MA 437

LESSON 18

LATTICES

The GGH. CrypTosystem

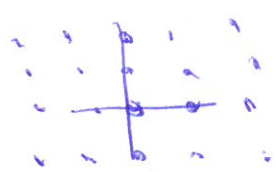
LATTICES

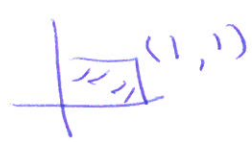
Let $v_1, \cdots, v_n$ be a basis for $\mathbb{R}^n$. All entries in the v_i are integers. The lattice, L, formed by the v_i consists of all <u>integer</u> combinations of the v_i. If $v \in L$, then

$$v = t_1 v_1 + \cdots + t_n v_n = (t_1, \cdots, t_n)\begin{pmatrix} v_1 \\ v_n \end{pmatrix}$$

where the t_i are integers. Another way of writing this is to let $A = \begin{pmatrix} v_1 \\ v_n \end{pmatrix}$ and then $v = (t_1, \cdots t_n) A$. The collection of all vectors $w = x_1 v_1 + \cdots + x_n v_n$ where $0 < x_i < 1$ is called the fundamental domain, F, of L.

Ex $v_1 = (1,0)$ $v_2 = (0,1)$ in $\mathbb{R}^2$

L: 

F 

Let U be a matrix whose entries are integers. Then U has an inverse, U^{-1}, whose entries are integers if and only if $\det U = \pm 1$. Again $A = \begin{pmatrix} N_1 \\ \vdots \\ N_n \end{pmatrix}$

Then $B = UA$ has rows that constitute another basis for L

Let $(x_1, \dots, x_n) \in F$. Then

$(x_1, \dots x_n) = t_1 N_1 + \dots + t_n N_n$, $0 < t_i < 1$

The volume of F =

$$\int_F dx_1 \dots dx_n = \int \det\left(\frac{\partial t_i}{\partial x_j}\right) dt_1 \dots dt_n$$

$$= \int \det A \, dt_1 \dots dt_n = \det A \int dt_1 \dots dt_n$$

where $\int dt_1 \dots dt_n$ is the n-volume of an n-cube $= 1$. Hence

$$\mathrm{vol}(F) = \det A.$$

If $UA = B$ and F' is the fundamental domain for the rows in 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 a fundamental domain for the lattice

F is the region bonded by $v_1, \ldots v_n$, so $\text{vol } F = |v_1| \cdots |v_n|$.

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

$$\left(\frac{\det A}{|v_1| \cdots |v_n|} \right)^{1/2} \leq 1$$

If this is close to 1, the vectors are close to being orthogonal

A good basis is one in which the vectors are close to orthogonal.

Otherwise, the basis is called a bad basis. The check is

$$\left(\frac{\det A}{|v_1| \cdots |v_n|} \right)^{1/2}$$

This quotient is called the Hadamard quotient.

Ex. Let $v_1 = (2,1,3)$ $v_2 = (1,2,0)$ $v_3 = (2,-3,-5)$

$$A = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \begin{pmatrix} 2 & 1 & 3 \\ 1 & 2 & 0 \\ 2 & -3 & -5 \end{pmatrix}$$

Let $U = \begin{pmatrix} 1 & 0 & 1 \\ 1 & -1 & 2 \\ 1 & 2 & 0 \end{pmatrix}$ $\det U = -1$

Let $B = UA = \begin{pmatrix} 4 & -2 & -2 \\ 5 & -7 & -7 \\ 4 & 5 & 3 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix}$

w_1 w_2 w_3 are also a basis for L

$w_1 = v_1 + v_3$ $w_2 = v_1 - v_2 + 2v_3$

$w_3 = v_1 + 2v_2$

Also $U^{-1} = \begin{pmatrix} 4 & -2 & -1 \\ -2 & 1 & 1 \\ -3 & 2 & 1 \end{pmatrix}$

so $A = U^{-1}B$ $\rightarrow$ we can express

The vector v_1, v_2, v_3 in terms of w_1, w_2, w_3 using $A = G^{-1}B$

Problem. Express v_1, v_2, v_3 in terms of w_1, w_2, w_3.

Shortest vector and closest vector Problems

Many problems depend on finding either the shortest vetor in L OR given vector w not in L to find the closest $v \in L$ to w. These are called the shortest and closest vector problems. They are much easier to solve if the basis is close to being orthogonal. For instance in the shortest vector case, if the basis consists of ortogonal vectors, then $w \in L \rightarrow$

$$w = \alpha_1 v_1 + \quad + \alpha_n v_n$$

$$|w|^2 = \alpha_1^2 |v_1|^2 + \quad + \alpha_n^2 |v_n| \quad \text{since}$$

$$v_i \cdot v_j = 0 \quad i \neq j.$$

The shortest w is then the shortest v_i (with $\alpha_i = 1$)

That is, the shortest vector in L is the shortest vector in the basis.

For the closest vector problem, let $w = t_1 v_1 + \cdots + t_n v_n$ where the t_i need not be integers, so $t \notin L$. If $v = s_1 v_1 + \cdots + s_n v_n \in L$, then

$$|v - w| = (s_1 - t_1)^2 |v_1|^2 + \cdots + (s_n - t_n)^2 |v_n|^2$$

The shortest of these is gotten by taking s_i = integer closest to t_i, i.e. $v = \lfloor t_1 \rceil v_1 + \cdots + \lfloor t_n \rceil v_n$

($\lfloor t \rceil$ = integer closest to t)

We have just described Babai's algorithm for finding the closest vector to w. This only works if the basis is a good basis (That's why we like them)

All this means that given a basis for L we would like to find a good basis for L, meaning an orthogonal basis. This would be Gram-Schmitt if the coefficents did not need to be integers. The algorithm that is used, the L.L.L is a take off on Gram-Schmitt but is much more computation heavy with a lot of repeated steps. It's downfall is when L has very large dimension because of the time involved.

A Cryptosystem: GGH.

In R^n, Alice picks a good integer basis for L, $\dim L = n$. She checks by using the Hadamard ratio. She constructs an integer matrix, U, with ~~U=~~ $\det U = \pm 1$ (so it has an inverse) Let $V = \begin{pmatrix} v_1 \\ v_n \end{pmatrix}$ and $W = UV = \begin{pmatrix} w_1 \\ w_n \end{pmatrix}$

She checks to see if W has rows that make up a bad basis by checking the Hadamard ratio for $w_1, \ldots w_n$. If they are a bad basis, she makes

$w_1, \ldots w_n$ public

Bob wants to send the message, a row vector m to Alice. He picks a small pertubation vector and computes the cipher text

$$e = mW + r$$

e is not in L because of r.

It is close to L because r is ~~small~~ short. Alice can compute m from e using her good basis and Babai's algorithm

Ex. Alice picks good basis

$v_1 = (10, 0)$ $v_2 = (0, 10)$ and

$$U = \begin{pmatrix} 1 & 9 \\ 1 & 10 \end{pmatrix} \quad \det U = 1$$

She computes

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

She checks the Hadamard ratio for both bases (The first clearly is a good basis since the vectors are perpendicular)

11 Alice makes public the bad basis $w_1 = (10 \quad 90)$ $w_2 = (10 \quad 100)$

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

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

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

and sends e to Alice

Alice computes

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

getting $n_1 = 8.1$ $n_2 = 77.1$

She uses Babai's algorithm

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

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

where

12

$$U \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \quad \text{gives}$$

$$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = U^{-1} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} \quad \text{and}$$

$$U^{-1} = \begin{pmatrix} 10 & -9 \\ -1 & 1 \end{pmatrix}$$

Eve has $e = (81, 771)$ and the bad basis w_1, w_2. She can try:

$$(81, 771) = a(10, 90) + b(10, 100)$$

$$81 = 10a + 10b$$
$$771 = 90a + 100b$$

$$\rightarrow 39 = 10a \qquad a = \lfloor 39/10 \rfloor = 4$$
$$4\sim = 10b \qquad b = \lfloor 4\sim/10 \rfloor = 4$$

$$(81, 771) \approx 4(10, 90) + 4(10, 100)$$

and does not find m.

13 GGH Signature Scheme

As in the cryptosystem, Alice picks good basis $v_1, \ldots v_n$ and computes a bad basis for the lattice using an integer matrix U with $\det U = \pm 1$.

Alice sets out the bad basis, $w_1, \ldots, w_n$ for everyone to see. This is her verification key

Alice is to send document d to Bob. She computes the closest vector s to d in L using Babai's algorithm and the good basis. She then writes s in terms of the bad basis. The coefficients are her signature, ie she

write $s = d_1 w_1 + \cdots + d_n w_n$ and $(d_1, \ldots d_n)$ is her signature.

She send the document to Bob as well as her signature

Bob uses her signature $(d_1, \ldots, d_n)$ and bad basis $w_1, \ldots w_n$ to get s back. If $|s-d|$ is small, then Alice sent the message.

Ex. Alice picks good basis

$v_1 = (10,0)$, $v_2 = (0,10)$ and

$U = \begin{pmatrix} 1 & 9 \\ 1 & 10 \end{pmatrix}$. Then

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

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

is Alices verification key which she makes public.

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

She writes

$$d = b_1(10,0) + b_2(0,10) \rightarrow$$

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

She uses Babi's algorithm to set $s \in L$ close to d

$$s = \lceil b_1 \rfloor v_1 + \lceil b_2 \rfloor v_2$$

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

She writes s in terms of the bad basis

$$(10,10) = c_1(10,90) + c_2(10,100)$$

$c_1 = 9 \quad c_2 = -8$

solved from

$$10 = c_1\, 10 + c_2\, 10$$
$$10 = c_1\, 90 + c_2\, 100$$

16 Her signature is $(9,-8) = (c_1, c_2)$

She sends $d, (c_1, c_2)$ to Bob

Bob uses her verification key w_1, w_2 to get

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

He checks if s is close to d

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

Pretty good

Eve can write d in terms of the bad basis

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

$$a_1 = \frac{43}{90} \qquad a_2 = 2/90$$

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

$$\|d-s\| = \sqrt{5^2+7^2} = \sqrt{74}$$ Larger than what Bob found.

Problem 1. Alice picks good basis

$v_1 = (0, 10)$ $v_2 = (10, 0)$ and $U = \begin{pmatrix} 1 & 10 \\ 1 & 10 \end{pmatrix}$

She computes bad basis

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = U \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

Alice make public (w_1, w_2)

What is her public key?

Bob sends her ciphertext

$(861, 81)$

What is Bob's message?

Problem 2: Alice picks good basis $v_1 = (0, 10)$, $v_2 = (10, 0)$ and

$$U = \begin{pmatrix} 1 & 10 \\ 1 & 11 \end{pmatrix}$$

What is Alices bad basis?
What is her verification key?

Alice makes her verification key public

Alice is to send document $d = (5, 2)$ to Bob. She computes the closest vector s to d and writes s in terms of the bad basis. What is her signature?

Alice sends d and her signature to Bob, who uses them and her verification key to get s back.

What does Bob compute to see if it is Alices signature?