Lecture 5: Lattices

Algorithmic Number Theory (Fall 2014)
Rutgers University
Swastik Kopparty
Scribe: Danny Scheinerman

1 Introduction to Lattices

Definition 1. *A* lattice *is a set $L \subset \mathbb{R}^n$ that is a discrete, additive group.*

For example, $\mathbb{Z}^n$ is a lattice. Note that $S = \{a + b\sqrt{2} \ : \ a, b \in \mathbb{Z}\}$ is a additive group, but is not a lattice because it is not discrete. One can find nonzero elements of S arbitrarily close to 0.

Definition 2. *If $L \subset \mathbb{R}^n$ is a lattice and $dim_{\mathbb{R}}(L) = n$ we say L is a* full rank lattice*.*

Theorem 3. *Every full rank lattice $L \subseteq \mathbb{R}^n$ is of the form $M \cdot \mathbb{Z}^n$ where M is a full rank $n \times n$ matrix.*

Proof. We first show that L is finitely generated as an abelian group.

Let $v_1, \ldots, v_n \in L$ be such that their $\mathbb{R}$-span equals $\mathbb{R}^n$. Observe that there is a constant C such that every $x \in \mathbb{R}^n$ is C-close to the integer span of $\{v_1, \ldots, v_n\}$. To see this, write $x = \sum \alpha_i v_i$ where each $\alpha_i \in \mathbb{R}$; then x is C close to $\sum_i \lceil \alpha_i \rfloor v_i$ for some C depending only on $v_1, \ldots, v_n$.

Suppose we have an infinite sequence $v_1, v_2, \ldots, \in L$ such that each v_i is not in the integer span of $v_1, \ldots, v_{i-1}$. Then for each $i > n$, v_i is C-close to some element w_i in the integer span of $v_1, \ldots, v_n$. Then $v_i - w_i \in B_C$, the ball of radius C around 0. We also have $v_i - w_i \in L$. Finally, we notice that $v_i - w_i$ are all distinct. Thus the collection of points $v_i - w_i$, which all lie in L, have a limit point. This contradicts the discreteness of L. Thus such an infinite sequence does not exist, and so L is a finitely generated group.

Now we show that L is contained in the $\mathbb{Q}$-linear span of $v_1, \ldots, v_n$.

Let $w \in L$, so we can uniquely write $w = \sum_{i=1}^n c_i v_i$. If the c_i are all rational, we are done. Otherwise we will contradict discreteness. Let $\epsilon > 0$. By an application of Dirichlet's pigeonhole principle, there exists an integer q such that qc_i is within ϵ of an integer (this is called a simultaneous diophantine approximation). Thus we have that $qw = u + \sum_{i=1}^n \delta_i v_i$ where $|\delta_i| < \epsilon$ and $u \in L$. But $qw - u \in L$, and $\|qw - u\| = \|\sum_{i=1}^n \delta_i v_i\| \leq \sum_{i=1}^n \|\delta_i v_i\| \leq \epsilon \sum \|v_i\|$. Since, this holds for arbitrary ϵ, we see that L has vectors arbitrarily close to 0. This contradicts discreteness.

So after a change of basis, we have that L is a finitely generated subgroup of $\mathbb{Q}^n$. Let $v_1, \ldots, v_m$ ($m > n$) be a set of vectors whose integer span equals L. By the Hermite Normal Form theorem from last class, there is a set of n vectors whose integer span equals L. $\square$

To represent a lattice, we can then give a basis $b_1, \ldots, b_n \in \mathbb{R}^n$. Then $L = \{\sum_i a_i b_i \ : \ a_i \in \mathbb{Z}\}$, i.e. is the integer linear span of this set of vectors. For computational problems we will often assume that the $b_i \in \mathbb{Q}^n$ or even in $\mathbb{Z}^n$.

There are two fundamental hard problems in the theory of lattices. They are

1. The Shortest Vector Problem (SVP): Given a lattice L (represented by basis vectors $b_1, \ldots, b_n$) find a nonzero vector of shortest length. Note that this vector need not be unique. If $L = \mathbb{Z}^n$ then $\pm e_i$ where e_i is the standard basis vector has shortest nonzero length.

2. The Closest Vector Problem (CVP): Given a lattice L and a vector y, find $x \in L$ such that $\|x - y\|$ is minimized.

Both of these problems are known to be NP-hard. These problems are *not* NP-hard if the dimension is fixed, however. To be precise: let n be the dimension that the lattice lives in and if every coordinate in the presented basis of L has absolute value $\leq A$, then the input size is $\leq n^2 \log(A)$. The following are known

- There is no algorithm to solve SVP or CVP in time $\text{poly}(n^2 \log(A))$.
- There exists an algorithm in time $2^{\text{poly}(n)} \cdot \text{poly}(\log(A))$.

2 Gauss' Algorithm for SVP in 2 Dimensions

It is worth noting that the SVP is already interesting in two dimensions. For example, let L be the lattice given by the integer column space of $M = \begin{bmatrix} 39129 & 26790 \\ 69680 & 47707 \end{bmatrix}$. It is perhaps not obvsious that the columns of M span the same lattice as the columns of $M' = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix}$. Below is Gauss' algorithm:

1. Start with $u, v \in \mathbb{Z}^2$. Assume (or swap to make true) that $\|u\| \leq \|v\|$.

2. Let $m = \left\lfloor \frac{\langle v,u \rangle}{\|u\|^2} \right\rceil$. Note that $\dfrac{\langle v, u\rangle}{\|u\|^2} u$ is the projection of u onto v. So this is an integer $m \in \mathbb{Z}$ such that $\|v - mu\|$ is minimized.

3. Set $v = v - mu$

4. If $\|v\| \leq \|u\|$ then swap u and v and goto step 2. Otherwise terminate.

There are two things to show. One that when the algorithm stops, that u is the shortest vector, and two that the algorithm is efficient. The second concern will be illustrated in homework exercises. Below we show that upon termination, u is the shortest vector.

Observe that we have $\|u\| \leq \|v\|$ and $\left| \dfrac{\langle v, u\rangle}{\|u\|^2} \right| \leq \dfrac{1}{2}$ $(*)$. We want to show that for all $a, b \in \mathbb{Z}$ not both zero that $\|au + bv\|^2 \geq \|u\|^2$. We can expand the left hand side as

$$\begin{aligned} \|au + bv\|^2 &= a^2\|u\|^2 + b^2\|v\|^2 + 2ab\langle u, v\rangle \\ &\geq a^2\|u\|^2 + b^2\|v\|^2 - \|ab\|\|u\|^2 \quad \text{(using equation *)} \\ &= \|u\|^2(a^2 + b^2 - |ab|) \end{aligned}$$

Now observe that if a or b is zero then $a^2 + b^2 \geq 1$ and $ab = 0$, so we obtain $\|au + bv\|^2 \geq \|u\|^2$. If both are nonzero, assume without loss of generality that $|a| \geq |b|$ and we have $a^2 \geq |ab|$ so $a^2 + b^2 - |ab| \geq b^2 \geq 1$, and again we have $\|au + bv\|^2 \geq \|u\|^2$. $\square$

Next, we have a variation of Gauss' algorithm that gives an "almost shortest vector" and can easily be seen to be efficient:

1. Start with $u, v \in \mathbb{Z}^2$. Assume (or swap to make true) that $\|u\| \leq \|v\|$.
2. Let $m = \left\lfloor \frac{\langle v,u \rangle}{\|u\|^2} \right\rceil$. Note that $\dfrac{\langle v, u\rangle}{\|u\|^2} u$ is the projection of u onto v. So this is an integer $m \in \mathbb{Z}$ such that $\|v - mu\|$ is minimized.
3. Set $v = v - mu$
4. If $\|v\| \leq 0.9\|u\|$ then swap u and v and goto step 2. Otherwise terminate.

The length of the shorter vector decreaes by a factor of at least 0.9 at each iteration and so the algorithm is fast. Upon termination we have $\|u\| \leq 1.1\|v\|$ and similar to above $\|au + bv\|^2 \geq \|u\|^2(a^2 + 0.9^2b^2 - |ab|)$ for all not both zero integers $a, b \in \mathbb{Z}$. Now in minimizing $a^2 + 0.9^2b^2 - |ab|$ we see that we may as well have $ab \geq 0$ so it suffices to consider $a, b \geq 0$. In this case, the expression becomes $a^2 + 0.9^2b^2 - ab$. If a or b is zero then we either obtain 1 or $0.9^2 = 0.81$. If both are positive then observe $a^2 + 0.9^2b^2 - ab = (a - 0.9b)^2 + 0.8ab \geq 0.8$. So we have shown $\|au + bv\|^2 \geq 0.8\|u\|^2$. So u is within a small factor of the shortest vector.