Chapter 5

Lattices

1 LATTICES

Let $v_1, \dots v_n$ be a basis for $\mathbb{R}^n$, all entries in the v_i are integers.
The Lattice, L, formed by the v_i are integer combinations of $v_1, \dots, v_n$
If $v \in L$, then $v = t_1 v_1 + \dots + t_n v_n = (t_1, \dots, t_n) \begin{pmatrix} v_1 \\ v_n \end{pmatrix}$

Let $A = \begin{pmatrix} v_1 \\ v_n \end{pmatrix}$, $v = (t_1, \dots t_n) A$

The collection of all $v = t_1 v_1 + \dots + t_n v_n$ with $0 < t_i < 1$ is called the fundamental domain, F, of L. If U is an integer matrix with integer inverse $\Leftrightarrow$ $\det U = \pm 1$. Then $UA = B$ has rows which are also a basis for L. If $(x_1, \dots, x_n) \in F$, then

$(x_1, \dots x_n) = t_1 v_1 + \dots + t_n v_n \quad 0 < t_i < 1$

The volume of $F = \int_F dx_1 \dots dx_n =$

$$\int_{n\text{ cube}} \det\left(\frac{dx_i}{dx_j}\right) dt_1 \dots dt_n = \int \det A \, dt_1 \dots dt_n = \det A$$

Hence the volume of $F = \det A$

If $UA = B$ and F' is the fundamental domain for B, then

$\text{vol } F' = \det B = \det U \det A = \det A = \text{vol } F$

So a change of basis does not change the volume of the fundamental domain. Clearly $\text{vol } F \leq |N_1| \cdots |N_n|$

Since F and F' have the same volume, the more orthogonal the basis vectors are in pairs, the shorter they are. So a check for how orthogonal they are is

$$\left(\frac{\det A}{|x_1| \cdots |x_n|} \right)^{1/2} \leq 1$$

If this is close to one, they are close to orthogonal, what is called a good basis. Such basis will be easier to work with. Otherwise, ~~they~~ the basis is called a bad basis

3 Closest vector

If $v_1, \ldots v_n$ are a basis for L and they are orthogonal, then when $v = t_1 v_1 + \ldots + t_n v_n$,

$$|v|^2 = t_1^2 |v_1|^2 + \ldots + t_n^2 |v_n|^2$$

If $w = k_1 v_1 + \ldots + k_n v_n$ is a vector in R^n (so the k_i may not be integer), $|v-w|^2 = (t_1-k_1)^2 |v_1|^2 + \ldots + (t_n-k_n)^2 |v_n|^2$

The closest vector in L to w is found by taking each t_i to be the integer that is closest to k_i: $v = \lfloor k_1 \rceil v_1 + \ldots + \lfloor k_n \rceil v_n$

The same is true if the v_i are close to being pairwise orthogonal, called a good basis

We have described Babai's algorithm for finding the closest vector. This does not work if the orthogonality condition is removed

There is a related shortest vector problem in a lattice

If $v_1, \ldots v_n$ is a basis made up of vectors that are mutually orthogonal. If $v \in L$, $v = c_1 v_1 + \ldots + c_n v_n$ and $\|v\| = c_1^2 |v_1| + \ldots + c_n^2 |v_n|$, $c_i \in \mathbb{Z}$. The shortest v is found by having $v = v_1$, v_1 the shortest vector in the basis. This simple solution fails if the vectors are not orthogonal. But the solution is close to being valid if the vectors are close to being orthogonal. Hence it is very useful if one can start with any basis and convert it to an almost orthogonal one. The important algorithm which does this is the LLL algorithm

THE GGH

In R^n, Alice picks a good integer vector basis. She checks using the Hadamard ratio. She constructs an integer matrix U with $\det U = \pm 1$. Let L be the lattice spanned by her basis, $v_1, \ldots, v_n$. Let V = matrix with rows $v_1, \ldots, v_n$ and $W = UV$. W is another basis for L and Alice checks that W is a bad basis using the Hadamard ratio. ~~Bob wants~~ Alice makes W public. Bob wants to send Alice the row vector m. He picks a small pertubation vector r and computes the ciphertext

$$e = mW + r$$

e is not in L because of r but it is close since r is small. Alice can compute m from e using her good basis and Babai's algorithm.

Ex. Alice picks $v_1 = (3,0)$ $v_2 = (0,2)$

Hadamard ratio $\left(\frac{\det\begin{pmatrix}3&0\\0&2\end{pmatrix}}{|v_1||v_2|}\right)^{1/2} = 1$

$V = \begin{pmatrix}3&0\\0&2\end{pmatrix}$. She picks $U = \begin{pmatrix}1&1\\2&3\end{pmatrix}$,

$\det U = 1$. She finds

$W = UV = \begin{pmatrix}3&2\\6&6\end{pmatrix} = \begin{pmatrix}w_1\\w_2\end{pmatrix}$ and this

is her public key

Bob wants to send $m = (3,5)$

He picks $r = (1,1)$ and finds

$e = mW + r = (40, 37)$

He sends e to Alice

Alice uses her good basis private

key $e = (40,37) = c_1(3,0) + c_2(0,2)$

$c_1 = \frac{40}{3}$ $c_2 = \frac{37}{2}$

Using Babai she get a vector

close to e that is in L

$e' = 13(3,0) + 18(0,2) = (13,18)\begin{pmatrix}v_1\\v_2\end{pmatrix}$

$= (13,18)\begin{pmatrix}3&-1\\-2&1\end{pmatrix}\begin{pmatrix}w_1\\w_2\end{pmatrix} = (3,5)\begin{pmatrix}w_1\\w_2\end{pmatrix}$

$m = (3,5)$

All Eve can use is the bad basis

$$(40, 37) = d_1 (3, 2) + d_2 (6, 6)$$

$$d_1 = 3 \qquad d_2 = 3/6$$

If she uses Babai she gets

$$(40, 37) = 3(3, 2) + 5(6, 6)$$

$$m = (3, 5)$$

She actually got the correct m

Lucky!

Note the Hadamard ratio for the bad basis:

$$\left(\frac{\det \begin{pmatrix} 3 & 2 \\ 6 & 6 \end{pmatrix}}{|w_1| \, |w_2|} \right)^{1/2} = \left(\frac{6}{\sqrt{13}\sqrt{72}} \right)^{1/2} = \left(\frac{1}{\sqrt{26}} \right)^{1/2} \approx \frac{1}{\sqrt{5}}$$

Ex. Alice picks good basis

$v_1 = (10, 0) \quad v_2 = (0, 10)$

and $U = \begin{pmatrix} 1 & 9 \\ 1 & 0 \end{pmatrix}$

$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = U \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} 10 & 90 \\ 10 & 100 \end{pmatrix}$

Alice makes public the bad basis

$w_1 = (10, 90) \quad w_2 = (10, 100)$

Bob's messese is $m = (3, 5)$

He picks $r = (1, 1)$ and computes

$e = (3, 5)\begin{pmatrix} 10 & 90 \\ 10 & 100 \end{pmatrix} + (1, 1) = (81, 771)$

Alice computes

$(81, 771) = e = n_1 (10, 0) + n_2 (0, 10)$

$n_1 = 8.1 \quad n_2 = 77.1$

Using Babel

$e = (8, 77)\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = (8, 77)\begin{pmatrix} 10 & -9 \\ -1 & 10 \end{pmatrix}\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$

$= (3, 5)\begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$

getting $m = (3, 5)$

9 GGH Signature Scheme

Alice picks a good basis $v_1, \ldots v_n$ and computes a bad basis as in the GGH cryptosystem, $w_1, \ldots, w_n$. She has a document d to send to Bob and she sends $w_1, \ldots, w_n$ to Bob as well. This is her verification key. She computes the closest vector, s, to d in L, ~~call it s~~, using Babai algorithm and the good basis. The coefficients of s with respect to the good basis is her signature. s is sent to Bob. Bob knows s and the bad basis. She writes s in terms of the bad basis and the coefficents are her signature.

$$s = d_1 w_1 + \ldots + d_n w_n \quad (d_1, \ldots d_n) = \text{signature}$$

The signature, document and bad basis go to Bob.

Bob uses the signature and bad basis

$$d_1 w_1 + \ldots + d_n w_n = s$$ ~~to~~ to get s

and checks $|s-d|$. If this is small, Alice signed it.

10 Ex. Alice picks $v_1 = (2,0)$ $v_2 = (0,1)$

and $U = \begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix}$. Then

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = W = U \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 2 & 1 \\ 4 & 3 \end{pmatrix}$$

Document $d = (5,4)$

Alice computed $d = c_1 v_1 + c_2 v_2$

$c_1 = 5/2$ $c_2 = 4$

Using Babai $s = 3(2,0) + 4(0,1) = (6,4)$

She computes $(6,4) = d_1(2,1) + d_2(4,3)$

$$\begin{aligned} 6 &= 2d_1 + 4d_2 \\ 4 &= d_1 + 3d_2 \end{aligned} \qquad d_1 = d_2 = 1$$

Signature $= (d_1, d_2) = (1,1)$

Bob uses (d_1, d_2) and the bad basis

$(2,1) + (4,3) = (6,4) = s$

And checks

$|s - d| = |(6,4) - (5,4)| = 1$ small

Alice sent it.

Ex Alice picks good basis $v_1 = (10,0)$, $v_2 = (0,10)$

11 and $U = \begin{pmatrix} 1 & 9 \\ 1 & 10 \end{pmatrix}$. Then $W = U \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} 10 & 90 \\ 10 & 100 \end{pmatrix}$

Alice's verification key is $\{w_1, w_2\}$

Alice has document $d = (5,7)$

She write's $d = b_1(10,0) + b_2(0,10)$

$b_1 = .5 \quad b_2 = .7$

She use's Babai to get $s \in L$ close to d $\quad s = \lceil b_1 \rceil v_1 + \lceil b_2 \rceil v_2 =$

$1(10,0) + 1(0,10) = (10,10)$

She writes s in terms of the bad basis

$s = (10,10) = b_1(10,90) + b_2(10,100)$

$b_1 = 9 \quad b_2 = -8$

Her signature is $(b_1, b_2) = (9,-8)$

She sends d, (b_1, b_2), (w_1, w_2) to Bob

Bob uses (b_1, b_2) and (w_1, w_2) to get

$9(10,90) + (-8)(10,100) = (10,10) = s$

so he gets s. He checks if s is close to d

$\|s - d\| = \left((10-5)^2 + (10-7)^2\right)^{1/2} = \sqrt{34}$

Pretty good

12

Eva could write d in terms of bad basis

$$(5,7) = a_1(10,10) + a_2(10,100)$$

$$a_1 = 43/90 \qquad a_2 = 2/90$$

$$s = \left\lfloor \frac{43}{90} \right\rfloor (10,10) + \left\lfloor 2/90 \right\rfloor (10,100) = (0,0)$$

$$\| d - s \| = \sqrt{5^2+7^2} = \sqrt{74}$$

13 However, if Eve could find a basis for L of almost mutually perpendicular vectors she could use the same scheme as Alice to find the closest vector. There is an algorithm to do this, the LLL named for Lenstra, Lenstra and Lovacs. It is much like Gram-Schmidt but needing integers for solutions. This makes it a much more time consuming process. So for very large lattices, it may not be feasible. The method can be found in the book by Hoffstein, Pipher and Silverman and is in any computer algebra package.

The Knapsack

Problem: Given integers $a_1, \ldots, a_n$ and S where S is the sum of some of them, which ones are used in the sum? This can be a hard problem, but in a special case, there is an algorithm which solves it.

Def. A sequence of positive integers, $a_1, \ldots a_n$ if

$$a_1 + \cdots + a_j < a_{j+1} \quad \text{for } j = 1, \ldots, n-1$$

Ex 1, 3, 7, 15, 31 is superincreasing

Suppose some of integers in the superincreasing sequence add to S, the following algorithm finds these integers

15

Algorithm: Check if $a_n < S$

If yes, then a_n is part of the sum, keep a_n and compute $S - a_n = S$

If no, drop a_n

Now repeat the process using the new S and a_{n-1}.

Continue until a new $S = a_j$, some j, then stop. The kept numbers add to the original S

Ex Consider 2, 5, 9, 20, 37 and $S = 51$

$37 < 51$ Yes, keep 37 $S = 51 - 37 = 14$

$20 < 14$ No, do not keep 20

$9 < 14$ Yes keep 9 $S = 14 - 9 = 5$

$5 < 5$ No but $S = 5$, keep 5 and stop

$S = 37 + 9 + 5$

16 Knapsack Cipher

Let $a_1, \dots a_n$ be a super increasing sequence constructed by Alice

She picks $p > 2a_n$, e with $\gcd(e,p)=1$ and d with d the inverse of e mod p

Alice computes a new sequence

$$b_i = da_i \bmod p$$

and makes the non super increasing sequence $b_1, \dots b_n$ and p public.

Bob wants to send $(m_1, \dots, m_n)$, each $m_i = 0$ or 1 to Alice

Bob computes $m = \sum m_i b_i \bmod p$ and sends m to Alice

Alice computes

$\cancel{s = md = \sum m_i b_i d = \sum m_i a_i e d}$

$s = me = \sum m_i b_i e = \sum m_i a_i d e = \sum m_i a_i$

She wants to find which a_i add to s. They are the a_i with $m_i = 1$

17 Ex Alice picks superincreasing sequence $\{3, 11, 27, 50, 115\} = \{a_1, \ldots a_5\}$, $p = 250$, $d = 113$ $e = 177$

Alice finds $d a_i = b_i \bmod p$

$\{89, 243, 212, 150, 245\}$

$= \{b_1, b_2, b_3, b_4, b_5\}$

Bob wants to send $(1,0,1,0,1) = m = (m_1, \ldots, m_5)$

~~Bob computes~~ $\cancel{\sum_{i=1}^{5} m_i b_i}$

Bob computes $\sum m_i b_i \bmod p = 46$

and sends $m = 46$ to Alice

Alice computes $m \cdot e = 142 \bmod p$

Apply the superincreasing sequence algorithm using 3, 11, 27, 50, 115 on $s = 142$. She finds

$1 \cdot 3 + 0 \cdot 11 + 1 \cdot 24 + 0 \cdot 50 + 1 \cdot 115 = 142$

18 What can Eve do?

Eve has $b_1, \cdots b_n$ and m and p

Construct a lattice with basis

The rows of

$$\begin{pmatrix} 2 & 0 & & 0 & b_1 \\ 0 & 2 & & 0 & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & & 2 & b_n \\ 1 & 1 & & 1 & m \end{pmatrix} = \begin{pmatrix} B_1 \\ \vdots \\ B_n \\ B \end{pmatrix}$$

Now $m_1 B_1 + \cdots + m_n B_n - B =$

$S = (2m_1 - 1, 2m_2 - 1, \cdots, 2m_n - 1, 0)$

The last term is 0 since

$m_1 b_1 + \cdots + m_n b_n = m$

The last vector $= (\pm 1, \pm 1, \cdots \pm 1, 0)$

a short vector in L If we

can find an orthogonal basis

for L, the short vector in the

basis likely will be the short

vector S and the solution appears

~~Each In S, +1 has mc the~~

In S a component 1 comes from

$m_i = 1$. A component -1 comes from

$m_i = 0$

So we get $(m_1, \ldots, m_n)$ Bob's message

One finds this short vector by find a mutually orthogonal basis (or close). This is using the LLL algorithm due to Lenstra, Lenstra and Lovaos

The algorithm follows the idea of Gram-Schmidt. But in Gram-Schmidt one obtains non-integer entries. The LLL fixes this problem at a big expense in complexity. The LLL is a package in any computer algebra system we will not explain it. However

1. The LLL is discussed in An Introduction to Mathematical Cryptography by Hoffstein, Pipher and Silverman

2 The LLL has many applications aside from cryptography whenever a close to orthogonal basis is useful

3. In the 1980's, the knapsack was a favorite cryptosystem. But then the attack just described made it insecure