k are chosen in such a way that the Euclidean length $\mathbf{e}_k^2$ is minimized for $k \neq i$. Consequently, m_k has to be determined by

$$(\mathbf{e}_k - (m_k - 1)\mathbf{e}_i)^2 \geqslant (\mathbf{e}_k - m_k\mathbf{e}_i)^2 \leqslant (\mathbf{e}_k - (m_k + 1)\mathbf{e}_i)^2.$$

This leads to

$$-\tfrac{1}{2}\mathbf{e}_i^2 \leqslant \mathbf{e}_i(\mathbf{e}_k - m_k\mathbf{e}_i) \leqslant \tfrac{1}{2}\mathbf{e}_i^2$$

or

$$-\tfrac{1}{2} + (\mathbf{e}_i\mathbf{e}_k)/\mathbf{e}_i^2 \leqslant m_k \leqslant \tfrac{1}{2} + (\mathbf{e}_i\mathbf{e}_k)/\mathbf{e}_i^2.$$

In order to determine m_k uniquely, the right-hand inequality sign $\leqslant$ is replaced by $<$. This suggests the choice

$$m_k = [0.5 + (\mathbf{e}_i\mathbf{e}_k)/\mathbf{e}_i^2] \tag{10}$$

in the transformation T_i. Similarly, for the transformation T_i^* the choice

$$m_k^* = [0.5 + (\mathbf{e}_i^*\mathbf{e}_k^*)/\mathbf{e}_i^{*2}] \tag{11}$$

minimizes all $\mathbf{e}_k^{*2}$ for $k \neq i$. This shows that transformation T_i fulfills property (M) and T_i^* fulfills (M*) for the Euclidean norm.

It would be nice if T_i could also be guaranteed to satisfy (M*). Explicitly, this would mean that

$$Q_i^*(z_1^*, \ldots, z_n^*) = \Big(\mathbf{e}_i^* + \sum_{k\neq i} z_k^*\mathbf{e}_k^*\Big)^2$$

assumes its minimum at $z_k^* = m_k$. Differentiation of $Q_i^*(z_1^*, \ldots, z_n^*)$ leads to the system

$$\mathbf{e}_j^*\Big(\mathbf{e}_i^* + \sum_{k\neq i} z_k^*\mathbf{e}_k^*\Big) = \mathbf{e}_j^*\mathbf{e}_i^* + \sum_{k\neq i} z_k^*\mathbf{e}_j^*\mathbf{e}_k^* = 0 \quad \text{for } j \neq i. \tag{12}$$

The matrix $(q_{ik}) = (\mathbf{e}_i\mathbf{e}_k)$ is orthogonal to the matrix $(q_{kj}^*) = (\mathbf{e}_k^*\mathbf{e}_j^*)$. This follows from the definition of the dual basis: Let

$$\mathbf{E} = \begin{pmatrix} \text{—} & \mathbf{e}_1 & \text{—} \\ & \vdots & \\ \text{—} & \mathbf{e}_n & \text{—} \end{pmatrix} \quad \text{and} \quad \mathbf{E}^* = \begin{pmatrix} \text{—} & \mathbf{e}_1^* & \text{—} \\ & \vdots & \\ \text{—} & \mathbf{e}_n^* & \text{—} \end{pmatrix}$$

denote the matrices whose rows are given by the components of the bases $\mathbf{e}_i$ and $\mathbf{e}_k^*$. The defining relation $\mathbf{e}_i\mathbf{e}_k^* = \delta_{ik}$ reads in matrix notation $\mathbf{E}\mathbf{E}^{*T} = \mathbf{E}^*\mathbf{E}^T = \mathbf{I}$ where $\mathbf{I}$ is the unit matrix and $\mathbf{E}^T$ the transpose of $\mathbf{E}$. $\mathbf{E}^*\mathbf{E}^T = \mathbf{I}$ yields $\mathbf{E}^T\mathbf{E}^* = \mathbf{I}$. Hence the matrix product $(\mathbf{e}_i\mathbf{e}_k)(\mathbf{e}_k^*\mathbf{e}_j^*)$ is equal to $\mathbf{E}\mathbf{E}^T\mathbf{E}^*\mathbf{E}^{*T} = \mathbf{E}(\mathbf{E}^T\mathbf{E}^*)\mathbf{E}^{*T} = \mathbf{E}\mathbf{E}^{*T} = \mathbf{I}$. This proves the assertion.

Equating z_k^* to $q_{ik}/q_{ii} = \mathbf{e}_i\mathbf{e}_k/\mathbf{e}_i^2$ in (12) leads to

$$\sum_{k\neq i} z_k^*q_{kj}^* + q_{ij}^* = \Big(\sum_{k\neq i} q_{ik}q_{kj}^* + q_{ii}q_{ij}^*\Big)\Big/q_{ii} = \delta_{ij}/q_{ii} = 0.$$

Hence $Q_i^*(z_1^*, \ldots, z_n^*)$ assumes its minimal value at $z_k^* = q_{ik}/q_{ii}$. The value $m_k = [0.5 + q_{ik}/q_{ii}]$ is the nearest integer to q_{ik}/q_{ii}. This shows that $Q_i^*(m_1, \ldots, m_n)$ is near its minimal value. However, numerical examples show that the minimal value is sometimes assumed at a point $z_k^* \neq m_k$. In practice, this did not much influence the efficiency of the algorithm.

In the case of an actual increase in the number P of z_k-combinations, it would be better to reverse the responsible transformation T_i and proceed with T_{i-1} or T_{i+1}. However, this was not done in the trial runs in which the method worked quite well in spite of occasional increases in P.

It should be noted that the transformations T_i and T_i^* decrease the lengths $\|\mathbf{e}_i\|$ and $\|\mathbf{e}_i^*\|^*$ only with respect to the Euclidean norm. However, since the compact, convex set **B** has positive measure, it contains a ball $\underline{\mathbf{B}} = \{\mathbf{x} \in R^n \mid x_1^2 + \cdots + x_n^2 \leqslant \underline{r}\}$ and it is contained in a similar ball $\overline{\mathbf{B}}$. Consequently, a norm defined by this set **B** is equivalent to the Euclidean norm. Therefore, the same transformations T_i and T_i^* were used for calculating shortest vectors of any kind. In extensive numerical experiments, the transformations T_i and T_i^* led always to a final basis for which the value P in (9) was small. Hence a direct search for a vector of shortest length could be carried out.

It should be mentioned that both transformations T_i and T_i^* were always used. Examples were found where a mere application of transformation T_i^* led to a large value of P in (9). A single application of transformation T_i decreased this value considerably. Subsequently, transformations T_i^* were applied again and the value of P was further decreased.

In another experiment, the transformations T_i were applied more than once, each time T_i^* got stuck. But this did not improve performance, so it was finally decided to use T_i as little as possible.

3. The Computer Program. The complete algorithm can now be prepared. First of all, the bounds c_i in (8) are calculated for the given basis $\mathbf{e}_i$ of **G**; and the number $P = \Pi_{i=1}^n (2c_i + 1)$ of possible choices of the z_i is worked out. If P is small, a direct search becomes possible. Otherwise, the transformations T_i^* are applied to the basis $\mathbf{e}_i$. For this the m_k^* defined in (11) are calculated first and the corresponding transformation T_i^* is applied unless all m_k^* are zero. The process is stopped when n successive calculations of the m_k^* have not led to any successful transformation T_i^*, that is to decrease P. After n failures the transformation T_i is tried instead, subject to the same limit on trials. If P is decreased during T_i, a new attempt at transformation T_i^* is started immediately. Therefore final failure occurs eventually only after n unsuccessful trials on both T_i^* and T_i. Afterwards, the smallest value of $\|\mathbf{x}\|$ is found through an enumeration of vectors $\mathbf{x} = \Sigma_{i=1}^n z_i \mathbf{e}_i$ for which $-c_i \leqslant z_i \leqslant c_i$. Since vectors $(0, \ldots, 0, z_i, \ldots, z_n)$ and $(0, \ldots, 0, -z_i, \ldots, -z_n)$ lead to the same $\|\mathbf{x}\|$, the procedure may assume that the first nonzero component is positive. It can be shown that this reduces the complete enumeration from P to $(P-1)/2$ steps.

In the special case of the Euclidean norm in dimension 2, i.e. if $\|x\| = (x_1^2 + x_2^2)^{1/2}$, no final search is necessary. For, if $m_1 = m_2 = 0$ one has

$$-0.5 \leqslant (\mathbf{e}_1\mathbf{e}_2)/\mathbf{e}_2^2 \leqslant 0.5 \quad \text{and} \quad -0.5 \leqslant (\mathbf{e}_1\mathbf{e}_2)/\mathbf{e}_1^2 \leqslant 0.5.$$

This is equivalent to the classical condition of Gauss and Legendre that $2|\mathbf{e}_1\mathbf{e}_2| \leqslant \mathrm{Min}(\mathbf{e}_1^2, \mathbf{e}_2^2)$ holds for a reduced lattice basis. Hence, $\mathbf{e}_1$ or $\mathbf{e}_2$ is a vector of shortest Euclidean length, and its length is already contained in D.

Variants of this procedure are possible. Knuth suggests that one should apply the transformations T_i^* and T_i as long as P is greater than some given C, say $C = 1000$. In the examples to follow this increased the computation times considerably. In a few cases $P < 1000$ was never reached. The above method of continuing reduction until the transformations T_i^* and T_i are stuck is at least theoretically finite, although in prac-

tice the final enumeration may still cost too much time.

The complete procedure is now stated as a formal algorithm.

Algorithm S (*Vector of smallest norm* $\|\mathbf{x}\|$ *in* $\mathbf{G} = \{\mathbf{x} = \Sigma_{i=1}^{n} z_i \mathbf{e}_i \mid z_i$ *integers*$\}$).

1. Set $C \leftarrow D \leftarrow M$ (M very large), $m \leftarrow -1$, $i \leftarrow n+1$.

2. For $1 \leqslant k \leqslant n$ set $r_k \leftarrow \|\mathbf{e}_k\|$, $r_k^* \leftarrow \|\mathbf{e}_k^*\|^*$; and if $r_k < D$, set $D \leftarrow r_k$. Then for $1 \leqslant k \leqslant n$ set $c_k \leftarrow [D\, r_k^*]$ and work out $P = \Pi_{k=1}^{n} (2c_k + 1)$. If $P < C$, set $t^* \leftarrow t \leftarrow n$, $C \leftarrow P$ and go to 3. Otherwise, if $m = -1$, set $t^* \leftarrow t^* - 1$; but if $m = 1$, set $t \leftarrow t - 1$. If $t^* = 0$, go to 4.

3. (Transformation T^*.) If $t^* + t = 0$, go to 5. Set $i \leftarrow i - 1$; and if $i = 0$, set $i \leftarrow n$. For all $1 \leqslant k \leqslant n$ do: if $k \neq i$, set $m_k^* = [0.5 + \Sigma_{j=1}^{n} e_{ij}^* e_{kj}^* / \Sigma_{j=1}^{n} e_{ij}^{*2}]$. If all m_k^* are zero, set $t^* \leftarrow t^* - 1$; and if $t^* = 0$ go to 4, else restart 3. If at least one m_k^* is not zero, do for all $1 \leqslant j \leqslant n$: set $e_{ij} \leftarrow e_{ij} + \Sigma_{k \neq i} m_k^* e_{kj}$; and for $1 \leqslant k \leqslant n$ do: if $k \neq i$, set $e_{kj}^* \leftarrow e_{kj}^* - m_k^* e_{ij}^*$ $(j = 1, \ldots, n)$. Set $m \leftarrow -1$ and go to 2.

4. (Transformation T.) If $t^* + t = 0$, go to 5. Set $i \leftarrow i - 1$; and if $i = 0$, set $i \leftarrow n$. For all $1 \leqslant k \leqslant n$ do: if $k \neq i$, set $m_k = [0.5 + \Sigma_{j=1}^{n} e_{ij} e_{kj} / \Sigma_{j=1}^{n} e_{ij}^2]$. If all m_k are zero, set $t \leftarrow t - 1$; and if $t = 0$ go to 3, else restart 4. If at least one m_k is not zero, do for all $1 \leqslant j \leqslant n$: set $e_{ij}^* \leftarrow e_{ij}^* + \Sigma_{k \neq i} m_k e_{kj}^*$; and for $1 \leqslant k \leqslant n$ do: if $k \neq i$, set $e_{kj} \leftarrow e_{kj} - m_k e_{ij}$ $(j = 1, \ldots, n)$. Set $m \leftarrow 1$ and go to 2.

5. (Final Search.) For all combinations of integers $(z_1, \ldots, z_n) \neq (0, \ldots, 0)$ in which $-c_i \leqslant z_i \leqslant c_i$ for all $1 \leqslant i \leqslant n$ and for which the first nonzero component is positive, calculate $W = \|\Sigma_{i=1}^{n} z_i \mathbf{e}_i\|$; store the smallest of these values in D and the corresponding z_i in d_i.

6. Deliver the vector $\mathbf{x} = \Sigma_{i=1}^{n} d_i \mathbf{e}_i$ and its norm $D = \|x\|$.

4. Applications. The task of determining nonzero vectors of shortest length appeared early in number theory, especially in the theory of quadratic forms started by Gauss and continued by Hermite and, notably, Minkowski. Hermite and Minkowski derived global bounds for the vector of shortest Euclidean length; these bounds were not sharp, and sharp bounds are only proved up to dimension 10. Furthermore, Minkowski obtained global bounds for the norms $\|\mathbf{x}\| = \Sigma_{i=1}^{n} |x_i|$ and $\|\mathbf{x}\| = \max_{i=1,\ldots,n} |x_i|$. His main tool was his famous "convex body theorem". Sharp global bounds for these norms are only known for dimensions 2 and 3. Hopefully, this note will help to establish guesses for global bounds in higher dimensions.

Initially, the above algorithm was developed for investigating the lattice structure of pseudo-random numbers generated by the linear congruential method. Only the simplest case will be considered here. Construct a sequence of integers $\{z_i\}$ by

$$z_i \equiv a z_{i-1} \pmod{2^e}, \qquad z_0 \equiv 1 \pmod 4, \qquad 0 \leqslant z_i < 2^e \text{ and } a \equiv 5 \pmod 8.$$

Since the sequence $\{z_i\}$ contains all numbers of the form $4k + 1$, the fractions $u_i = z_i / 2^e$ are used as samples from the uniform distribution in $[0, 1)$. Consider the points $\mathbf{P}_n = (u_i, u_{i+1}, \ldots, u_{i+n-1})$ in the n-dimensional space R^n. Equating u_i and $(4k + 1)2^{-e}$, one obtains

$$\begin{aligned}\mathbf{P}_n &\equiv 2^{-e}(4k + 1, a(4k + 1), \ldots, a^{n-1}(4k + 1)) \pmod 1 \\ &\equiv \mathbf{e}_0 + k\mathbf{e}_1 \pmod 1 \quad \text{where } \mathbf{e}_1 = 2^{-(e-2)}(1, a, \ldots, a^{n-1}), \quad \mathbf{e}_0 = \tfrac{1}{4}\mathbf{e}_1.\end{aligned}$$

Here the integer k runs from 0 to $2^{e-2}-1$. If the integer k is smaller than 0 or greater than 2^{e-2}, the corresponding $\mathbf{P}_n = \mathbf{e}_0 + k\mathbf{e}_1$ is congruent modulo 1 to one of the former $\mathbf{P}_n$ for which $0 \leqslant k < 2^{e-2}$. Consequently, if one enlarges the set $\{\mathbf{P}_n\}$ to the set

$$\{\mathbf{Q}_n\} = \mathbf{e}_0 + \{k_1\mathbf{e}_1 + k_2\mathbf{e}_2 + \cdots + k_n\mathbf{e}_n \,|\, k_1, k_2, \ldots, k_n \text{ integers}\}$$

where

$$\mathbf{e}_1 = 2^{-(e-2)}(1, a, \ldots, a^{n-1}), \quad \mathbf{e}_2 = (0, 1, 0, \ldots, 0), \ldots, \quad \mathbf{e}_n = (0, \ldots, 0, 1),$$

the new set $\{\mathbf{Q}_n\}$ is the translate of a lattice $\mathbf{G}$ generated by $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n$. Its dual lattice $\mathbf{G}^*$ generated by

$$\mathbf{e}_1^* = (2^{e-2}, 0, \ldots, 0), \quad \mathbf{e}_2^* = (-a, 1, 0, \ldots, 0), \ldots, \quad \mathbf{e}_n^* = (-a^{n-1}, 0, \ldots, 0, 1)$$

has a simple geometric meaning: $\mathbf{G}^*$ corresponds uniquely to the set of parallel hyperplanes $\mathbf{x}^*\mathbf{x} \equiv 0 \pmod 1$, on which all points of $\mathbf{G}$ lie. This may be proved as follows: First, all points of $\mathbf{G}$ lie on the set of hyperplanes $\mathbf{x}^*\mathbf{x} \equiv 0 \pmod 1$ where $\mathbf{x}^* = \Sigma_{i=1}^n z_i^*\mathbf{e}_i^*$ is a fixed element of $\mathbf{G}^*$ (which means z_i^* integral). Conversely, if the set of hyperplanes $\mathbf{x}^*\mathbf{x} = (\Sigma_{i=1}^n z_i^*\mathbf{e}_i^*)\mathbf{x} \equiv 0 \pmod 1$, z_i^* fixed, contains all points of $\mathbf{G}$, it contains especially the points $\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n$. Consequently, z_i^* has to be integral.

For qualifying random number generators, the following questions can now be answered:

(i) Determine the minimal number of parallel hyperplanes on which all points $\mathbf{P}_n$ lie.

(ii) Determine the maximal distance of parallel hyperplanes on which all points $\mathbf{P}_n$ lie.

For (i) one has to compute

$$N_n^* = \operatorname{Min}\left\{\sum_{i=1}^n |x^*| \,\middle|\, \mathbf{x}^* = (x_1^*, \ldots, x_n^*) \in \mathbf{G}^*, \mathbf{x}^* \neq 0\right\},$$

since the right-hand side of the equation $\mathbf{x}^*\mathbf{x} = \Sigma_{i=1}^n x_i^* x_i = v$ can only attain the values v for which $-\Sigma_{i=1}^n (x_i^*)_- < v < \Sigma_{i=1}^n (x_i^*)_+$. Here x_+ and x_- are defined by

$$x_+ = \begin{cases} x & \text{if } x \geqslant 0, \\ 0 & \text{if } x < 0, \end{cases} \qquad x_- = \begin{cases} 0 & \text{if } x \geqslant 0, \\ -x & \text{if } x < 0. \end{cases}$$

The number of these v is equal to

$$\sum_{i=1}^n (x_i^*)_+ + \sum_{i=1}^n (x_i^*)_- - 1 = \sum_{i=1}^n |x_i^*| - 1.$$

For (ii) one has to calculate

$$D_n^* = \operatorname{Max}\left\{1\Big/\left(\sum_{i=1}^n (x_i^*)^2\right)^{1/2} \,\middle|\, \mathbf{x}^* = (x_1^*, \ldots, x_n^*) \in \mathbf{G}^*, \mathbf{x}^* \neq 0\right\} = 1/W_n^*$$

where

$$W_n^* = \operatorname{Min}\left\{\left(\sum_{i=1}^n (x_i^*)^2\right)^{1/2} \,\middle|\, \mathbf{x}^* = (x_1^*, \ldots, x_n^*) \in \mathbf{G}^*, \mathbf{x}^* \neq 0\right\}.$$

Question (i) was raised by G. Marsaglia in his famous paper [4], where he derived upper bounds for N_n^* using Minkowski's 'convex body theorem'. The table below con-

tains exact values of N_n^* for some random number generators; the Minkowski bounds are listed at the end of the table.

Question (ii) was considered by R. R. Coveyou and R. D. MacPherson in their Fourier analysis of random number generators. For this purpose they developed an algorithm to calculate W_n^*; the algorithm was improved by D. E. Knuth in [3, pp. 89–97]. The systematic use of the dual basis as outlined above simplified the algorithm considerably in this special case of the Euclidean norm.

Table of values of N_n^* *and* $[W_n^*]$ *(in parentheses) for some generators* mod 2^{31}

$a \pmod{2^{31}}$	N_2^*	N_3^*	N_4^*	N_5^*	N_6^*
65 533	32 765 (23 169)	15 (10)	15 (10)	15 (10)	15 (10)
258 585 933	22 107 (17 440)	1 115 (698)	257 (146)	69 (40)	31 (17)
414 536 077	27 307 (19 758)	1 115 (781)	209 (124)	91 (49)	41 (20)
Minkowski bounds for N_n^*	32 768	1 476	336	145	85

Institut für Mathematische Statistik
Technische Hochschule in Graz
A 8010 Graz, Austria

Visiting Department of Statistics
Stanford University
Stanford, California 94305

1. R. R. COVEYOU & R. D. MACPHERSON, "Fourier analysis of uniform random number generators," *J. Assoc. Comput. Mach.*, v. 14, 1967, pp. 100–119. MR 36 #4779.

2. U. DIETER & J. H. AHRENS, *Pseudo-Random Numbers*, Preliminary version in preprint (430 pages), Wiley, New York. (To appear.)

3. D. E. KNUTH, *The Art of Computer Programming*. Vol. 2: *Seminumerical Algorithms*, Addison-Wesley, Reading, Mass., 1969. MR 44 #3531.

4. G. MARSAGLIA, "Random numbers fall mainly in the planes," *Proc. Nat. Acad. Sci. U. S. A.*, v. 61, 1968, pp. 25–28. MR 18, 947.

5. H. MINKOWSKI, *Gesammelte Abhandlungen*, especially Vol. I, pp. 243–260, Vol. II, pp. 3–42, Teubner-Verlag, Leipzig, 1911.