
Efficient Search for Two-Dimensional Rank-1 Lattices with Applications in Graphics

Sabrina Dammertz¹, Holger Dammertz¹, and Alexander Keller²

¹ Institute of Media Informatics, Ulm University, Germany

{sabrina, holger}.dammertz@uni-ulm.de

² mental images GmbH, Berlin, Germany

alex@mental.com

Summary. Selecting rank-1 lattices with respect to maximized mutual minimum distance has been shown to be very useful for image representation and synthesis in computer graphics. While algorithms using rank-1 lattices are very simple and efficient, the selection of their generator vectors often has to resort to exhaustive computer searches, which is prohibitively slow. For the two-dimensional setting, we introduce an efficient approximate search algorithm and transfer the principle to the search for maximum minimum distance rank-1 lattice sequences. We then extend the search for rank-1 lattices to approximate a given spectrum and present new algorithms for anti-aliasing and texture representation in computer graphics.

1 Introduction

Due to their algorithmic efficiency, rank-1 lattices [17, 20] and rank-1 lattice sequences [9, 10] are very interesting objects for computer graphics [3, 4]: The n points \mathbf{x}_i of an s -dimensional rank-1 lattice

$$L_{n,\mathbf{g}} := \left\{ \mathbf{x}_i := \frac{i}{n} \mathbf{g} \bmod 1 : i = 0, \dots, n-1 \right\} \subset [0, 1]^s \quad (1)$$

are generated by a suitable vector $\mathbf{g} \in \mathbb{N}^s$. Rank-1 lattices $L_{n,a}$ in Korobov form [20] use generator vectors of the restricted form $\mathbf{g} = (1, a, a^2, \dots, a^{s-1})$.

Using a van der Corput sequence (radical inverse) Φ_b in base b [17] instead of the fraction $\frac{i}{n}$ extends rank-1 lattices to rank-1 lattice sequences

$$L_{\mathbf{g}}^{\Phi_b} := \{ \mathbf{x}_i := \Phi_b(i) \cdot \mathbf{g} \bmod 1 : i \in \mathbb{N}_0 \} \subset [0, 1]^s \quad (2)$$

in the sense that for any $m \in \mathbb{N}_0$ the first b^m points $\mathbf{x}_0, \dots, \mathbf{x}_{b^m-1}$ are a rank-1 lattice $L_{b^m, \mathbf{g}}$ [10]. Thereby the van der Corput sequence Φ_b mirrors the b -ary representation of an integer i at the decimal point

$$\begin{aligned} \Phi_b(i) : \mathbb{N}_0 &\longrightarrow \mathbb{Q} \cap [0, 1) \\ i = \sum_{j=0}^{\infty} a_j(i)b^j &\longmapsto \sum_{j=0}^{\infty} a_j(i)b^{-j-1}, \end{aligned} \quad (3)$$

where $a_j(i)$ denotes the j -th digit of the integer i represented in base b .

In [4] we investigated the concept of maximized minimum distance (MMD) rank-1 lattices with applications to image synthesis and representation. Since lattices are closed under addition and subtraction, the minimum distance

$$d_{\min}(L_{n,\mathbf{g}}) := \min_{0 < i < n} \|\mathbf{x}_i\| \quad (4)$$

of a rank-1 lattice $L_{n,\mathbf{g}}$ is determined by the minimum norm of the lattice points themselves. In this paper we use the L_2 -norm on the unit torus unless noted otherwise.

Algorithms for computing the shortest vector in a general lattice have been developed in [5, 7, 12] and efficient implementations exist even for higher dimensions [14]. Specializing the setting to rank-1 lattices in two dimensions allows one to take simpler approaches, as for example the Gaussian basis reduction [11, 18]. This basis reduction is a simple algorithm to efficiently determine a lattice basis where the first basis vector is the shortest vector in the lattice and thus yields its minimum distance. In two dimensions, this algorithm computes a Minkowski-reduced basis and has a computational complexity of $\mathcal{O}(\log n)$ which is sufficient for our application [11].

The problem of constructing lattices with longest possible shortest nonzero vectors for a given lattice density is connected to the problem of finding the densest packing of spheres which has been studied for a long time [1, 16, 19]. Computer searches for good lattices based on the lengths of shortest nonzero vectors have been reported in [13, 15] for example. They focus on the dual lattice, though, and use either exhaustive or random searches, the latter of which poses the problem of deciding how much time to spend on the search process. Due to the low-dimensional structure of many graphics applications, we will consider only $s = 2$ dimensions henceforth. However, the number of potential generator vectors for the number n of points required in graphics applications is so large that a naïve search algorithm for MMD rank-1 lattices as well as tables become prohibitive in time and space. For image storage or sampling it is not uncommon have $n > 4000^2$.

We present efficient approximate search algorithms for MMD rank-1 lattices and sequences, and introduce a method that searches rank-1 lattices to better represent and integrate functions with an anisotropic Fourier spectrum. The findings result in new algorithms for anti-aliasing and texture representation [3], i.e. numerical integration and function approximation.

2 Efficient Search by Restricting the Search Space

There exists a construction for MMD rank-1 lattices [2], where the generator vector \mathbf{g} and the number of lattice points n are described by the sequence of convergents of the continued fraction equal to $\sqrt{3}$. However, the number n of points of this construction increases very fast, reducing their applicability in practical applications. For other n the generator vector has to be determined by computer search. The naïve algorithm enumerates all possible generator vectors in order to find MMD lattices. Already for only $s = 2$ dimensions, scanning $\mathcal{O}((n-1)^2)$ candidates becomes prohibitive for large n as used in our applications. Restricting the search space to lattices in Korobov form (i.e. $\mathbf{g} = (1, a)$), the minimum distance can be determined efficiently, as described in [12]. However, not all MMD rank-1 lattices can be represented in Korobov form [4]. For example the MMD rank-1 lattice for $n = 56$ points has the generator vector $\mathbf{g} = (4, 7)$. In the following we examine a restriction of the search space for which the efficient search algorithm resembles rasterization algorithms as used in computer graphics. This search is not restricted to Korobov lattices and we show that it can find MMD rank-1 lattices that cannot be represented in Korobov form. This allows a much more flexible use of rank-1 lattices.

2.1 Approximate Search for MMD Rank-1 Lattices

In order to enable an accelerated search for MMD rank-1 lattices we restrict the search space to a small subset of all possible lattice generator vectors. We base our restriction on two observations: First, for any lattice there is more than one generator vector for the identical lattice. For example if the number n of lattice points is prime, all lattice points scaled by n are generator vectors and thus the shortest vector is generator vector, too. For arbitrary n we noticed that it is still often the case that the shortest vector is also a generator vector. The second observation is, that the largest possible minimum distance l would result from a point set, whose triangulation consists of only equilateral triangles [1] (analogue to hexagonal lattices). This distance is an upper bound on the maximized minimum distance that can only be approximated by rank-1 lattices. Equating the area $A = \frac{1}{n}$ of the basis cell of a rank-1 lattice and twice the area of such an equilateral triangle of side length l yields

$$A = \frac{1}{n} = 2 \left(\frac{1}{2} \cdot l \cdot h \right), \quad h = l \cdot \frac{\sqrt{3}}{2} \quad \iff \quad l = \sqrt{\frac{2}{n \cdot \sqrt{3}}}. \quad (5)$$

With the assumption that the generator vector is also the shortest vector it would suffice to search the integer generator vector only within a circle of the radius $n \cdot l$. However as noted above this is not always the case. Experiments showed that using a slightly larger upper bound allows one to find better lattices. To perform the approximate search we restrict the search space for the generator vectors \mathbf{g} to a ring around the origin with inner radius r and

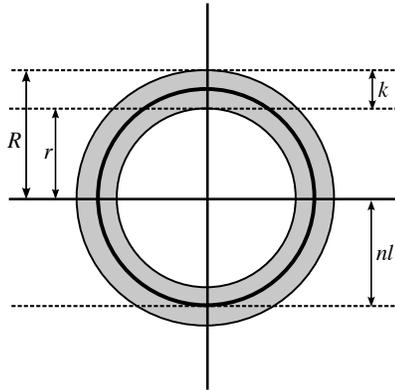


Fig. 1. Idea of the restricted search space.

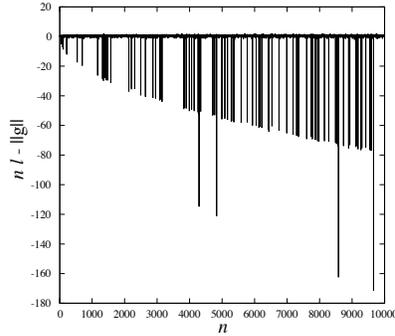


Fig. 2. Difference $n \cdot l - \|\mathbf{g}\|$ of the maximally possible length l scaled by n and the shortest generator vector of the exhaustive search for $n = 4, \dots, 10000$.

outer radius R , where $r = n \cdot l - \frac{k}{2} < n \cdot l < n \cdot l + \frac{k}{2} = R$ and k is a selected positive integer (see Figure 1).

By rasterizing this ring on the integer lattice \mathbb{Z}^2 using efficient algorithms from computer graphics [8], all potential generator vectors are enumerated. However, due to symmetry only one eighth of the ring needs to be rasterized (see Figure 3). Fixing the ring width k independent of n , the rasterization runs in $\mathcal{O}(n \cdot l) = \mathcal{O}(\sqrt{n})$ time. The approximate search then runs in $\mathcal{O}(\sqrt{n} \log n)$ time, where the minimum distances are computed using the Gaussian basis reduction.

Restriction of the Search Space

We computed the difference $n \cdot l - \|\mathbf{g}\|$ for $n = 4, \dots, 10000$, where $\|\mathbf{g}\|$ is the length of the shortest generator vectors found by the exhaustive search. Generator vectors are integer vectors and therefore l has to be scaled by n . Note that when the generator vector of the MMD lattice is not the shortest vector the difference can be negative. The graph in Figure 2 justifies the approach to restrict the search space to a ring of a fixed width. Due to the complexity of the exhaustive search, the range of $n > 10000$ has been investigated for random samples only. An empirically chosen value of $k = 6$ has proven to be a reasonable ring width as described now.

Numerical Evidence

For $n = 4, \dots, 10000$ and $k = 6$ we now compare the approximate rasterization search to the exact exhaustive search. In 99.1% (i.e. 9908 out of 9997 cases), the approximate algorithm finds the optimal generator vector with respect to maximized minimum distance. The percentage of lattices for which a generator

vector coincides with a shortest vector equals 71% (7098 cases), whereas in 28.1% (2810 cases) a generator vector producing a lattice with maximum possible minimum distance is determined inside the ring with width k even if the generator vector is not the shortest vector. Otherwise the new search algorithm yields a maximized minimum distance that is never worse than 90% of the optimum.

The restricted search always yields the correct results for n being prime. We showed above that it is likely to also find generator vectors for MMD rank-1 lattices for arbitrary n . Additionally if the best lattice is not found, at least an acceptable one is found (i.e. one with a minimum distance not worse than 90% of the optimum). Examples for the different cases are visualized in Figure 3. The search space is depicted by the light gray squares, which represent the rasterized region of a ring with radius $n \cdot l$ and width $k = 6$. Due to a very simple rasterization algorithm the rasterized region is slightly larger than required. The light gray circle is of radius $n \cdot l$, while the black circle's radius is the maximized minimum distance MMD_e determined by the exhaustive search. The set of generator vectors which result from this exhaustive search algorithm and lie in the displayed range are plotted using hollow dots. The solid dots belong to the lattice generated by the displayed vector as one element of the generator vectors resulting from the approximate search with maximized minimum distance MMD_r .

In order to show the improvements of our new algorithm we compare the best lattice found in Korobov form with minimum distance (MMD_k) and the resulting maximized minimum distance using the approximate search (MMD_r). In Figure 4 the ratio MMD_k/MMD_r for $n = 4, \dots, 10000$ is plotted. As is apparent from the graph the new search yields nearly optimal results with respect to the search criterion and delivers better results than the Korobov form in most cases. More precisely $MMD_r \geq MMD_k$ in 99.1% of the cases, of which for 6.2% we have $MMD_r > MMD_k$, if the MMD rank-1 lattice

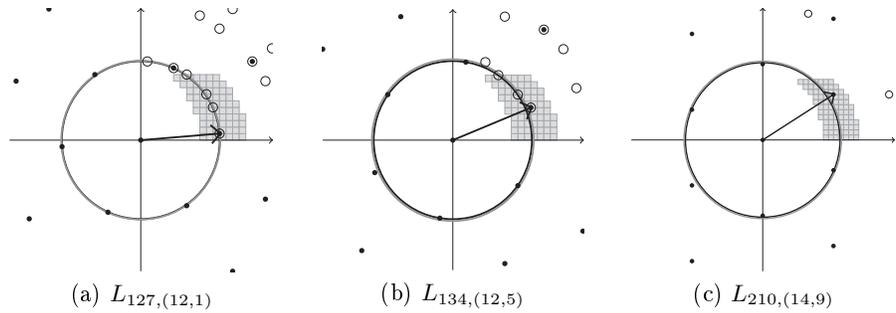


Fig. 3. Illustration of the rasterization search. (a) $n = 127$, $MMD_r = MMD_e$. The generator vector $\mathbf{g} = (12, 1)$ is a shortest vector in the lattice. (b) $n = 134$, $MMD_r = MMD_e$. $\mathbf{g} = (12, 5)$ does not correspond to a shortest vector. (c) $n = 210$, $MMD_r < MMD_e$.

cannot be represented in Korobov form, and $\text{MMD}_r = \text{MMD}_k$ in 92.9% of the cases.

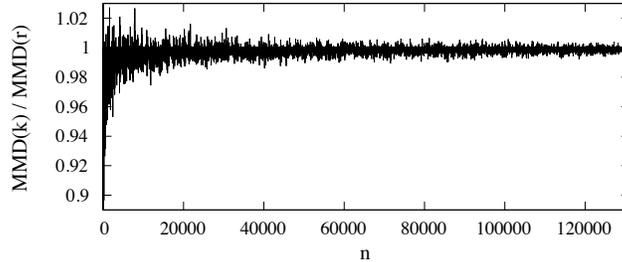


Fig. 4. Ratio $\text{MMD}_k/\text{MMD}_r$ for $n \in [4, 131072)$.

2.2 Search for MMD Rank-1 Lattice Sequences

Using a fixed n for the number of lattice points is often insufficient for graphics applications. For example hierarchical representations of images or progressive sampling need a varying number of sample points. Lattice sequences can provide this functionality and we examine two approaches in this section how to construct rank-1 lattice sequences with MMD property.

As defined in Section 1, a rank-1 lattice sequence $L_{\mathbf{g}}^{\Phi_b}$ contains a sequence of rank-1 lattices $L_{b^m, \mathbf{g}}$ for $m \in \mathbb{N}_0$. We search for rank-1 lattice sequences with maximized minimum distance in the sense that the weighted sum

$$\sum_{m=m_{\min}}^{m_{\max}} (d_{\min}(L_{b^m, \mathbf{g}}))^2 b^m \quad (6)$$

is maximized. Scaling the squared minimum distance by b^m assigns equal importance to all lattices of the sequence since the area of a basis cell is $\frac{1}{b^m}$.

Lattice Sequences based on an Initial MMD Rank-1 Lattice

One way of constructing a MMD rank-1 lattice sequence is by taking a generator vector \mathbf{g} of a MMD rank-1 lattice $L_{b^m, \mathbf{g}}$ and using it in Equation (2). For $q \in \mathbb{N}_0$ and a fixed m , each set of points $\{x_{q \cdot b^m}, \dots, x_{(q+1)b^m - 1}\} \subset L_{\mathbf{g}}^{\Phi_b}$ is a copy of $L_{b^m, \mathbf{g}}$ shifted by $\Delta(q) := \Phi_b(q)b^{-m}\mathbf{g}$ [10]. The minimum distance of all copies is identical, as d_{\min} is shift invariant. For the example of $L_{(1,3)}^{\Phi_2}$ this structural property [9] is depicted in Figure 5. We now consider a two-dimensional generator vector $\mathbf{g} = (g_1, g_2)$ with $\gcd(n, g_1, g_2) = 1$. Then all

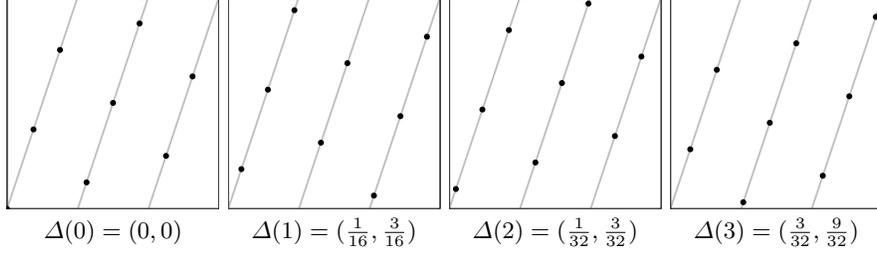


Fig. 5. The shifted lattices $L_{8,(1,3)} + \Delta(q) = L_{8,(1,3)} + \Phi_2(q)2^{-3}(1, 3)$ for $q = 0, 1, 2, 3$ from the lattice sequence $L_{(1,3)}^{\Phi_2}$.

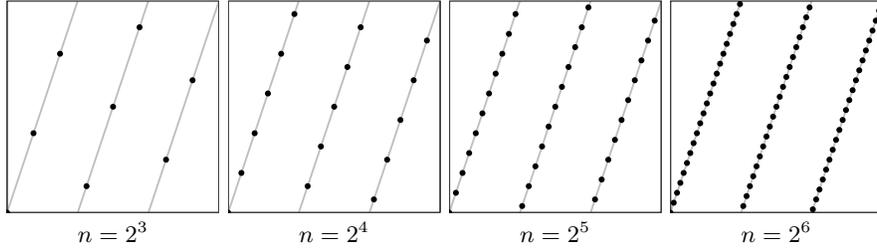


Fig. 6. The lattices $L_{2^m,(1,3)}$ of the lattice sequence $L_{(1,3)}^{\Phi_2}$ started with the initial MMD rank-1 lattice $L_{8,(1,3)}$.

points of the rank-1 lattice sequence $L_{\mathbf{g}}^{\Phi_b}$ lie on at most $n_h = g_1 + g_2 - 1$ hyperplanes, independent on the number of points. As a consequence, all points of the previous example $L_{(1,3)}^{\Phi_2}$ reside on three hyperplanes (induced by the generator vector), as illustrated in Figure 6. This means that the generator vector has to be modified such that the undesirable uniform bound on the minimum distance induced by the number of hyperplanes is improved.

Considering generator vectors of the form

$$\mathbf{g}_{i,j} := (g_1 + i \cdot b^m, g_2 + j \cdot b^m) \text{ for } i, j \in \mathbb{N}_0,$$

we have $\mathbf{g}_{i,j} \equiv \mathbf{g} \pmod{b^m}$. As a consequence $L_{b^m, \mathbf{g}} = L_{b^m, \mathbf{g}_{i,j}}$, i.e. the minimum distance remains unchanged for b^m points. However, the upper bound on the number of hyperplanes is increased to $n_h = g_1 + i \cdot b^m + g_2 + j \cdot b^m - 1$, as desired. For example $L_{(41,11)}^{\Phi_2}$ with $(41, 11) = (1 + 5 \cdot 8, 3 + 1 \cdot 8)$ does not restrict points to only three hyperplanes, but for $n = 8$ points generates the same rank-1 lattice as $L_{(1,3)}^{\Phi_2}$, i.e. $L_{8,(1,3)} = L_{8,(41,11)}$ (compare Figures 6 and 7).

The search procedure is started by selecting both a minimum number of points $b^{m_{\min}}$ and maximum $b^{m_{\max}}$. First a search of the previous section is run to find an initial MMD rank-1 lattice generator vector \mathbf{g} for $b^{m_{\min}}$ points. Then the sum of minimum distances (6) is evaluated for each potential generator

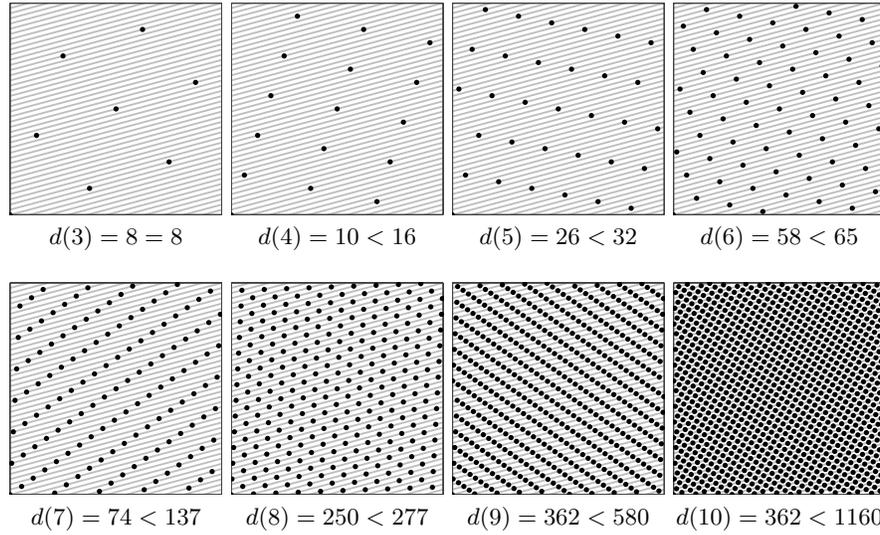


Fig. 7. Searching an MMD rank-1 lattice sequence for the initial lattice $L_{2^3, (1,3)}$ (see Figure 6) and $m_{\max} = 7$, yields $L_{(41,11)}^{\Phi_2}$ with $\mathbf{g}_{5,1} = (1 + 5 \cdot 2^3, 3 + 1 \cdot 2^3) = (41, 11)$. The gray lines show all possible hyperplanes. For each lattice of the rank-1 lattice sequence we compare its minimum distance $d(m) := d_{\min}(L_{b^m, \mathbf{g}})^2 b^m$ to the maximum minimum distance that can be obtained by a single MMD rank-1 lattice.

vector $\mathbf{g}_{i,j}$ in order to find the maximum, where the search range is determined by

$$g_1 + i \cdot b^{m_{\min}} \leq b^{m_{\max}} \Rightarrow i \leq \frac{b^{m_{\max}} - g_1}{b^{m_{\min}}} < b^{m_{\max} - m_{\min}} \text{ and}$$

$$g_2 + j \cdot b^{m_{\min}} \leq b^{m_{\max}} \Rightarrow j \leq \frac{b^{m_{\max}} - g_2}{b^{m_{\min}}} < b^{m_{\max} - m_{\min}}.$$

Due to symmetry, an obvious optimization is to bound the range of j by $b^{m_{\max} - m_{\min}} - i$. Again, minimum distances are computed using the Gaussian basis reduction. An example result of the search is illustrated in Figure 7, where minimum distances obtained by the rank-1 lattice sequence are compared to the distances that can be obtained by rank-1 lattices alone.

Approximate Search by Restricting the Search Space

In the second approach the search is not based on an initial MMD rank-1 lattice. Instead we choose $m_{\min} = 1$ and fix a value for m_{\max} , looking for a generator vector that maximizes Equation (6).

In order to accelerate the search process, the search space can be restricted using the same strategy as in the rasterization search algorithm for

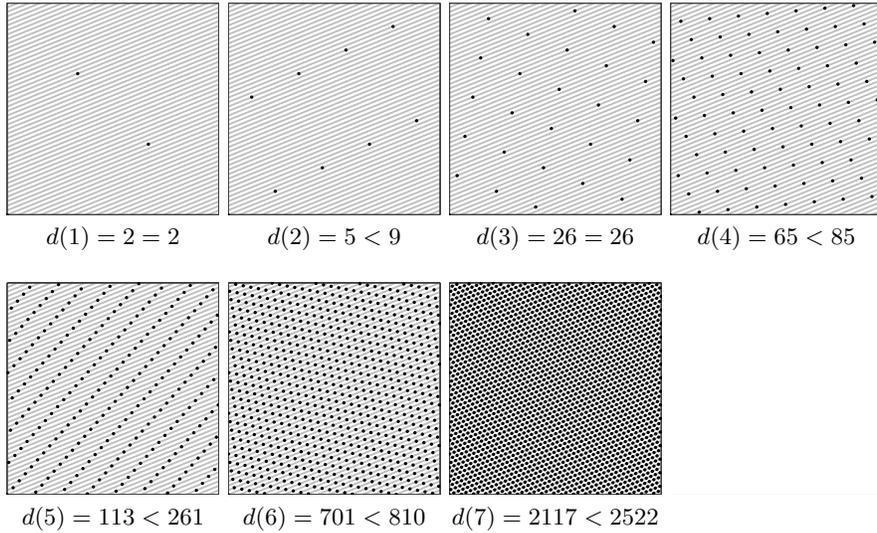


Fig. 8. $L_{(47,19)}^{\Phi_3}$ in base $b = 3$. For each lattice of the rank-1 lattice sequence we compare its minimum distance $d(m) := d_{\min}(L_{b^m, \mathbf{g}})^2 b^m$ to the maximum minimum distance that can be obtained by a single MMD rank-1 lattice.

rank-1 lattices (see Section 2.1). Then the search space is the union of the restricted search spaces for $L_{b^m, \mathbf{g}}$, $1 < m \leq m_{\max}$. In experiments, the restricted search achieved the same results as the exhaustive computer search for $n_{\max} := b^{m_{\max}} \leq 256$ and $b = 2, 3, 4$, simultaneously reducing the run-time from $\mathcal{O}(n_{\max}^2 \log n_{\max})$ to $\mathcal{O}(\sqrt{n_{\max}} \log n_{\max})$.

We compare the two search approaches presented in this section by summing the minimum distances of the first b^m points of each sequence

$$\sum_{m=2}^7 d(m) = \sum_{m=2}^7 d_{\min}(L_{b^m, \mathbf{g}})^2 b^m.$$

Although the second approach is more general than the first one, the lattices produced by the sequence might not necessarily have the maximal possible minimum distance for the corresponding $n = b^m$ points, which is assured at least for the initial lattice in the first approach. Figure 8 shows the resulting lattice sequence for $b = 3$ and $m_{\max} = 7$, while Table 1 shows the results of the numerical comparison. By definition for $m = 2$ the lattice of the sequence $L_{(82,129)}^{\Phi_3}$ represents an MMD rank-1 lattice, whereas for $m = 3$ the rank-1 lattice of the sequence $L_{(47,19)}^{\Phi_3}$ achieves the largest possible minimum distance as well.

m	$d(m)$ first approach	$d(m)$ second approach
2	9	5
3	25	26
4	34	65
5	229	113
6	745	701
7	1033	2117
Σ	2075	3027

Table 1. Comparing the lattice sequences $L_{(82,129)}^{\Phi_3}$ and $L_{(47,19)}^{\Phi_3}$ with respect to the minimum distance of the first b^m points of the lattice sequence for $b = 3$ and $2 \leq m \leq 7$. The initial MMD rank-1 lattice for $L_{(82,129)}^{\Phi_3}$ is given by $L_{3^2,(1,3)}$.

3 Search of Anisotropic Rank-1 Lattices to Approximate Spectra

In many graphics applications the image functions exhibit a strong anisotropic behavior in their Fourier spectrum. By constructing rank-1 lattices with knowledge of these functions the image quality can be improved. The Fourier transform of the Shah function

$$\text{III}_{L_{n,\mathbf{g}}}(\mathbf{x}) := \sum_{\mathbf{p} \in \mathbb{Z}^s} \delta(\mathbf{x} - B \cdot \mathbf{p})$$

over the lattice $L_{n,\mathbf{g}}$ with basis B , where $\delta(\mathbf{x})$ is Dirac's delta function, yields another Shah function over its dual lattice $L_{n,\mathbf{g}}^\perp$ [6]. This means that we can describe the spectrum $\mathcal{S}_{n,\mathbf{g}}$ of $L_{n,\mathbf{g}}$ by the fundamental Voronoi cell of the dual lattice $L_{n,\mathbf{g}}^\perp$. We characterize the shape of this cell by two parameters, namely by its orientation $\vec{\omega}_L$ and by its width w_L , which are computed by means of the basis B^\perp of $L_{n,\mathbf{g}}^\perp$. Given a lattice basis $B = (\mathbf{b}_1 \mathbf{b}_2)^\dagger$, where \dagger means transposed, the dual basis can be easily determined by $B^\perp = (B^{-1})^\dagger$. In order to assure that B^\perp spans the Delaunay triangulation and thus the Voronoi diagram, the dual basis has to be reduced, for example using the Gaussian basis reduction. Let

$$\mathbf{v} := \begin{cases} \mathbf{b}_1^\perp + \mathbf{b}_2^\perp & \text{if } \mathbf{b}_1^\perp \cdot \mathbf{b}_2^\perp < 0 \\ \mathbf{b}_2^\perp - \mathbf{b}_1^\perp & \text{otherwise} \end{cases}$$

be the diagonal of the basis cell spanned by \mathbf{b}_1^\perp and \mathbf{b}_2^\perp , such that \mathbf{v} and \mathbf{b}_1^\perp or \mathbf{v} and \mathbf{b}_2^\perp form a valid basis of the dual lattice as well. Then we approximate the orientation of the fundamental Voronoi cell by

$$\vec{\omega}_L := \mathbf{b}_2^\perp + \mathbf{v} = \begin{cases} 2 \cdot \mathbf{b}_2^\perp + \mathbf{b}_1^\perp & \text{if } \mathbf{b}_1^\perp \cdot \mathbf{b}_2^\perp < 0 \\ 2 \cdot \mathbf{b}_2^\perp - \mathbf{b}_1^\perp & \text{otherwise.} \end{cases} \quad (7)$$

The width w_L of $\mathcal{S}_{n,\mathbf{g}}$ is defined as the length of the shortest basis vector normalized by the hexagonal bound l , i.e.

$$w_L = \frac{\|\mathbf{b}_1^\perp\|}{l \cdot n}. \quad (8)$$

Note that $l \cdot n$ also represents an upper bound on the maximized minimum distance of the dual lattice, as the length of shortest vector in $L_{n,\mathbf{g}}^\perp$ corresponds to the length of the shortest vector in $L_{n,\mathbf{g}}$ scaled by n [4].

The spectrum $\mathcal{T}_{\mathbf{d},w}$, according to which we want to search the rank-1 lattice, is specified by its main direction, i.e. orientation, $\mathbf{d} \in \mathbb{R}^2$ and its width w . The two-dimensional vector \mathbf{d} and the scalar w are passed as an input parameter to the lattice search by an application. The width w takes values in the range of $[0, 1]$ and represents the measure of desired anisotropy. The most anisotropic spectrum is denoted by $w = 0$, whereas $w = 1$ represents the isotropic one. Note that we have to allow $g_i = 0$, $i = 1, 2$ for the generator vector in order to be able to approximate spectra aligned to axes of the Cartesian coordinate system.

For $n \in \mathbb{N}$ the search algorithm steps through all distinct lattices. This can be realized for example by using an $n \times n$ array, where the generator vectors of identical lattices are marked. Given any $\mathbf{g} \in [0, n)^2$, the set of vectors yielding identical lattices is $\{k \cdot \mathbf{g} \bmod n : \gcd(n, k) = 1, k = 1, \dots, n-1\}$. After computing a Minkowski-reduced basis of the dual lattice, the orientation and width of the fundamental Voronoi cell are determined according to Equations (7) and (8). Then the lattices are sorted with respect to $|w_L - w|$. For the smallest difference we choose the lattice, whose orientation $\vec{\omega}_L$ best approximates the main direction \mathbf{d} of $\mathcal{T}_{\mathbf{d},w}$. Thereby the similarity

$$\text{sim} = \frac{\mathbf{d} \cdot \vec{\omega}_L}{\|\mathbf{d}\| \cdot \|\vec{\omega}_L\|}$$

between those two vectors is measured by calculating the cosine of the angle between $\vec{\omega}_L$ and \mathbf{d} . Figure 9 shows an example for anisotropic rank-1 lattices having $n = 56$ points, where the spectrum is specified by $\mathbf{d} = (\cos \alpha, \sin \alpha)$ with $\alpha = 303^\circ$ and the width varies from 0.1 to 1.0 in steps of 0.1. Using the Gaussian basis reduction for the lattice basis search, the algorithm runs in $\mathcal{O}(n^2 \log n)$ time.

4 Weighted Norms

So far we considered rank-1 lattices only on the unit square. However, graphics applications often require arbitrary rectangular regions. Just selecting a corresponding region of the lattice defined in the entire real space and scaling it to the unit square is not an option as this would destroy for example the needed periodicity and complicate address computations in image applications. We now show how to extend our approximate search for isotropic and anisotropic rank-1 lattices to such regions.

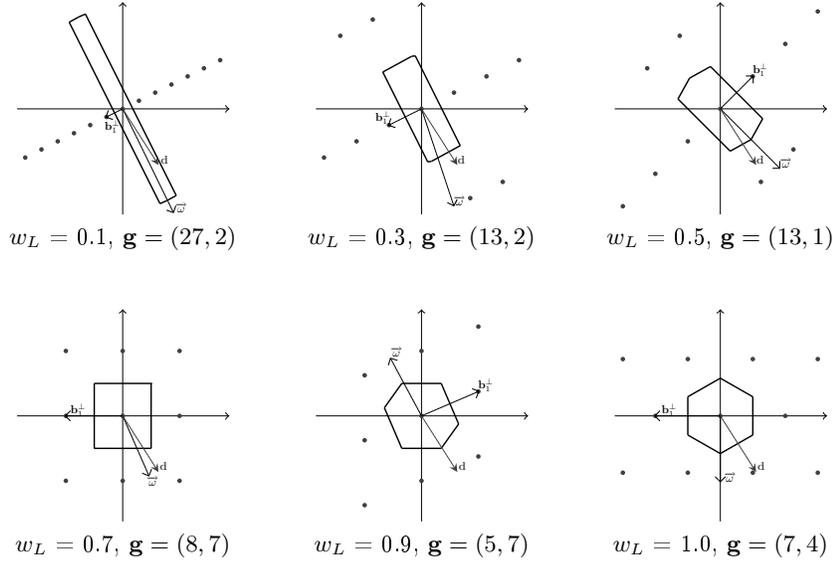


Fig. 9. Resulting spectra for a fixed direction $\mathbf{d} = (\cos 303^\circ, \sin 303^\circ)$ and width varying from 0.1 to 1.0.

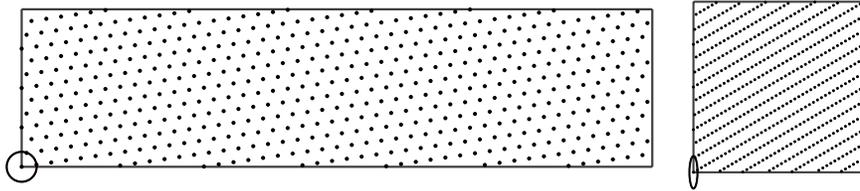


Fig. 10. Searching on a rectangular domain. Left: MMD rank-1 lattice $L_{512, (4, 45)}$ in a domain of width-to-height ratio $x : y = 4 : 1$ in world coordinates. Right: The same lattice in the scaled basis with $x : y = 1 : 1$. The search region becomes an ellipse.

All that needs to be done is considering the weighted norm $\|B^r \mathbf{x}_i\|$ in the definition of the minimum distance in Equation (4) instead of the Euclidean norm where B^r describes the transformation of the unit square to the desired region. Note that as before the distance to the origin has to be computed with respect to the unit torus.

Approximate Search for MMD rank-1 lattices

For the special case of scaled rectangular domains, i.e. $B^r = (\mathbf{b}_1^r \mathbf{b}_2^r) = ((x, 0)^t (0, y)^t)$, the rasterization search can be adapted easily. Therefore the lattice basis B has to be transformed into world coordinates before computing

its determinant, i.e. area A . For the “weighted” lattice basis $B^w = B^r \cdot B$ the area of the basis cell is

$$A = |\det B^w| = |\det B^r| \cdot |\det B| = \frac{x \cdot y}{n} \Rightarrow l = \sqrt{\frac{2 \cdot x \cdot y}{n \cdot \sqrt{3}}}$$

in analogy to Equation (5).

Since we perform the rasterization directly in the sheared basis, the shortest vectors lie within an ellipse (see Figure 10). Its axes $\mathbf{a}_x = ((n \cdot l)/x, 0)^t$ and $\mathbf{a}_y = (0, (n \cdot l)/y)^t$ result from transforming the circle axes $((n \cdot l), 0)^t$ and $(0, (n \cdot l))^t$ into the sheared basis B^r of the actual region.

As the rasterization runs in less than $\mathcal{O}(\|\mathbf{a}_x\| + \|\mathbf{a}_y\|)$, with $\|\mathbf{a}_x\|, \|\mathbf{a}_y\| \in \mathcal{O}(\sqrt{n})$, we still have a run-time complexity of $\mathcal{O}(\sqrt{n})$. Finally the Gaussian basis reduction needs to be adapted to weighted norms in order to compute the minimum distance. For that purpose the only modification consists in weighting the initial basis before performing the reduction steps. Therefore the search algorithm maintains a run-time complexity of $\mathcal{O}(\sqrt{n} \log n)$.

Anisotropic Rank-1 Lattices

Using the algorithm from Section 3 with weighted norms only requires to transform the desired main direction $\mathbf{d} \in \mathbb{R}^2$ into the sheared basis B^r of the desired domain.

5 Applications in Computer Graphics

The search algorithms from Section 2.1 allow one to find suitable generator vectors for the graphics applications introduced in [3, 4] much faster. Here, we introduce two new applications of anisotropic rank-1 lattices.

5.1 Anti-Aliasing by Anisotropic Rank-1 Lattices

In graphics applications rank-1 lattices can be used to integrate the image function over the pixels. By adapting the quadrature rule to the Fourier spectrum of the image function in a way that more of the important frequencies are captured, aliasing artifacts can be reduced. The improved anti-aliasing is illustrated by a comparison in Figure 11.

Given the algorithm from Section 3, an anisotropic MMD rank-1 lattice is specified by the main direction \mathbf{d} and the width w . We globally assume maximum anisotropy by fixing $w = 0$. The main direction \mathbf{d} is determined by projecting the normal of the first object intersected by a ray through the center of a pixel onto the image plane and normalizing the resulting vector. This way the samples from the anisotropic rank-1 lattice in the pixel become

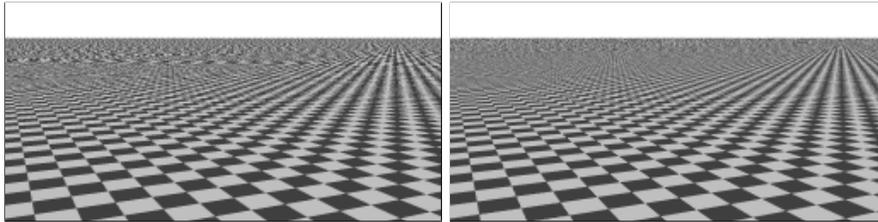


Fig. 11. An infinite checker board rendered with 16 samples for each pixel. The left image uses the same MMD rank-1 lattice $L_{16,(1,4)}$ for all pixels, while in the right image an anisotropic MMD rank-1 lattice adapted to the spectrum of each pixel is used. Clearly some aliases under the horizon become much more attenuated.

isotropic and more uniform, when projected onto the surface seen in the scene (see Figure 12 on the left). As a consequence the texture is averaged more efficiently, resulting in reduced aliasing. Note that for this argument, we assumed only one plane perpendicular to the normal seen through a pixel, which is a useful approximation in many cases.

As the perspective projection does not have an impact on the variance of the checker board until a certain distance from the camera, anisotropic rank-1 lattices are used only for those pixels for which the hit point of a ray through a pixel midpoint and the checker board exceeds a certain distance to the camera (which is determined experimentally for this special setting). Otherwise MMD rank-1 lattices are used per pixel.

In Figure 12 on the right, we compared the anisotropic rank-1 lattices to MMD rank-1 lattices and jittered grid by computing the L_2 -norm of a converged reference image to the corresponding test images for an increasing number of sampling points per pixel. Note that both axes in the error graph are scaled logarithmically and that the reference image was computed by applying a jittered grid sampling pattern with 1024×1024 samples at each pixel. We observe that using the anisotropic rank-1 lattice outperforms the other sampling patterns especially for lower sampling rates. In contrast to the MMD rank-1 lattices, the error curve of the anisotropic lattices does not expose a strong oscillation any more.

5.2 Rank-1 Lattice Images and Textures

In [4] the Voronoi diagram of MMD rank-1 lattices was used as an approximation to hexagonal pixel layout. While the visual quality at the same number of pixels was superior to classic rectangular layouts, the algorithms were simpler than for hexagonal layouts.

If now an image, or more specifically a texture, exposes an anisotropy, anisotropic MMD rank-1 lattices can be used to further improve the visual appearance, i.e. the approximation power. This is illustrated in Figure 13 for a wood grain texture, which exposes one main direction with large variance.

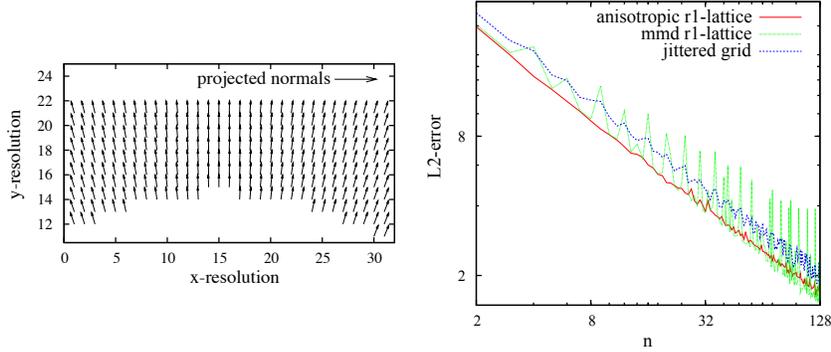


Fig. 12. Left: The arrows indicate the pixels and directions for which anisotropic rank-1 lattices are used. Right: Comparison of the anisotropic rank-1 lattices, to MMD rank-1 lattices and jittered grid.

The parameters for determining the anisotropic MMD rank-1 lattice are computed from the structure tensor of each pixel. Without loss of generality let $\lambda_{1,i} > \lambda_{2,i}$ be the eigenvalues of the structure tensor and $\mathbf{v}_{1,i}$ and $\mathbf{v}_{2,i}$ the corresponding eigenvectors for each pixel $i \in [0, xRes \cdot yRes)$. Then the main direction \mathbf{d} is computed by averaging the eigenvector of the largest eigenvalue over all pixels. The width

$$w = 1.0 - \frac{1}{A_{max}} \cdot \sum_{i=0}^{xRes \cdot yRes} \frac{\lambda_{1,i}}{\lambda_{2,i}}$$

subtracts the normalized texture anisotropy from 1, since 0 means maximum anisotropy for the search algorithm from Section 3. The normalization constant A_{max} must be determined experimentally for a set of textures.

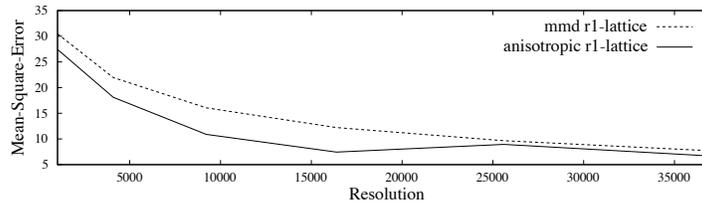


Fig. 14. Error graph showing the different approximation qualities measured with respect to a reference image.

In Figure 14 isotropic rank-1 lattice textures are compared to anisotropic ones by means of the L_2 -error of the test images to a reference solution for

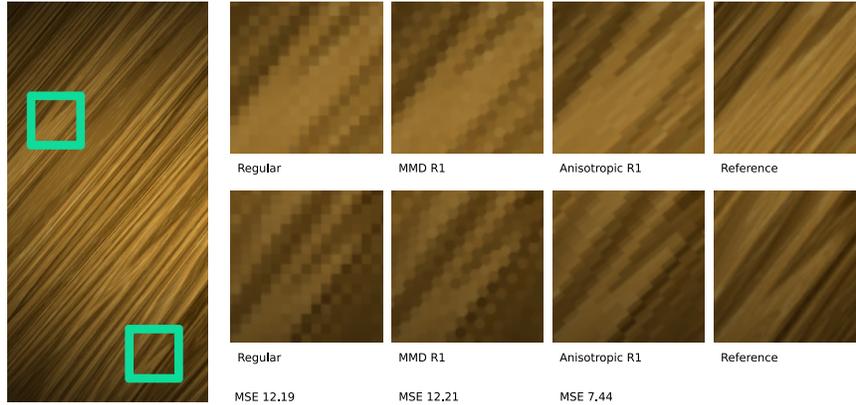


Fig. 13. Magnifications of the highlighted squares in the texture on the left represented on the regular grid, MMD rank-1 lattice, and anisotropic rank-1 lattice by 16384 pixels each. Note that for the anisotropic rank-1 lattice the mean square error (MSE) to the high resolution reference on the right is about half of the regular and MMD rank-1 lattice.

an increasing number of lattice points for the source image of Figure 13. As can be seen from the error graph, the anisotropic rank-1 lattice textures are superior, as they are able to capture even small details, which are lost in the isotropic case.

6 Conclusions

We introduced algorithms that efficiently search for generator vectors of rank-1 lattices and sequences with important new applications in computer graphics. Useful results were obtained for both image synthesis and representation. Future research will concentrate on applications of rank-1 lattice sequences and the fast search of generator vectors for the anisotropic case.

Acknowledgement. The authors would like to thank mental images GmbH for support and funding of this research.

References

1. Conway, J., Sloane, N., Bannai, E.: Sphere-packings, Lattices, and Groups. Springer-Verlag New York, Inc. (1987)
2. Cools, R., Reztsov, A.: Different quality indexes for lattice rules. *Journal of Complexity* **13**(2), 235–258 (1997)
3. Dammertz, H., Keller, A., Dammertz, S.: Simulation on rank-1 lattices. In: A. Keller, S. Heinrich, H. Niederreiter (eds.) *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pp. 205–216. Springer (2008)

4. Dammertz, S., Keller, A.: Image synthesis by rank-1 lattices. In: A. Keller, S. Heinrich, H. Niederreiter (eds.) *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pp. 217–236. Springer (2008)
5. Dieter, U.: How to Calculate Shortest Vectors in a Lattice. *Math. Comp.* **29**(131), 827–833 (1975)
6. Entezari, A., Dyer, R., Möller, T.: From sphere packing to the theory of optimal lattice sampling. In: T. Möller, B. Hamann, R. Russell (eds.) *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*. Springer (2009)
7. Fincke, U., Pohst, M.: Improved Methods for Calculating Vectors of Short Length in a Lattice, Including a Complexity Analysis. *Math. Comp.* **44**, 463–471 (1985)
8. Foley, J., van Dam, A., Feiner, S., Hughes, J.: *Computer Graphics, Principles and Practice*, 2nd Edition in C. Addison-Wesley (1996)
9. Hickernell, F., Hong, H.: Computing multivariate normal probabilities using rank-1 lattice sequences. In: G. Golub, S. Lui, F. Luk, R. Plemmons (eds.) *Proceedings of the Workshop on Scientific Computing (Hong Kong)*, pp. 209–215. Springer (1997)
10. Hickernell, F., Hong, H., L'Ecuyer, P., Lemieux, C.: Extensible lattice sequences for quasi-Monte Carlo quadrature. *SIAM J. Sci. Comput.* **22**, 1117–1138 (2001)
11. Kannan, R.: *Algorithmic Geometry of Numbers*. *Annual Reviews in Computer Science* **2**, 231–267 (1987)
12. Knuth, D.: *The Art of Computer Programming Vol. 2: Seminumerical Algorithms*. Addison Wesley (1981)
13. L'Ecuyer, P.: Tables of Linear Congruential Generators of Different Sizes and Good Lattice Structure. *Math. Comput.* **68**(225), 249–260 (1999)
14. L'Ecuyer, P., Couture, R.: An Implementation of the Lattice and Spectral Tests for Multiple Recursive Linear Random Number Generators. *INFORMS Journal on Computing* **9**(2), 206–217 (1997)
15. L'Ecuyer, P., Lemieux, C.: Variance Reduction via Lattice Rules. *Manage. Sci.* **46**(9), 1214–1235 (2000)
16. Martinet, J.: *Perfect Lattices in Euclidean Spaces*. Springer-Verlag (2003)
17. Niederreiter, H.: *Random Number Generation and Quasi-Monte Carlo Methods*. SIAM, Philadelphia (1992)
18. Rote, G.: Finding a Shortest Vector in a Two-Dimensional Lattice Modulo m . *Theoretical Computer Science* **172**(1–2), 303–308 (1997)
19. Siegel, C.: *Lectures on Geometry of Numbers*. Springer-Verlag (1989)
20. Sloan, I., Joe, S.: *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford (1994)