LATTICES AND SPHERE PACKING

GARRETT MITCHENER

Contents

1. Introduction

The purpose of this paper is to discuss some results in the areas of sphere packing and lattices which I've learned about over the past year or so. Along the way, we will explore spaces of lattices, some algorithms, and a numerical experiment. For the moment, we will keep the discussion fairly intuitive and get into some more rigorous definitions and analysis in the next section.

Date: May 4, 1999.

1.1. **What is a Lattice?** A lattice is a collection of regularly spaced points in some real vector space. See Figure 1 for some examples of lattices the plane. Each one consists of an array of evenly spaced points that fill the whole plane. We can pick one lattice point to be the origin, or starting point, and carefully choose two other points called *basis vectors*. The basis vectors determine a box known as a *fundamental parallelepiped*, and the lattice can be used to tile the plane with copies of it. From there, we can assign to each lattice a set of integer coordinates which count how many boxes from the origin along each axis the point lies. See Figure 2.

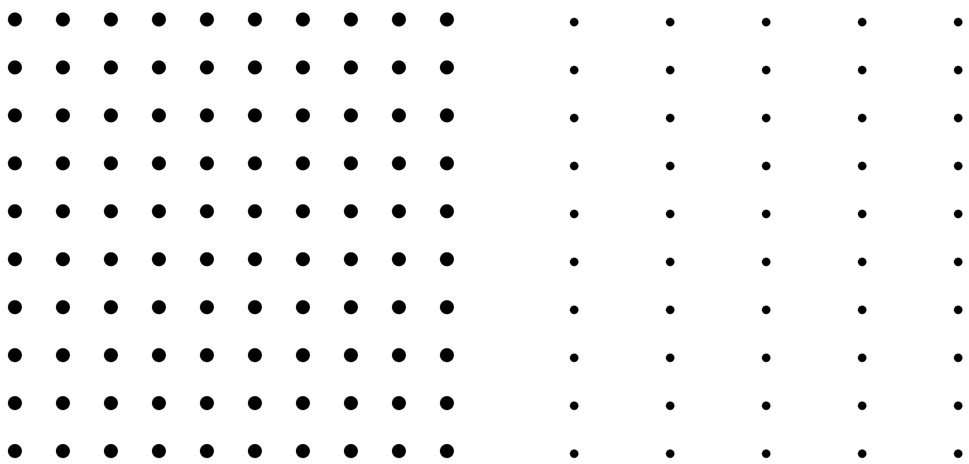

FIGURE 1. Two example lattices.

Note that we can choose the basis vectors in any number of ways, and doing so yields a different box and coordinate system, but the lattice is the same. See Figure 3.

1.2. **The Sphere Packing Problem.** The sphere packing problem is fairly easy to state: Given a real vector space $\mathbf{R}^n$, what is the most efficient way to fill it up with non-overlapping, n-dimensional spheres, all the same size? *Efficient* in this case means how do we arrange them so as to waste the smallest fraction of space possible. Obviously, a packing by large spheres wastes more actual space than a geometrically similar packing with smaller spheres, but both packings fill the same fraction of space. So, we will take into account the size of the spheres in the formal definition of efficiency.

For simplicity's sake, we can restrict our attention to packings in which each sphere is centered on a point in a lattice. Each lattice has a shortest non-zero vector. Since there must be a sphere at the origin

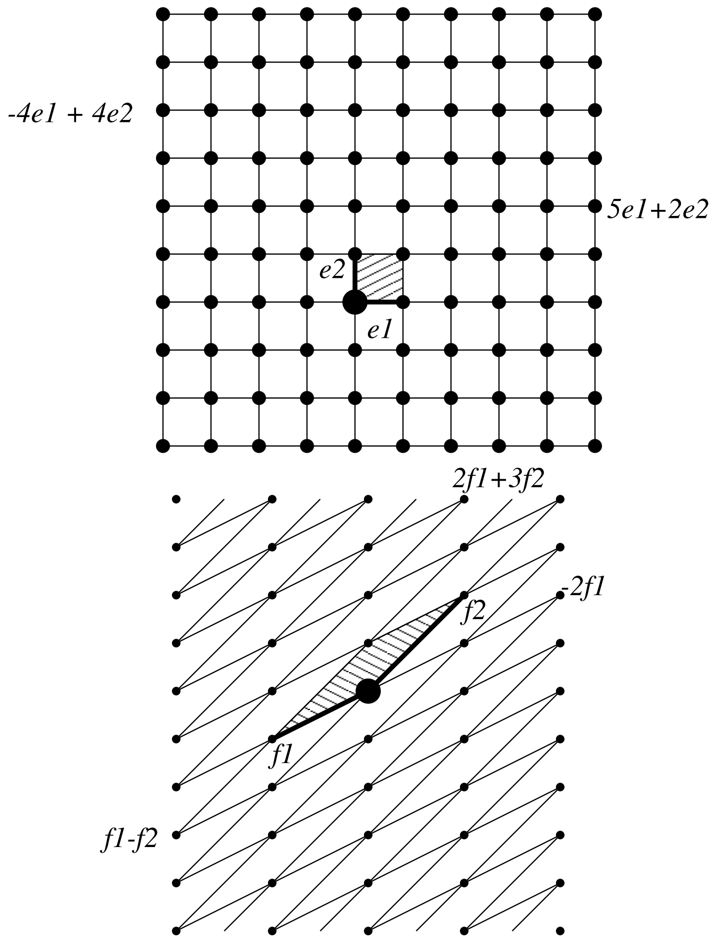

FIGURE 2. The same two example lattices with bases, fundamental parallelepipeds, and some coordinates marked in.

and at the end of this shortest vector, the largest possible radius of the spheres is half the length of the shortest vector. This is why shortest vectors will be very important over the course of this discussion. The efficiency of a lattice packing is measured numerically by the density δ which is the n-dimensional size of its spheres divided by the size of its fundamental parallelepiped.

For the general problem of packing circles in the plane on a lattice or not, the answer is well known; we place them so as to form a regular pattern with each circle surrounded by six others. See Figure 4.

However, in three dimensions and most others, the answer to the general problem is not completely known. The face-centered cubic packing, also known as D_3 and the "cannonball packing," is known to

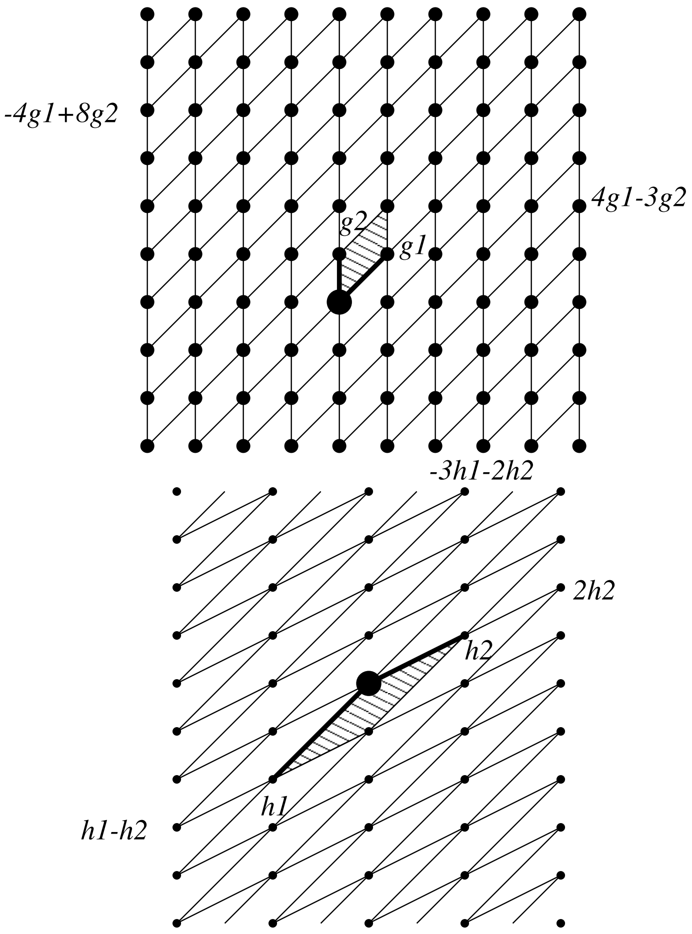

FIGURE 3. Same two lattices, new bases.

be the most efficient among lattice packings, but there is no known proof that it is the most efficient among non-lattice packings.[1] If we add the additional requirement that the spheres must be placed on a lattice, then the best lattice packings are known in a number of low dimensions. See Figure 5.

In particular, the lattices E_8 and Λ_{24} are known to be the best possible packings, lattice or non-lattice, in eight and twenty-four dimensions respectively. In higher dimensions (1000 or more), there are some known efficient packings. However, Minkowski proved a theorem[2] based on an averaging argument which asserts the existence of much

[1]Thomas Hales claims to have a proof that this is true, but as of April, 1999, it is still being reviewed for correctness.

[2]There seems to be some discrepancy over who actually proved this theorem. Conway and Sloane state in [2] that Minkowski proved it in 1905. However, there

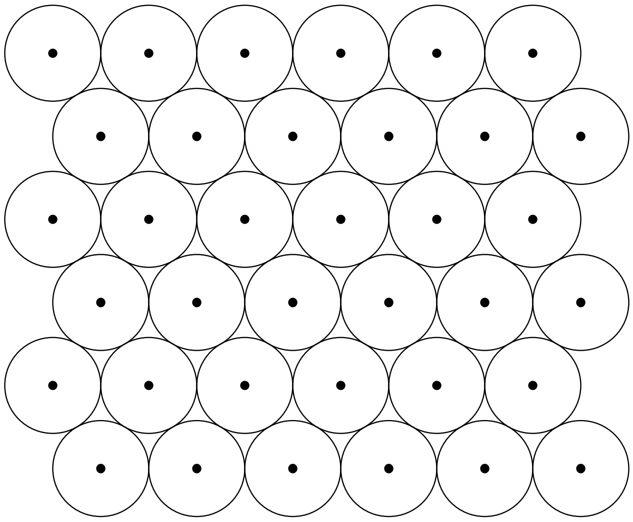

FIGURE 4. The best circle packing in the plane

Dim.	Name	Center Density
1	$\Lambda_1 = A_1 = \mathbf{Z}$	1
2	$\Lambda_2 = A_2$	$\frac{\sqrt{3}}{2}$
3	$\Lambda_3 = A_3 = D_3$	$\frac{\sqrt{2}}{4}$
4	$\Lambda_4 = D_4$	$\frac{1}{8}$
5	$\Lambda_5 = D_5$	$\frac{\sqrt{2}}{8}$
6	$\Lambda_6 = E_6$	$\frac{\sqrt{3}}{8}$
7	$\Lambda_7 = E_7$	$\frac{1}{16}$
8	$\Lambda_8 = E_8$	$\frac{1}{16}$
24	Λ_{24}	1

FIGURE 5. Table of best possible lattice packings in some special cases. The center density in the third column is the density divided by the volume of the spheres in the packing.

more efficient lattice packings than any known. The *density* of a sphere packing is defined to be the fraction of space covered by spheres, and

are remarks in Siegel [5] which suggest Minkowski first posed the conjecture, but Hlawka first proved it.

Minkowski's theorem asserts that the density of the densest possible lattice packing is at least

$$\frac{\zeta(n)}{2^{n-1}}$$

where $\zeta(n)$ is the Riemann ζ function.

This appears to be a rather weak bound for large n. Since $\zeta(n)$ is almost equal to 1 for large n, Minkowsi's bound is going to zero at an exponential rate. However, an example will show that it is still an important result. Consider the square lattice $\mathbf{Z}^n$. In the plane and three dimensions, it is a reasonably good way to pack spheres, although not the best. In general, $\mathbf{Z}^n$ tiles space with boxes of unit volume, each of which contains pieces of spheres which add up to one whole sphere. Its density is therefore just the volume of an n-dimensional sphere with radius $\frac{1}{2}$ divided by 1, which is given by

$$\left(\frac{1}{2^n}\right)\left(\frac{\pi^{n/2}}{(n/2)!}\right).$$

for even n and by

$$\frac{\pi^{(n-1)/2}\left(\frac{1}{2}(n-1)\right)!}{n!}$$

for odd n. In either case, this density is going to zero at a factorial rate, which is much faster than the exponential in Minkowski's bound. This means that for large n, Minkowski's bound, as small as it is, is huge compared to the density of the square lattice.

1.3. **What are we Trying to Do?** The overall problem is, given some number of dimensions, find the most efficient sphere packing lattice. This is a hard problem that no one has solved yet, so we will instead focus on why it is such a difficult problem, and eventually look for ways to at least find *good* lattice packings, if not the best one.

Minkowski's bound on efficiency is an obvious place to start. Its proof requires an understanding of the space of lattices and integration over that space. It is also closely related to the theory of reduction of lattices and quadratic forms. (See [3].) There are numerous ways to define a reduced basis for a lattice, that is, one that satisfies some extremal property. Reduction allows us to deal with fundamental domains of lattice spaces. In such a region, almost all lattices have exactly one representative basis, which is useful when we need to make sure that no lattices are counted twice.

All this suggests a numerical experiment: Write a computer program to create lattices at random, and eventually, one might turn up with high efficiency, and perhaps better than any previously known. The trouble is developing an algorithm which picks a random lattice. There are many ways to pick a basis at random and we must somehow decide which is the right way. The first constraint is that the algorithm should produce bases in some fundamental region, to ensure that it can construct every possible lattice. The second constraint is that it should obey the proper probability distribution. It turns out that satisfying either one of those is difficult, and we will have to make some approximations along the way. Then we can get down to number crunching.

2. Spaces of Lattices in $\mathbf{R}^n$

Suppose V is an n-dimensional real vector space with a positive definite inner product. Denote the corresponding orthogonal group by $\mathrm{O}(V)$. A lattice Λ is a subgroup of V generated by taking all integer linear combinations of a basis $v_1, v_2, \ldots v_n$ of V:

$$\Lambda = \{h_1 v_1 + h_2 v_2 + \cdots + h_n v_n : h_i \in \mathbf{Z}\}$$

The important thing to remember is that the lattice is the subgroup, and although we define it here in terms of a basis, it has many bases, none of which is obviously preferable to the others.

Two lattices Λ_1 and Λ_2 are *similar,* which I will denote

$$\Lambda_1 \sim \Lambda_2, \tag{1}$$

if there is a scaling factor $a \in \mathbf{R}$, $a > 0$, and an isometry $Q \in \mathrm{O}(V)$ such that $\Lambda_1 = aQ\Lambda_2$. That is, two lattices are similar if one is a scaled, rotated, and possibly reflected version of the other. Two lattices are *congruent,* which I will denote

$$\Lambda_1 \cong \Lambda_2 \tag{2}$$

if they are similar and the scaling factor $a = 1$.

A *framed lattice* is a lattice together with an ordered basis. Two framed lattices $(\Lambda_1, \{u_1, u_2, \ldots u_n\})$ and $(\Lambda_2, \{v_1, v_2, \ldots v_n\})$ are *similar* if the lattices themselves are similar with $\Lambda_1 = aQ\Lambda_2$, and the bases also satisfy $u_i = aQv_i$ for $i = 1, 2, \ldots, n$. Note that the lattices may in fact be the identical, but have different, non-similar framings. Two framed lattices are *congruent* if they are similar with $a = 1$.

The Gram-Schmidt algorithm can be used to find an orthonormal basis $\{q_1, q_2, \ldots q_n\}$ for V. Then the mapping

$$\begin{aligned} \mathbf{R}^n &\to V \\ x &\mapsto \sum x_i q_i \end{aligned}$$

is an isomorphism between V and $\mathbf{R}^n$, and the inner product on V corresponds to the standard inner product on $\mathbf{R}^n$. So, there is no loss of generality if consider only lattices in $\mathbf{R}^n$.

2.1. **The Space of Framed Lattices as a Quotient Space.** Each framed lattice in $\mathbf{R}^n$ must have a basis consisting of n linearly independent vectors $a_1, a_2, \ldots a_n$. Every vector in the lattice can be written as a sum of integer multiples of these basis vectors. The *basis matrix* M is an n-by-n matrix whose rows consist of the coordinates of the a_i's with respect to the standard basis:

$$M = \begin{pmatrix} \cdots a_1 \cdots \\ \cdots a_2 \cdots \\ \vdots \\ \cdots a_n \cdots \end{pmatrix}$$

Since the a_i's are linearly independent, M is non-singular and therefore an element of $\mathrm{GL}_n(\mathbf{R})$. Furthermore, each possible M determines the basis of a lattice. Thus we can identify $\mathrm{GL}_n(\mathbf{R})$ with the space of all framed lattices and may denote a framed lattice by (Λ, M).

Each point in the lattice has two canonical sets of coordinates, one with respect to the standard basis, and a second one, called *lattice coordinates*, with respect to the lattice basis. The lattice coordinates are all integers.

The *fundamental parallelepiped* of a framed lattice is the polytope whose vertices are all vectors in Λ whose lattice coordinates are all 0 or 1. Copies of the fundamental parallelepiped made by translating each point in it by some lattice vector will fit neatly into the lattice with each vertex falling on a lattice point.[3] The entire vector space can be filled with translates of it. Its signed volume is given by $\det M$. The sign indicates whether the basis is positively or negatively oriented. This comes from the order of basis vectors in the matrix and can be toggled by multiplying the first row by -1.

Points of the space

$$\mathcal{F}_n = \mathrm{GL}_n(\mathbf{R}) / \mathrm{O}(n) \tag{3}$$

correspond to congruence classes framed lattices. Points of the related space

$$\mathcal{UF}_n = \mathrm{SL}_n(\mathbf{R}) / \mathrm{SO}(n) \tag{4}$$

[3] It is in fact the fundamental region of the translation action of the lattice on $\mathbf{R}^n$, but will be called the fundamental parallelepiped to avoid confusion with other fundamental regions that will be discussed later.

correspond to congruence classes of framed lattices all of determinant 1. These are called *unimodular* framed lattices (hence the "u") because of their unit determinants. Note that $\mathfrak{U}\mathfrak{F}_n$ is the space of all framed lattices modulo similarity under $\sim$ because each framed lattice can be scaled to have determinant 1. However, it will often be more convenient to work with $\mathfrak{F}_n$.

Incidentally, $\mathfrak{U}\mathfrak{F}_n$ excludes framed lattices with determinant -1, even though they define a parallelepiped with unit volume. This is not unreasonably restrictive since such framed lattice can be turned into an element of $\mathfrak{U}\mathfrak{F}_n$ by negating the first basis vector, which does not change the lattice.

Note that the volumes of the fundamental parallelepipeds of congruent framed lattices are the same: Suppose $(\Lambda_1, M_1) \cong (\Lambda_2, M_2)$ where $M_1 = QM_2$ for some $Q \in \mathrm{O}(n)$. Then the volume of the fundamental parallelepiped of (Λ_1, M_1) is $|\det M_1| = |\det Q|\,|\det M_2| = |\det M_2|$ which is the volume of the fundamental parallelepiped of (Λ_2, M_2).

We would also like to identify different framings which determine the same lattice points. Suppose (Λ, M) is a framed lattice. Every vector $x \in \Lambda$ can be written as

$$x = hM$$

where $h \in \mathbf{Z}^n$ is a row vector of integers. Now suppose $H \in \mathrm{GL}_n(\mathbf{Z})$ is a non-singular integer matrix and let

$$N = H^{-1}M$$

The general lattice vector x can be re-written as

$$x = hH^{-1}M$$

which expresses x as an integral linear combination of the rows of N, which can be thought of as a sequence of vectors $b_1, b_2, \ldots b_n$. Therefore, the vectors $b_1, b_2, \ldots b_n$ form a basis for the same lattice. Likewise, given a basis $b_1, b_2, \ldots b_n$, a non-singular integer matrix H can be constructed from the lattice coordinates of the b_i's. We therefore define two framed lattices to be *equivalent*, which I will denote

$$(\Lambda_1, M_1) \equiv (\Lambda_2, M_2) \tag{5}$$

if they are congruent and $M_1 = HM_2$ for some $H \in \mathrm{GL}_n(\mathbf{Z})$. A point in the quotient space

$$\mathfrak{L}_n = \mathrm{GL}_n(\mathbf{Z}) \backslash \mathrm{GL}_n(\mathbf{R}) / \mathrm{O}(n) \tag{6}$$

is an equivalence class of framed lattices. Likewise, a point in the quotient space

$$\mathcal{UL}_n = \mathrm{SL}_n(\mathbf{Z})\backslash\mathrm{SL}_n(\mathbf{R})/\mathrm{SO}(n) \tag{7}$$

is an equivalence class of unimodular framed lattices. In both spaces the equivalence class of the identity corresponds to the square lattice $\mathbf{Z}^n \subset \mathbf{R}^n$. In short, $\mathrm{GL}_n(\mathbf{Z})$ acts by transforming one basis of a lattice into a basis for the same lattice.

The volumes of fundamental parallelepipeds of a lattice are all the same. To see this, suppose that we have two equivalent framings of the same lattice $(\Lambda, M_1) \equiv (\Lambda, M_2)$, where $M_1 = HM_2$ for some $H \in \mathrm{GL}_n(\mathbf{Z})$. Since H is an integer matrix, its determinant is an integer. However, H^{-1} is also an integer matrix with integer determinant, and $\det HH^{-1} = 1 = \det H \det H^{-1}$, so we must have $\det H = \pm 1$. The volume of the fundamental parallelepiped of (Λ, M_1) is therefore $|\det M_1| = |\det H|\,|\det M_2| = |\det M_2|$, which is the volume of the fundamental parallelepiped of (Λ, M_2).

Later, we will discuss other ways to think about the space of framed lattices and the $\mathrm{GL}_n(\mathbf{Z})$ action on it.

2.2. The Space of Framed Lattices as Positive Definite Matrices.

The space of n-by-n positive definite symmetric matrices is denoted $\mathcal{P}_n$. Consider the map

$$\begin{aligned} \phi : \mathrm{GL}_n(\mathbf{R})/\mathrm{O}(n) &\to \mathcal{P}_n \\ \phi : [M] &\mapsto MM^T \end{aligned}$$

where $[M] = \{MQ : Q \in \mathrm{O}(n)\}$ denotes the orbit of M.

Proposition 1. *The map ϕ is a bijection.*

Proof. First, ϕ is well defined because if $Q \in \mathrm{O}(n)$, then $MQ(MQ)^T = MQQ^TM^T = MM^T$, so M and MQ correspond to the same positive definite symmetric matrix. Next, note that ϕ is surjective because every positive definite matrix $A \in \mathcal{P}_n$ can be factored as $A = MM^T$ for some $M \in \mathrm{GL}_n(\mathbf{R})$ using the Cholesky factorization. Finally, ϕ is injective since $\phi([M]) = \phi([N])$ means $MM^T = NN^T$ which implies

$$N^{-1}M = N^T\left(M^T\right)^{-1}$$

and after some manipulation

$$\left(N^{-1}M\right)^T = \left(N^{-1}M\right)^{-1}.$$

This proves that $N^{-1}M$ is orthogonal, so $[M] = [N]$.

$\square$

Thus, $\mathcal{P}_n$ is equivalent to the space of framed lattices $\mathcal{F}_n$. For a framed lattice (Λ, M) the matrix MM^T is called its *Gram Matrix*. The determinant 1 subset of $\mathcal{P}_n$ is denoted $\mathcal{S}_n$ and is likewise equivalent to the space of unimodular framed lattices $\mathcal{UF}_n$.

A more intuitive way to think of the equivalence between positive definite matrices and framed lattices is to use the fact that each $A \in \mathcal{P}_n$ can be used to define a positive definite inner product on $\mathbf{R}^n$:

$$\langle x, y \rangle_A = xAy^T, \quad x, y \in \mathbf{R}^n.$$

There is a fundamental result of linear algebra called the Principal Axis Theorem which states that every symmetric matrix has a full set of eigenvectors. Moreover, all its eigenspaces are mutually orthogonal. Therefore every symmetric A can be diagonalized $A = QDQ^T$ by an orthogonal change of coordinates $Q \in \mathrm{SO}(n)$. Finally, A is positive definite if and only if all its eigenvalues are positive, which is the only case we are interested in at the moment.

The most important intuitive consequence of the this theorem is that the set of vectors satisfying $|x|_A \leq r$ for $r > 0$ is an ellipsoid. To see this, notice that the A-norm of a vector can be re-written as

$$|x|_A^2 = \langle x, x \rangle_A = x^T Ax = xQ\sqrt{D}\sqrt{D}Q^T x^T = \left| xQ\sqrt{D} \right|^2$$

In short, the A-inner product in the original vector space is equivalent to the standard inner product on a rotated and stretched version of the same space. The Q term is a rigid rotation of the vector space, and the $\sqrt{D}$ term applies stretching to the coordinates of x with respect to the rotated axes. In the stretched and rotated space, the set of vectors one unit away from the origin is a sphere because we are using the standard inner product. If we imagine stretching and turning it back, the unit sphere gets distorted, and we end up with an ellipsoid in the original vector space.

Suppose (Λ, M) is a framed lattice, and let $A = MM^T$ be the Gram matrix of the framed lattice. Given $x \in \Lambda$, let $h \in \mathbf{Z}^n$ be its lattice coordinates, $x = hM$. Its Euclidean length is given by

$$|x|^2 = hMM^T h^T = |h|_A^2 .$$

In short, when dealing with lattices and lengths of vectors, we can either use the Euclidean inner product and the spatial coordinates of lattice vectors, or use a stretched and rotated inner product and work with lattice coordinates, which is the same as working with the square lattice $\mathbf{Z}^n$.

2.2.1. *Measure and Integration on $\mathcal{P}_n$.* By considering the entries on and above the diagonal of a symmetric matrix as independent variables, we can think of $\mathcal{P}_n$ as a subset of $\mathbf{R}^{n(n+1)/2}$, on which there is an obvious basic measure, given by

$$dY = \prod_{1 \leq i \leq j \leq n} dy_{i,j} \tag{8}$$

We are really interested in integration on $\mathcal{L}_n$, which is necessary for Minkowski's theorem and to define what it means to pick a matrix at random. It turns out that the best way to integrate over a region in $\mathcal{L}_n$ is to think of it as a set of regions in $\mathcal{F}_n = \mathcal{P}_n$ which differ from each other by transformation by an element of $\mathrm{GL}_n(\mathbf{Z})$, then pick one of them and integrate over it. The integral should be the same no matter which equivalent region we pick, which means we must integrate with respect to a measure which is invariant under the action of $\mathrm{GL}_n(\mathbf{Z})$.

The action of $\mathrm{GL}_n(\mathbf{Z})$ on $\mathcal{F}_n$ is discussed in Section 2.1 but we need the equivalent action on $\mathcal{P}_n$ which comes from the interpretation of $A \in \mathcal{P}_n$ as the Gram matrix of some framed lattice (Λ, M) such that matrix $A = MM^T$. A matrix $H \in \mathrm{GL}_n(\mathbf{Z})$ acts on basis matrices by left multiplication, turning them into differently-shaped bases for the same lattice, $H : M \mapsto HM$. The equivalent action on the Gram matrix is of course $H : A \mapsto HAH^T$ which will be denoted $H * A$ (with the H on the left because it is a left action).

The action is not restricted to elements of $\mathrm{GL}_n(\mathbf{Z})$. In fact, $\mathrm{GL}_n(\mathbf{R})$ acts on $\mathcal{P}_n$ in the same way. This action has an associated invariant measure:

Proposition 2. *The measure*

$$d\mu(Y) = \frac{dY}{(\det Y)^{(n+1)/2}} \tag{9}$$

on $\mathcal{P}_n$ is invariant under the action of $\mathrm{GL}_n(\mathbf{R})$.

Proof. We must show that $d\mu(M * Y) = d\mu(Y)$ for all $M \in \mathrm{GL}_n(\mathbf{R})$. First, consider a diagonal matrix $B \in \mathrm{GL}_n(\mathbf{R})$ with diagonal entries b_i. The entries of $W = B * Y = BYB$ are given by $w_{i,j} = b_i y_{i,j} b_j$.

$$d\mu(W) = (\det W)^{-(n+1)/2} \prod_{1 \leq i \leq j \leq n} d(b_i y_{i,j} b_j).$$

Note that in expanding that differential product, we get one factor of b_k for each $y_{i,j}$ where $i = k$ and $j > k$, one factor for $j = k$ and $i < k$, and two for $i = j = k$ for a total of $n+1$ factors of b_k. Thus, after rearranging some terms, we get

$$d\mu(W) = \frac{b_1^{n+1} b_2^{n+1} \cdots b_n^{n+1}}{(\det W)^{(n+1)/2}} dY.$$

Since $\det W = (\det B)^2 \det Y$ and $\det B = b_1 b_2 \cdots b_n$, all that boils down to

$$d\mu(W) = (\det Y)^{-(n+1)/2} dY = d\mu(Y).$$

For a more general matrix M, we must compute the change of measure with the Jacobian determinant

$$J(M) = \left| \det \frac{dM}{dY} \right|$$

$$\begin{aligned} d\mu(M * Y) &= \frac{J(M)}{(\det M * Y)^{(n+1)/2}} dY \\ &= \frac{J(M)}{(\det M)^{n+1}} \frac{dY}{(\det Y)^{(n+1)/2}} \end{aligned}$$

For the diagonal matrix B, what we just proved by plugging in the change of coordinates directly is that $J(B) = (\det B)^{n+1}$, which is exactly what it needs to be to preserve the measure.

Now consider the action of a diagonalizable matrix $M = UBU^{-1}$. By the chain rule, the Jacobian of a composition of actions is a product of two Jacobians, so we get

$$J(M) = J\left(UBU^{-1}\right) = J(U)J(B)J(U)^{-1} = J(B).$$

From the case of a diagonal matrix, $J(B) = (\det B)^{n+1}$ but since $\det B = \det M$, we have $J(M) = (\det M)^{n+1}$, so M preserves the measure.

In the case of matrices that are not diagonalizable, we can write them as $M = UBU^{-1}$ where B is a Jordan block matrix and add $k\varepsilon$ to the k-th diagonal entry of B. The resulting product M_ε is diagonalizable because it has distinct eigenvalues, so it falls into the previous case. Therefore $J\left(M_\varepsilon\right) = \left(\det M_\varepsilon\right)^{n+1}$. Since the Jacobian is continuous, we can let ε go to zero, and it follows that $J(M) = (\det M)^{n+1}$.

□

This measure makes it possible to integrate over $\mathcal{P}_n$ in a $\mathrm{GL}_n(\mathbf{R})$-equivariant manner, and therefore over $\mathcal{L}_n$.

3. Fundamental Domains of $\mathcal{P}_n$

As was noted in Section 2.2.1, $\mathrm{SL}_n(\mathbf{Z})$ acts on $\mathcal{P}_n$. That action can be used to split up $\mathcal{P}_n$ into *fundamental domains* $\mathbf{F} \subset \mathcal{P}_n$ which satisfy the following:

- $\mathbf{F}$ is closed and connected.
- The space $\mathcal{P}_n$ is tiled by $\mathbf{F}$, meaning

$$\mathcal{P}_n = \bigcup_{A \in \mathrm{GL}_n(\mathbf{Z})} A * \mathbf{F}. \tag{10}$$

- If Y and $A * Y$ are in $\mathbf{F}$ for some $A \in \mathrm{GL}_n(\mathbf{R}) - \{\pm I\}$, then Y must be on the boundary of $\mathbf{F}$.

3.1. Grenier's Fundamental Domain.

We will work with a particular fundamental domain documented in [7] that lends itself to random lattice generation. Every $Y \in \mathcal{P}_n$ can be factored as

$$Y = \begin{pmatrix} 1 & 0 \\ x^T & I \end{pmatrix} \begin{pmatrix} v & 0 \\ 0 & W \end{pmatrix} \begin{pmatrix} 1 & x \\ 0 & I \end{pmatrix} \tag{11}$$

where $x \in \mathbf{R}^{n-1}$, $v \in \mathbf{R}$, $v > 0$, $W \in \mathcal{P}_{n-1}$. This decomposition can be formed from Y adding a multiple of the first row to each of the other rows so as to make the left entry 0, then adding a multiple of the first column to each of the other columns to make the top entry 0. Note that v turns out to be just the upper-left entry of Y. Consider $H \in \mathrm{GL}_n(\mathbf{Z})$ with entries

$$H = \begin{pmatrix} a & b \\ c^T & D \end{pmatrix}$$

where $a \in \mathbf{Z}$, $b, c \in \mathbf{Z}^{n-1}$, and D is an $n-1$ by $n-1$ integer matrix. Computing $\tilde{Y} = H * Y$ yields the following:

$$\begin{aligned} H * Y &= \left(\begin{pmatrix} a & b \\ c^T & D \end{pmatrix} \begin{pmatrix} 1 & 0 \\ x^T & I \end{pmatrix} \right) * \begin{pmatrix} v & 0 \\ 0 & W \end{pmatrix} \\ &= \begin{pmatrix} a + \langle b, x \rangle & b \\ c^T + D x^T & D \end{pmatrix} * \begin{pmatrix} v & 0 \\ 0 & W \end{pmatrix} \\ &= \begin{pmatrix} v \left(a + \langle b, x \rangle \right)^2 + b W b^T & \cdots \\ \vdots & \ddots \end{pmatrix} \end{aligned}$$

The upper-left entry is the important one for defining the fundamental domain. If $\tilde{Y}$ is factored in terms of $\tilde{v}$, $\tilde{x}$, and $\tilde{W}$ analogously to Equation 11, the preceding computation of $\tilde{Y}$ proves that $\tilde{v} = v \left(a + \langle b, x \rangle \right)^2 + b W b^T$.

Intuitively, we want each element of the fundamental domain to have the smallest possible upper-left entry. Thus we define Grenier's fundamental domain $\mathbf{F}_n$ to be the set of $Y \in \mathcal{P}_n$ with factorization as in Equation 11 which satisfy the following:

- $v \leq v\left(a + \langle b, x\rangle\right)^2 + bWb^T$ for all $a \in \mathbf{Z}$ and all $b \in \mathbf{Z}^{n-1} - \{0\}$ such that $\begin{pmatrix} a & b \end{pmatrix}$ is the top row of some element of $\mathrm{GL}_n(\mathbf{Z})$.
- $0 \leq x_1 \leq \frac{1}{2}$ and $|x_j| \leq \frac{1}{2}$ for $j = 2 \ldots n-1$.
- $W \in \mathbf{F}_{n-1}$.

The proof that this is indeed a fundamental domain is beyond the scope of this discussion. (See [7].) However, we will need the following lemma to develop an algorithm for generating lattices at random:

Proposition 3. *Given $Y \in \mathbf{F}_n$ and its factorization as in Equation 11, the diagonal entries $w_{j,j}$ of W satisfy*

$$w_{j,j} \geq \frac{3v}{4}.$$

Proof. Let b be e_j, the j-th standard basis vector, and let $a = 0$. Then from the definition of $\mathbf{F}_n$,

$$v \leq v\left(\langle e_j, x\rangle\right)^2 + e_j W e_j^T.$$

After some manipulation,

$$v \leq \frac{w_{j,j}}{1 - |x_j|^2}.$$

Since $|x_j| \leq \frac{1}{2}$, the denominator is no smaller than $\frac{3}{4}$, and the result follows.

$\square$

4. Reduction Theory

Reduction is a process which transforms one basis for a lattice into another basis which has certain extremal properties. Reduction algorithms are used for several things, such as assigning a unique, canonical basis to almost every lattice, and determining the length of the shortest vectors in a lattice. Unfortunately, reduction is a very time consuming process except in some special cases, and quicker approximate reduction algorithms must often be used instead.

4.1. **The Shortest Vector Problem.** The important attribute of a lattice for sphere packing purposes is its *density.* Imagine that identical spheres of radius r are placed such that their centers lie on lattice points and their interiors do not overlap. Suppose M is a basis matrix for the lattice and consider its fundamental parallelepiped. The fraction of space that the spheres fill can be computed by taking the fraction of the fundamental parallelepiped covered by parts of spheres centered at its corners. That fraction is given by

$$\frac{\sigma_n r^n}{|\det M|}$$

where σ_n is the volume of a unit sphere in $\mathbf{R}^n$. Recall that all framings of a lattice produce fundamental parallelepipeds with the same volume, so we may frame the lattice with any basis matrix M and get the same number. We may therefore define the *density* to be the following function of a lattice:

$$\delta(\Lambda) = \frac{\sigma_n r^n}{|\det M|} \tag{12}$$

Similar lattices have the same center density because rotation and reflection do not affect the size of the fundamental parallelepiped or the spheres, and the scaling term scales both by the same amount.

The best possible packing on a given lattice can be determined from the length of the shortest vector as follows. Suppose Λ is a lattice and there is a sphere of radius r centered on every lattice point. Somewhere, there is a shortest non-zero lattice vector m, that is, a point closest to the origin. Since there must be spheres at the origin and at the other end of m, r can be no more than $\frac{1}{2}|m|$. If r is exactly $\frac{1}{2}|m|$, any pair of spheres overlap at at most one point on their surfaces, which means the spheres are properly packed. This choice of r maximizes the center density of the packing, which is why we care about finding m.

The trouble is that finding that shortest vector given some basis for Λ turns out to be extremely difficult computationally in all but a few special cases. Suppose $a_1, a_2, \ldots a_n$ is a basis for Λ. Every vector $v \in \Lambda$ can be written

$$v = \begin{pmatrix} h_1 & h_2 & \ldots & h_n \end{pmatrix} \begin{pmatrix} \cdots a_1 \cdots \\ \cdots a_2 \cdots \\ \vdots \\ \cdots a_n \cdots \end{pmatrix}.$$

Its norm is given by

$$|v|^2 = vv^T = \begin{pmatrix} h_1 & h_2 & \ldots & h_n \end{pmatrix} \begin{pmatrix} \cdots a_1 \cdots \\ \cdots a_2 \cdots \\ \vdots \\ \cdots a_n \cdots \end{pmatrix} \begin{pmatrix} \vdots & \vdots & & \vdots \\ a_1 & a_2 & \cdots & a_n \\ \vdots & \vdots & & \vdots \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_n \end{pmatrix}$$

The problem of finding the shortest vector in the lattice is equivalent to choosing integers h_i such that $|v|^2$ is minimal. This is a kind of integer programming problem, and there is no known efficient algorithm for solving it in general. (See [1].)

4.2. **Reduction in Two Dimensions.** Reduction in two dimensions is easy to state: Given a framed lattice, find a basis which includes one of the shortest vectors in the lattice. The two-dimensional reduction problem turns out to be quite simple, unlike higher dimensional cases. The algorithm can best be understood by placing it in the upper half plane model of hyperbolic space.

4.2.1. *Reduction in the Upper Half Plane.* The upper half plane is defined to be

$$\mathbf{H} = \{z \in \mathbf{C} : \operatorname{Im} z > 0\} \tag{13}$$

Each two-dimensional framed lattice can be represented by a single point in the upper half plane as follows. Given a framed lattice

$$(\Lambda, \{a_1, a_2\}),$$

we may interpret the basis vectors as complex numbers. Dividing a_1 and a_2 by a_1 is just a rotation followed by scaling, so every such framed lattice is similar to some $(\Lambda', \{1, \tau\})$. If $\operatorname{Im} \tau < 0$, we can again divide the basis by τ or replace τ by $-\tau$. The final result is that every two-dimensional framed lattice is similar to a framed lattice of the form $(\Lambda, \{1, \tau\})$ where $\operatorname{Im} \tau > 0$. Consequently, the ordered basis $\{1, \tau\}$ is always positively oriented.

As discussed in Section 2.1, $\mathrm{GL}_n(\mathbf{Z})$ acts on the space of framed lattices. Since each $\tau \in \mathbf{H}$ defines a positively oriented framed lattice, we can ignore the orientation-reversing elements of $\mathrm{GL}_2(\mathbf{Z})$ and consider the action of $\mathrm{SL}_2(\mathbf{Z})$ on $\mathbf{H}$:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} z = \frac{az + b}{cz + d}. \tag{14}$$

This is a well-known action with fundamental domains as shown in Figure 6. The principal fundamental domain $\mathbf{F}$, which is shaded in the figure, is the key to finding the shortest vector.

Given the framed lattice (Λ, τ), the basis $\{1, \tau + n\}$ is just an integer change of coordinates applied to the old basis, which preserves the lattice. This type of coordinate change, denoted T^n where

$$T(\tau) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \tau = \tau + 1$$

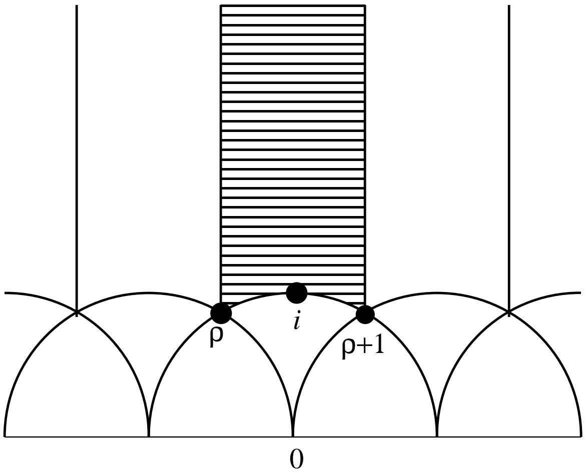

FIGURE 6. Fundamental Domains of $\mathrm{SL}_2(\mathbf{Z})$ acting on $\mathbf{H}$. The shaded region is the principal fundamental domain $\mathbf{F}$. The point $\rho = \frac{1+i\sqrt{3}}{2}$ satisfies $\rho^3 = 1$.

moves τ horizontally from one fundamental domain to the next. Similarly, $\{1, -\frac{1}{\tau}\}$ is a rotation and rescaling and yields a similar framed lattice. This type of coordinate change, denoted

$$S(\tau) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \tau = -\frac{1}{\tau}$$

swaps the two basis vectors and negates one so as to maintain the positive orientation of the basis. It exchanges the inside of the unit semicircle with the outside.

A combination of T and S can be used to find the shortest vector in a lattice. The idea is to transform the basis until 1 is known to be the shortest vector in the lattice. Suppose we have the basis $\{1, \tau\}$ and $|\tau| < 1$. In this case, we can apply S to swap the two vectors and attempt to shorten the other vector. If $|\tau| > 1$, we can repeatedly apply T or T^{-1} until $\mathrm{Re}\,\tau$ is between $-\frac{1}{2}$ and $\frac{1}{2}$. If this new τ satisfies $|\tau| < 1$, we apply S and continue. If not, τ is in the principal fundamental domain $\mathbf{F}$, and no further improvements are possible. The reduction is complete, and we have a framed lattice similar to the one we started with for which 1 is the shortest vector.

4.3. **The LLL Reduction Algorithm.** Lenstra, Lenstra, and Lovász developed an approximate reduction algorithm that works efficiently in any number of dimensions. (See [1].) Let $(\Lambda, \{b_1, b_2, \ldots b_n\})$ be a framed lattice, let $b^*{}_1, b^*{}_2, \ldots b^*{}_n$ be the corresponding orthogonal basis of $\mathbf{R}^n$ (but not of Λ) generated by the Gram-Schmidt algorithm, and let $\mu_{i,j} = \left\langle b_i, b_j^* \right\rangle / \left|b_j^*\right|^2$. The basis $b_1, b_2, \ldots b_n$ is said to be *LLL-reduced* if it satisfies

$$|\mu_{i,j}| \leq \frac{1}{2} \text{ for } 1 \leq j < i \leq n$$

and

$$|b_i^*|^2 \geq \left(c - \mu_{i,i-1}^2\right) |b_i^* - 1|^2 \text{ for } 1 \leq i \leq n$$

where c is a parameter between $\frac{1}{4}$ and 1. LLL-reduced bases satisfy a number of useful inequalities, one of which states that for all lattice vectors $x \neq 0$,

$$|b_1| \leq \alpha^{(n-1)/2} |x| \text{ where } \alpha = \frac{1}{c - \frac{1}{4}}.$$

In other words, b_1 is almost the shortest vector in the lattice, and in most cases actually is the shortest vector.

The corresponding LLL-reduction algorithm computes an LLL-reduced basis from any given basis. The parameter c, which must be between $\frac{1}{4}$ and 1, is normally set to $\frac{3}{4}$, so $\alpha = 2$. If c is increased but less than 1, the algorithm is likely to produce an even shorter b_1. It will take longer to finish, though its run-time remains of polynomial order. If $c = 1$, the run-time is unknown.

The details of the algorithm, given in [1], are beyond the scope of this discussion and will not be reproduced here. However, the general idea is to work inductively as follows. The sequence $b_1, b_2, \ldots b_{k-1}$ is assumed to be LLL-reduced and we attempt to add in b_k. One step of two-dimensional reduction is performed on the sub-lattice spanned by b_{k-1} and b_k, which yields new values for these two vectors. After that step, if $b_1, b_2, \ldots b_k$ is LLL-reduced, we continue building with the next basis vector. Otherwise, we exchange b_{k-1} and b_k and back up to where we know $b_1, b_2, \ldots b_{k-2}$ is LLL-reduced and try to add in b_{k-1}, and so on. This process of advancing and backing up does terminate in a reasonable amount of time, and the resulting basis is LLL-reduced.

5. A Numerical Experiment

We now have everything necessary to at least describe an algorithm for generating lattices at random:

- It will have to generate bases since there is really no other way for a computer to represent a lattice. We can either generate the bases directly or generate positive definite matrices and a apply the Cholesky factorization to obtain a basis.
- Since we are interested mostly in the density of the lattice, it is sufficient to generate elements of $\mathfrak{UL}_n$.

- It should be able to generate every possible lattice in $\mathcal{UL}_n$, which means it should generate bases representing elements of a fundamental domain of the action of $\mathrm{SL}_n(\mathbf{Z})$ on $\mathcal{UF}_n$. Since this will likely be computationally difficult, we will settle for a superset of the fundamental domain.
- It should obey the measure given in Section 2.2.1. This means that if it generates elements of some region $\Omega \subset \mathcal{P}_n$, the resulting probability density should be the normalized invariant measure

$$\frac{d\mu(Y)}{\int_\Omega d\mu(Y)}. \tag{15}$$

Once we have a random basis generator, we can run it, use the LLL reduction algorithm to find what we hope is the shortest vector in the resulting lattice, and compute the density. With luck it will beat Minkowski's bound.

5.1. **Grenier Generators.** Grenier's fundamental domain $\mathbf{F}_n$ for $\mathcal{P}_n$, described in Section 3.1, is well-suited for generating random bases inductively. Assuming $A_k \in \mathbf{F}_k$, we can extend it to an element $A_{k+1} \in \mathbf{F}_{k+1}$ by generating x_k and v_k:

$$A_{k+1} = \begin{pmatrix} 1 & 0 \\ x_k^T & I \end{pmatrix} \begin{pmatrix} v_k & 0 \\ 0 & A_k \end{pmatrix} \begin{pmatrix} 1 & x_k \\ 0 & I \end{pmatrix} \tag{16}$$

or:

$$A_{k+1} = \begin{pmatrix} v_k & v_k x_k \\ v_k x_k^T & A_k \end{pmatrix}. \tag{17}$$

From the first form, Equation 16, we may calculate that $\det A_{k+1} = v_k \det A_k$ because the left and right matrices have 1's down the diagonal and the determinant of the matrix in the middle can by found by cofactors. By induction, we have

$$\det A_n = v_1 v_2 \cdots v_n. \tag{18}$$

Expanding A_n recursively yields:

$$\begin{pmatrix} v_n & v_n x_{n,1} & \cdots & \cdots & v_n x_{n,n-1} \\ v_n x_{n,1} & \ddots & \vdots & \vdots & \vdots \\ \vdots & \cdots & v_3 & v_3 x_{3,1} & v_3 x_{3,2} \\ \vdots & \cdots & v_3 x_{3,1} & v_2 & v_2 x_{2,1} \\ v_n x_{n,n-1} & \cdots & v_3 x_{3,2} & v_2 x_{2,1} & v_1 \end{pmatrix} \tag{19}$$

from which we may derive that the plain measure dY at A_n is:

$$dY(A_n) = v_n^n v_{n-1}^{n-1} \cdots v_2^2 v_1 \prod_{1 \le k \le n} dv_k \prod_{1 \le j < k \le n} dx_{k,j} \tag{20}$$

which will be abbreviated

$$dY(A_n) = \left(\prod_{1 \le k \le n} v_k^k \right) dV dX. \tag{21}$$

The $\mathrm{GL}_n(\mathbf{R})$-invariant measure is:

$$\begin{aligned} d\mu(A_n) &= \frac{dY(A_n)}{(\det A_n)^{(n+1)/2}} \\ &= \left(\prod_{1 \le k \le n} v_k^{k-(n+1)/2} \right) dV dX. \end{aligned} \tag{22}$$

To generate $x_{k,1}$, we use a uniform distribution on $[0, \frac{1}{2}]$. To generate $x_{k,j}$ where $j = 2 \dots n-1$, we use a uniform distribution on $[-\frac{1}{2}, \frac{1}{2}]$. The uniform distribution is selected because there are no x's in the density term of Equation 22, and the ranges come from the definition of Grenier's fundamental domain. For the moment, assume that v_k is generated in $[0, \infty)$ according to some unspecified, not necessarily independent, probability density function f_k.

A serious difficulty in constructing a random basis generator arises at this point: If A_k is to be in $\mathbf{F}_k$, v_k cannot be selected independently because it must satisfy inequalities that depend on entries in A_{k-1}. The following sections describe some compromises that work around this problem and result in working but not ideal random basis generators.

5.2. **A Power Distribution.** It is possible to choose the probability density functions f_k so that the resulting random matrices are distributed according to the density in Equation 22, but the v_k are chosen independently and should not be expected to satisfy the inequalities defining $\mathbf{F}_n$. Consequently, this algorithm generates matrices in a crude approximation of the fundamental domain that is not even a superset of it.

Since the v_k's are selected independently, the probability density associated with a particular A_n is

$$2^{n-1} \prod_{1 \le k \le n} f_k(v_k). \tag{23}$$

The 2^{n-1} comes from the fact that the $x_{i,i+1}$ terms in Equation 19 are uniform over $[0, \frac{1}{2}]$ and have density 2. Since we want this to equal

Equation 22 up to a normalizing constant, it makes sense to set

$$f_k(v) = Cv^{k-(n+1)/2}.$$

Unfortunately, the integral of f_k over all of $[0,\infty)$ is infinite and no choice of C can normalize it to a proper probability density function.

If $k-(n+1)/2 < -1$, we may pick some $a > 0$ and require $v \geq a$. Then the cumulative density function is:

$$F_k(v) = C\int_a^v t^{k-(n+1)/2}dt.$$

By setting

$$C = -\frac{a^{k+1-(n+1)/2}}{k+1-(n+1)/2}$$

we can get $F_k(v) \to 1$ as $v \to \infty$, which would make f_k a proper density function.

If $k-(n+1)/2 = -1$, which can only happen if n is odd, we must require $0 < a \leq v \leq b < \infty$. Then the cumulative density function is:

$$F_k(v) = C\int_a^v \frac{dt}{t}$$

and $C = \log b - \log a$ forces $F_k(v) \to 1$ as $v \to \infty$.

If $k-(n+1)/2 > -1$, we must bound v from above by $v \leq b < \infty$, and the cumulative density function is:

$$F_k(v) = C\int_0^v \frac{dt}{t}$$

The constant should be set to

$$C = \frac{b^{k+1-(n+1)/2}}{k+1-(n+1)/2}$$

to make $F_k(v) \to 1$ as $v \to \infty$.

In each of these cases, an explicit closed form for F_k can be found by simple integration. This is good because it allows us to use the simple inverse CDF method to generate random numbers according to these distributions. The tricky part is picking values of a and b for each F_k, and there is no obvious way to do that.

5.3. **Better Approximations to $\mathbf{F}_n$.** The general random basis generator described in Section 5.2 generates matrices in a set S which turns out to be a very rough approximation of Grenier's fundamental domain $\mathbf{F}_n$. There is an adjustment to the random generator that trims pieces from S that are definitely not in $\mathbf{F}_n$ at the cost of altering the distribution.

Recall Proposition 3, which in the notation of this section asserts that

$$v_k \leq \frac{4v_j}{3} \text{ for all } j < k.$$

When generating v_k, we may therefore restrict it to

$$0 < v_k \leq \frac{4}{3} \min \{v_1, v_2, \dots v_{k-1}\}.$$

This restriction cuts pieces off S that are definitely not in $\mathbf{F}_n$, however, the probability density is now much more complex because v_k depends on $v_1, v_2, \dots v_{k-1}$.

6. Concluding Remarks

It is definitely feasible to implement the random matrix algorithms described in Section 5. In fact, a prototype has been written in the Python programming language and can be translated to C++ for large-scale calculations once all the implementation details are worked out. The difficulty is picking appropriate bounds for the random number generators as described in Section 5.1. Alternatively, there may be a nice way to generate only unimodular lattices, that is, we pick the v_k's and set the last one so that their product is 1.

It is unclear at this point how successful such a random matrix generator would be at finding good lattices. The only trials I have been able to run were in low dimensions, 27 at the most, with generators that were flawed in that they did not obey the distribution in Equation 22. The program was able to examine thousands of such matrices a day, a number which will increase dramatically once it has been re-written in C++. At that point numerical experiments in much larger dimensions will become feasible.

Appendix A. Notation and Conventions

Here is a list of notation and notational conventions used in this paper:

- All vectors x are assumed to be row vectors unless otherwise noted. Normally, w, x and other letters near the end of the alphabet represent vectors with real entries, while h and letters near the middle represent integer vectors (lattice coordinates).
- x^T represents a column vector.
- $\det A$ is the determinant of the matrix A.
- $\mathrm{GL}_n(\mathbf{R})$ is the group of all non-singular n-by-n matrices with entries in $\mathbf{R}$. They are used to represent general bases or framed lattices, with the rows interpreted as basis vectors. Basis matrices are normally denoted by letters B and M.
- $\mathrm{SL}_n(\mathbf{R})$ is the subgroup of $\mathrm{GL}_n(\mathbf{R})$ consisting of all matrices of determinant 1. These represent unimodular bases.
- $\mathrm{O}(n)$ is the subgroup of $\mathrm{GL}_n(\mathbf{R})$ consisting of orthogonal matrices. $Q \in \mathrm{O}(n)$ if $QQ^T = I$. This is the group of all rotations and reflections in $\mathbf{R}^n$.
- $\mathrm{SO}(n)$ is the subgroup of $\mathrm{O}(n)$ consisting of matrices of determinant 1. These are all the rotations.
- $\mathrm{GL}_n(\mathbf{Z})$ is the group of all non-singular n-by-n matrices with entries in $\mathbf{Z}$. They all have determinant ± 1.
- $\mathrm{SL}_n(\mathbf{Z})$ is the subgroup of $\mathrm{GL}_n(\mathbf{Z})$ consisting of all matrices with determinant 1. These matrices are often called *unimodular transformations.* They represent a change of basis for a lattice.
- $\mathcal{P}_n$ is the subset of $\mathrm{GL}_n(\mathbf{R})$ consisting of positive definite symmetric matrices. $A \in \mathcal{P}_n$ if $A = A^T$ and the inner product $\langle \cdot, \cdot \rangle_A$ is positive definite. Given a basis matrix M, its corresponding Gram matrix is given by $Y = MM^T \in \mathcal{P}_n$.
- $\mathcal{S}_n$ is the subset of $\mathcal{P}_n$ consisting of matrices with determinant 1.
- $\mathcal{F}_n$ is the set of all framed lattices, that is, lattices together with a basis.
- $\mathcal{UF}_n$ is the set of all unimodular (determinant 1) framed lattices.
- $\mathcal{L}_n$ is the set whose elements are sets of equivalent framed lattices.
- $\mathcal{UL}_n$ is the set whose elements are sets of equivalent unimodular framed lattices, meaning all framings have determinant 1.

- $\langle x, y \rangle = xy^T$ is the standard inner product. I have adopted this notation for inner products to avoid the confusing notation (x, y) which already means at least two other things.
- $\langle x, y \rangle_A = xAy^T$ is an inner product based on the symmetric matrix A. It is positive definite if $\langle x, x \rangle_A > 0$ for $x \neq 0$.
- $|x|$ is the length of x. For an inner product based on A, $|x|_A = \sqrt{\langle x, x \rangle_A}$.
- $M * Y = MYM^T$ where $M \in \mathrm{GL}_n(\mathbf{R})$ and $Y \in \mathcal{P}_n$ is a Gram matrix. There is a common alternative notation $Y[M] = M^T Y M$ that is fine as long as M and Y are represented by single letters, but gets extremely confusing when writing matrices with entries or doing calculations by hand. I have decided not to use it here.

References

[1] Henri Cohen. *A Course in Computational Algebraic Number Theory.* Springer-Verlag, New York, 1993. (The first edition of this book has some serious errors. In particular, it omits some steps from the LLL reduction algorithm. Corrections may be found in later editions or at the author's web site `http://www.math.u-bordeaux.fr/~cohen/`.).

[2] J. H. Conway and N. J. Sloane. *Sphere Packings, Lattices, and Groups.* Springer, New York, third edition, 1999.

[3] John H. Conway. *The Sensual (Quadratic) Form.* The Mathematical Association of America, 1997.

[4] Sheldon Ross. *A First Course in Probability.* Prentice Hall, Upper Saddle River, New Jersey, 1998.

[5] Carl Ludwig Siegel. *Lectures on the Geometry of Numbers.* Springer-Verlag, New York, 1989.

[6] Gilbert Strang. *Linear Algebra and its Applications.* Harcourt Brace Jovanovich College Publishers, New York, 1986.

[7] Audrey Terras. *Harmonic Analysis on Symmetric Spaces and Applications*, volume 2. Springer-Verlag, New York, 1985.

[8] Audrey Terras. *Harmonic Analysis on Symmetric Spaces and Applications*, volume 1. Springer-Verlag, New York, 1985.