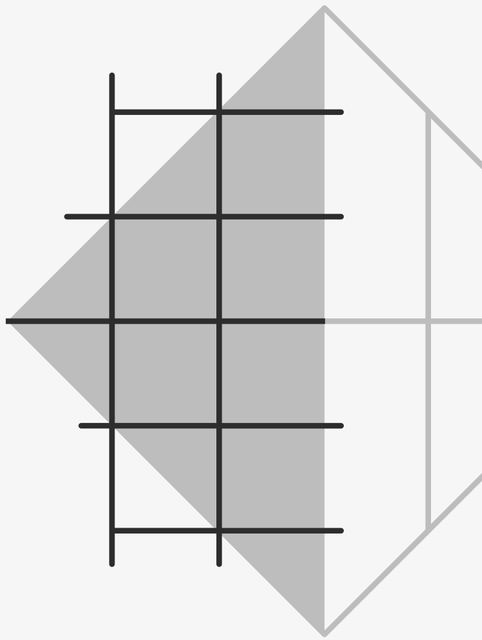
Learning with error

- Panupong Sangaphunchai
- 6580587

Outline

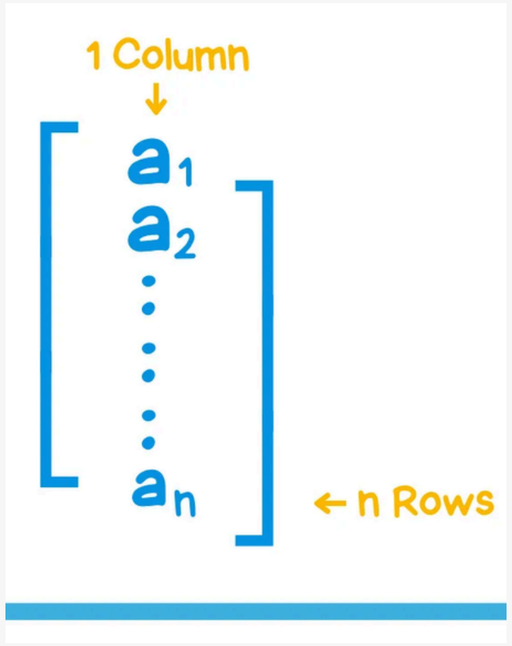
- What is learning with error?
- Learning with error and lattices
- Learning with error and Module-LWE



Notation

$$\mathbb{Z}_q = \{0,1,2,\ldots,q-1\}$$

$$x \in_R S$$



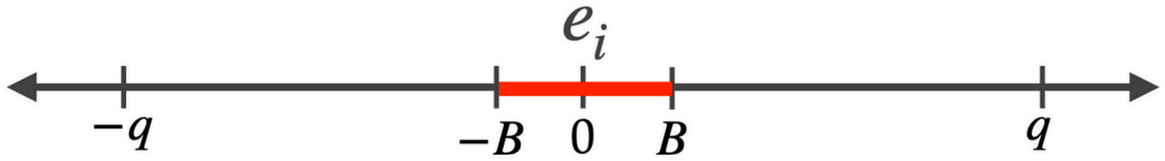
1. a set of integers from 0 to q-1

2. x is chosen at random from the set S
   - R = random
   - random = selected uniformly and independely
   - Uniformly = each item in the set has equal probability
   - Independently = The outcome of selecting other items doesn't affect the other.

3. All vectors are column vectors, a matrix with 1 column

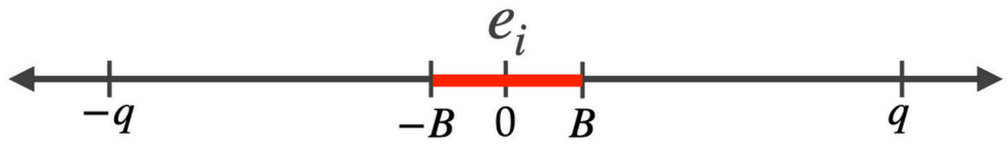
Learning with error

Definition. *Learning With Errors problem*: LWE(m, n, q, B)
Let $s \in_R \mathbb{Z}_q^n$ and $e \in_R [-B, B]^m$ where $B \ll q/2$.
Given $A \in_R \mathbb{Z}_q^{m \times n}$ and $b = As + e \pmod{q} \in \mathbb{Z}_q^m$, find s.

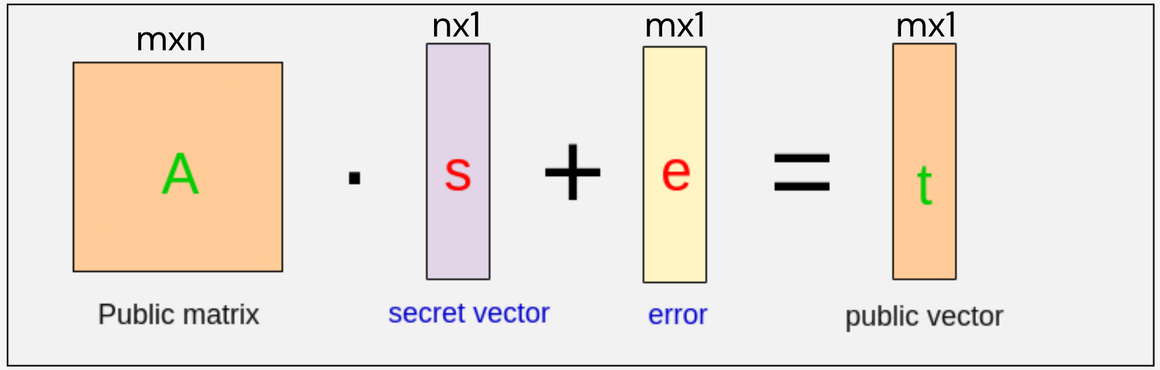


- 4 parameters
  - m
  - n
  - q is a prime number.
  - B is an error-bound that is much smaller than q/2

Definition. *Learning With Errors problem*: LWE(m, n, q, B)
Let $s \in_R \mathbb{Z}_q^n$ and $e \in_R [-B, B]^m$ where $B \ll q/2$.
Given $A \in_R \mathbb{Z}_q^{m \times n}$ and $b = As + e \pmod{q} \in \mathbb{Z}_q^m$, find s.

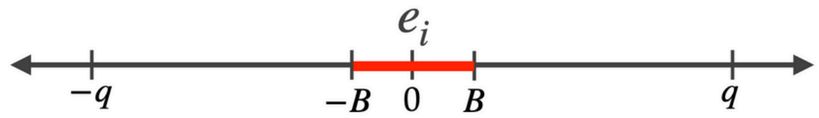


- **s** is a length n vector whose components are randomly selected integers between 0 and q-1
- **e** is a length m vector whose components are randomly selected integers between -B and B
- s and e are secret; A and b are public
- **A** is an m x n matrix whose entries are randomly selected integers between 0 and q-1
- **b** is the result vector from As+e (modq). note that you need to mod q the result after computing As + e.
  - b is a vector of length m.
  - entries will be between 0 and q-1
- **B** << q/2
- the noise can push past the mid point
- might mistake 0 for 1 when decrypting



Example

Definition. *Learning With Errors problem*: LWE(m, n, q, B)
Let $s \in_R \mathbb{Z}_q^n$ and $e \in_R [-B, B]^m$ where $B \ll q/2$.
Given $A \in_R \mathbb{Z}_q^{m \times n}$ and $b = As + e \pmod{q} \in \mathbb{Z}_q^m$, find s.



- m = 5, n = 3, q =31 and B = 2;
- b = As + e (mod 31)
- A = 5 x 3 matrix with entries between 0 - 30
- s = a vector of length 3 with entries between 0 - 30
- e = a vector of length 5 with entries between - 2 and 2
- b = a vector of length 5 with entries between 0 - 30

✦ **LWE instance:**

$$A = \begin{bmatrix} 11 & 3 & 27 \\ 12 & 21 & 7 \\ 6 & 23 & 30 \\ 5 & 6 & 2 \\ 21 & 0 & 14 \end{bmatrix} \quad \begin{matrix} s \\ \begin{bmatrix} \\ \\ \end{bmatrix} \\ 3\text{x}1 \end{matrix} + \begin{matrix} e \\ \begin{bmatrix} \\ \\ \\ \\ \\ \end{bmatrix} \\ 5\text{x}1 \end{matrix} = \begin{bmatrix} 25 \\ 25 \\ 12 \\ 29 \\ 17 \end{bmatrix}.$$

5x3

5x1

B parameter

1. B should not be 0 becaus e would be 0. As = b can be solved efficiently using Gaussian elimination
    a. Given a linear system expressed in matrix form, *Ax = b, we reduce matrix into row echelon form using row operations*
    b. row operations → interchange 2 rows, multiply a row by a constant, add a multiple of 1 row to another.

$$\begin{aligned} a_1x + b_1y + c_1z &= d_1 \\ a_2x + b_2y + c_2z &= d_2 \\ a_3x + b_3y + c_3z &= d_3 \end{aligned} \quad \blacktriangleright \quad \underset{[A]}{\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \underset{[B]}{\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}}$$

Gauss-Jordan elimination

$$\left[\begin{array}{ccc|c} 1 & 0 & 0 & * \\ 0 & 1 & 0 & * \\ 0 & 0 & 1 & * \end{array}\right]$$

reduced row echelon form

B parameter

2. If B = (q-1)/2, then e will have the full range of integer q (-(q-1)/2 to (q-1)/2, therefore, the answer b vector will cover the full range of q, making finding s is theorectically impossible to find. Therefore, we are assuming B < q/4

For example, m = 1, n =1, B = 3, q =7, b = As + e (mod q)

- A is a vector of length 1 i.e. [2]
- e = [-3,3]
- s^1 = 3
- b = 2*3 + e (mod 7) = 6 + e (mod 7)
- if e = -3, b = 6 - 3 (mod 7) = 3
- if e = -2, b = 6 -2 (mod 7) = 4
- if e = -1, b = 5
- if e = 0, b = 6; e = 1 → b = 0.
- if e = 2 → b = 1, e = 3 → b = 2.
- b is a member of {0,1,2,3,4,5,6}
- s^2 = 5
- b = 2*5 + e (mod 7) = 10 + e (mod 7)
- if e = -3, b = 10 - 3 (mod 7) = 0
- if e = -2, b = 10 -2 (mod 7) = 1
- if e = -1, b = 2
- if e = 0, b = 3; e = 1 → b = 4
- if e = 2 → b = 5, e = 3 → b = 6.
- b is a member of {0,1,2,3,4,5,6}

B parameter

3. If B is asymptotically smaller than $\sqrt{n}$, then LWE can be solved within subexponential runtime for sufficiently large m >> n

- Arora–Ge attack, if noise is small,treat equations as if it’s noiseless, turn algebraic equations and solve them algebraically using algrebraic tools for solving multivariate polynomials

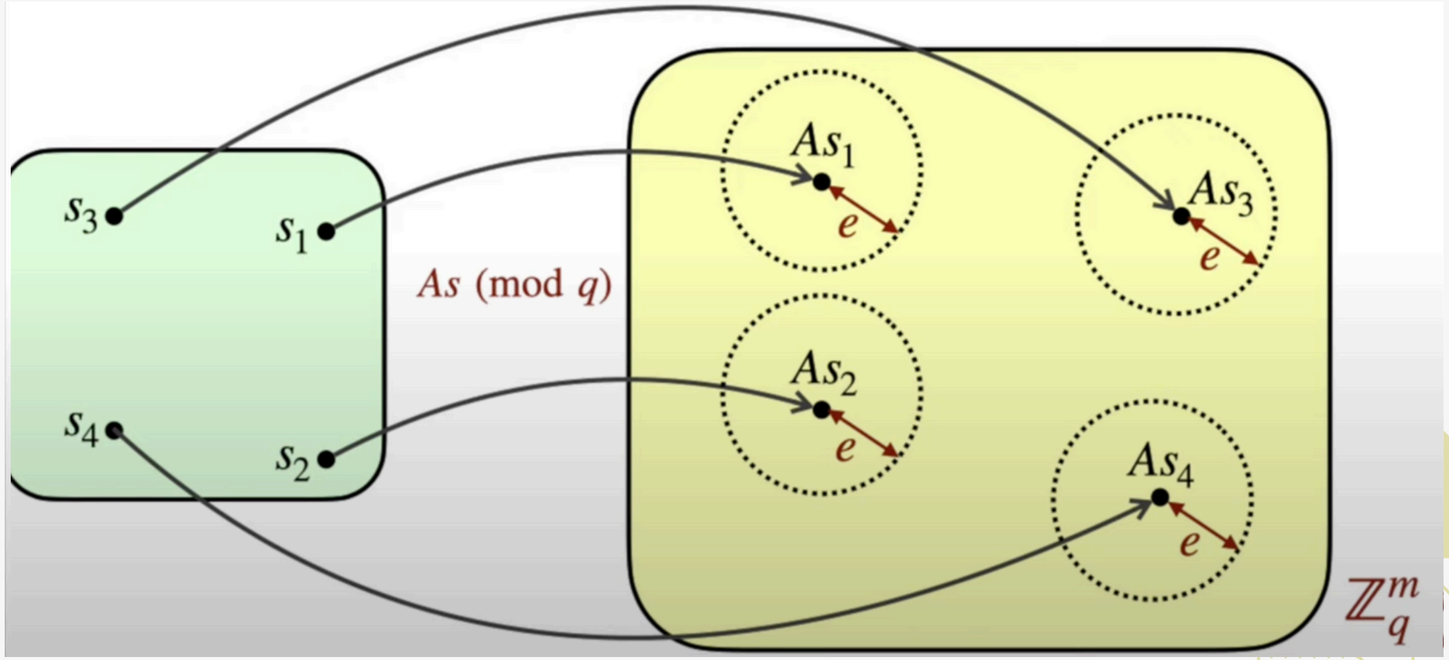
Constraints	Comments
$B < q/2$	To ensure modular arithmetic doesn't wrap unpredictably
$B < q/4$	A safer margin — standard in crypto settings
$B << \sqrt{n}$	Makes LWE easier — attackers can exploit this
$B \gtrsim \sqrt{n}$ and $B<q/4$	Hard regime for LWE — used in cryptography

m and n parameter

- if m is much larger than n, then we can assume that there is a unique LWE solution (s,e)
- s is a vector of length n that is chosen from all possible sets of vectors whose entries are between 0 and q-1
- As (mod q) is vector of length m that is chosen from all possible sets of vectors whose entires are between 0 and q - 1
- Adding e will create a sphere of all possible answer in the set of all possible vectors of length m modulo q
- If all q^n spheres do not overlap then the LWE solution is unique

$$\begin{bmatrix} q \\ q \\ q \end{bmatrix}$$

q^3



Lattice

A lattice L in R^n is the set of all integer linear combination of m linearly indepedent vector ,B = {v1,v2,...,vm} in R^n where m<= n. the dimension of the lattice and the rank is n and m respective.

- R^n is n-dimensional real space, which means each real number represent n axes.
- i.e n = 2 → (1.0,3.48)
- B is m linearly indepedent vectors i.e. {v1,v2,...,vm}
- integer linear combination is a combination of addition or subtraction of many vectors with coefficient in front as integers
- dimension is the number of axes
- rank is the number of vector used to generate the lattice
- vectors are in Z^m . each vector is a nx1 matrix
- We can look at B as nxm matrix then L(B) = {Bx:x∈Z^m }

vectors:

$$v_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

nation might be:

$$2v_1 - 1v_2 = 2\begin{bmatrix} 1 \\ 2 \end{bmatrix} - 1\begin{bmatrix} 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \end{bmatrix} - \begin{bmatrix} 3 \\ 1 \end{bmatrix} = \begin{bmatrix} -1 \\ 3 \end{bmatrix}$$

$$B = \begin{bmatrix} | & | & \cdots & | \\ v_1 & v_2 & \cdots & v_m \\ | & | & \cdots & | \end{bmatrix}.$$

nxm

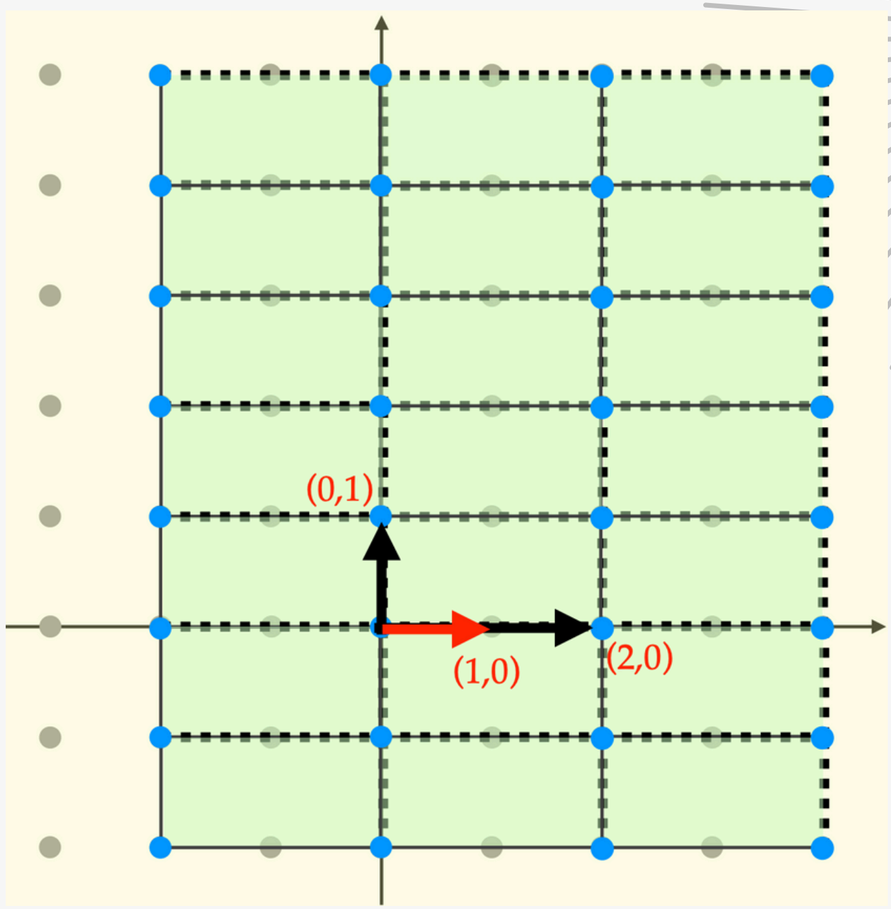
x = L(B)

mx1 nx1

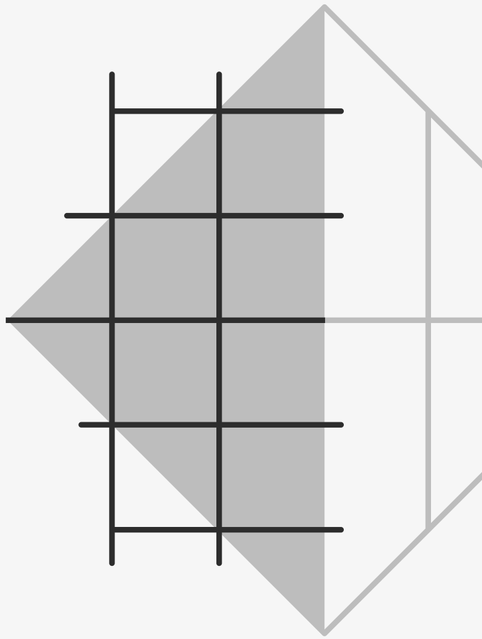
Example

- n=2 and B = {(2,0),(0,1)}
- Then lattice L(B) = {Bx:x∈Z^2 }

$$\underset{2\times 2}{\begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}}^{B} \quad \underset{2\times 1}{\begin{bmatrix} \\ \end{bmatrix}}^{x} = \underset{2\times 1}{\begin{bmatrix} \\ \end{bmatrix}}^{L(B)} \quad \underset{2\times 1}{\begin{bmatrix} \\ \end{bmatrix}}^{L(B)} \quad \underset{2\times 1}{\begin{bmatrix} \\ \end{bmatrix}}^{L(B)} \cdots$$



SVP (Shortest vector problem)



Determining the shortest non-zero vector of a lattice in L, given L = L(B) is a subset of Z^n .shortest = has the smallest Euclidean norm

- (3,4) = $\sqrt{(3^2 + 4^2)}$
- SVP is NP-hard
- NP-hard means that Solving requires exponential time.
- Checking a solution to an NP-hard problem may or may not be polynomial time.
- **Ajtai, Miklós. "The shortest vector problem in L2 is NP-hard for randomized reductions.**
- L2 means l2-norm or euclideans norm (|| ||).
- randomized reductions is using random input from a hard problem to reduce another problem.

$$||x|| = \sqrt{x_1^2 + x_2^2}$$

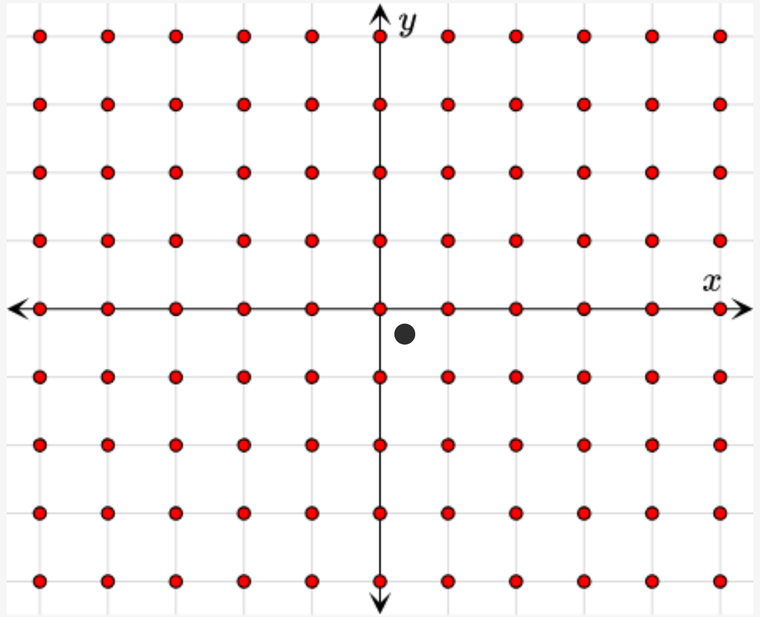
LWE is related to lattice

- because LWE is also considered a lattice problem because it maps a problem called Bounded distance decoding (BDD)

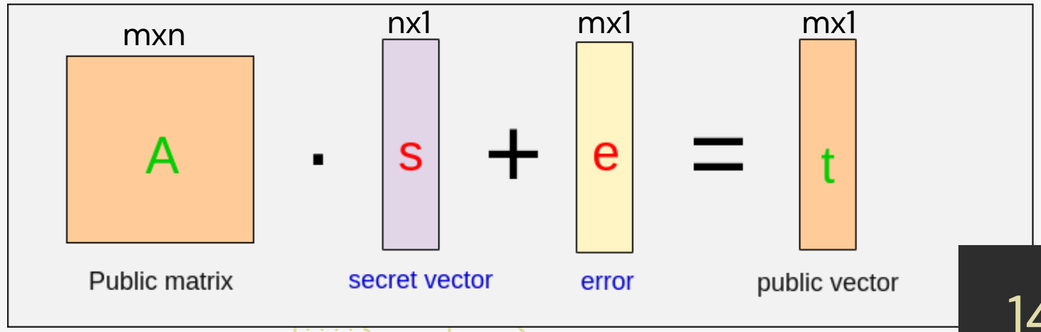
LWE(m, n, q, B). Let $s \in_R \mathbb{Z}_q^n$ and $e \in_R [-B, B]^m$.
Given $A \in_R \mathbb{Z}_q^{m \times n}$ and $b = As + e \pmod q$, find s.

LWE lattice:
$L_A = \{y \in \mathbb{Z}^m : As = y \pmod q \text{ for some } s \in \mathbb{Z}^n\} \subseteq \mathbb{R}^m$.

  - LWE lattice is all possibles point derived from As (mod q) for some s.
  - s is not in modulo q anymore because we want to measure euclidean distance to find the distance of noise or the upperbound α (√m * B)



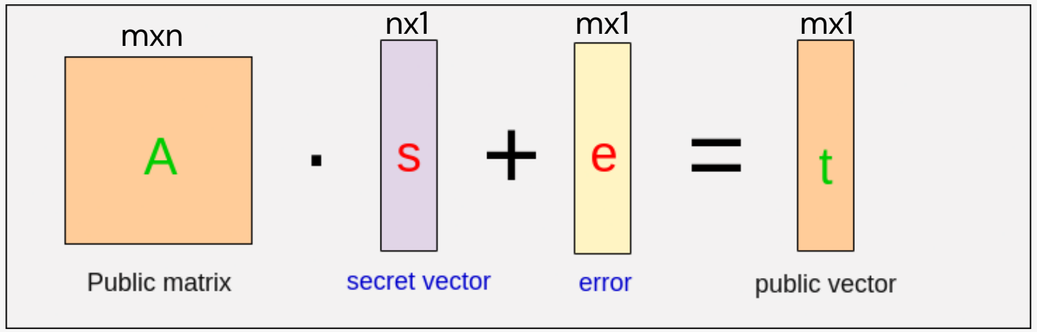
Note that for an LWE instance (A, b, s, e), we have
$y = As \bmod q \in L_A$, and $\|b - y\|_2 = \|e\|_2 \leq \sqrt{m}\, B$.



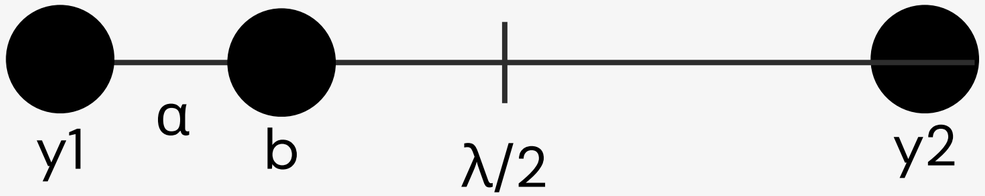
BDD

Bounded Distance Decoding (BDD_α):
Given a lattice $L = L(D) \subseteq \mathbb{R}^m$ and $b \in \mathbb{R}^m$ with the guarantee that there is a unique $y \in L$ within distance α of b, find y.

- R^m is m-dimensional space, m is the amount of elements in public vectors that are the result of As+e (mod q) = b.
- Find the unique lattice point y within radius α of b



- We can guarantee that there is only 1 y because we assume α < λ/2
- λ is the shortest distance between distince lattice points



- Since there is only 1 lattice point with α and other points are further away, meaning that is only one y that is clossest to b.
- If you can solve BDD on the right kind of lattice, you can solve LWE.

BDD to SVP

Summary: We can solve the BBD_α instance by solving SVP for $L(D')$ where $D' = \begin{bmatrix} D & -b \\ 0 & \alpha \end{bmatrix}$.

- This method is called “primal attack using Kannan embedding”
- Embed the BDD problem into a higher-dimensional lattice such that solving SVP in that larger lattice gives you the solution to the original BDD
- the security of LWE (and cryptographic schemes based on it) is tied to the hardness of solving SVP
- This is why LWE is considered a "hard problem" and a foundation for post-quantum cryptography.

How is LWE related to kyber?

- Kyber is based on Module-LWE
- Module-LWE is a variant of LWE but instead of using matrices with integer entires, it use matrices with Polynomial ring
- Polynomial ring : Rq = Zq [x]/(x^n+1)
- Zq[x] is a set of all polynomial in x with coefficient integer modulo q
- Degree = n-1 bound by /(x^n+1)
- making the scheme faster because ring elements can be computed faster using Number Theoretic Transform (NTT)., mathematical tool using Fast-Fourier transform-style algorithm.

e.g. $q = 17, n = 4$

$f(x) = 2 + 16x + 3x^2 + 5x^3$

$g(x) = 9 + x + 14x^3$

$f(x)g(x) = 18 + 146x + 43x^2 + 76x^3 + 229x^4 + 42x^5 + 70x^6$

↓ coef. mod q

$= 1 + 10x + 9x^2 + 8x^3 + 8x^4 + 8x^5 + 2x^6$

↓ modular reduction $/(x^n + 1)$

$= 10 + 2x + 7x^2 + 8x^3$

9

Module learning with error (MLWE)

$$\begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{l\times 1}$$

MLWE(n, k, ℓ, q, B):
Let $s \in_R R_q^\ell$ and $e \in_R S_B^k$ where $k > \ell$ and $B \ll q/2$.
Let $a_1, a_2, \ldots, a_k \in_R R_q^\ell$ and $b_i = a_i^T s + e_i \in R_q$ for $i = 1, \ldots, k$.
Given the a_i and b_i, determine s.

- Rq^l is polynomials ring in matrix lx1 (s and a)
- SB^k is small polynomials coefficient in [-B,B] in matrix kx1 (e)
- k>l because we want to provide more noise, making it statistically hard to recover s.

equation:
$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1\ell} \\ a_{21} & a_{22} & \cdots & a_{2\ell} \\ \vdots & \vdots & & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{k\ell} \end{bmatrix}_{k\times\ell} \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_\ell \end{bmatrix}_{\ell\times 1} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{bmatrix}_{k\times 1} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix}_{k\times 1}.$$

Kyber PKE - key generation

Key generation: Alice does:

1. Select $s \in_R S_{\eta_1}^k$.
2. Select $A \in_R R_q^{k \times k}$ and $e \in_R S_{\eta_2}^k$.
3. Compute $b = As + e$.
4. Alice's **public key** is (A, b); her **private key** is s.

- $q = 3329, \quad n = 256.$
- $R_q = \mathbb{Z}_{3329}[x]/(x^{256} + 1).$
- $k \in \{2,3,4\}.$
- $(\eta_1, \eta_2) \in \{(3,2), (2,2), (2,2)\}.$

- Computing s is an instance of single secret MLWE because s dimension is k the same as the number of noise
- note that coefficient of polynomials in s and e must be [-η1,η1] and [-η2,η2]

$$s = \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1}$$

$$b = As + e = \left(\begin{bmatrix} R_q & R_q & R_q \\ R_q & R_q & R_q \\ R_q & R_q & R_q \end{bmatrix}_{kxk} * \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1} \right) + \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1}$$

Kyber PKE - encryption

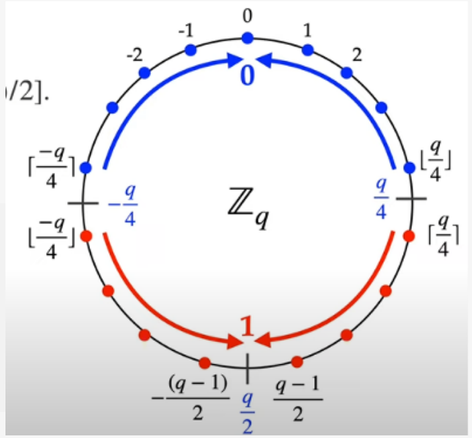
Encryption: To encrypt a message $m \in \{0,1\}^{256}$ for Alice, Bob does:

1. Obtain an authentic copy of Alice's encryption key (A, b).
2. Select $r, z \in_R S_{\eta_1}^k$ and $z' \in_R S_{\eta_2}$.
3. Compute $c_1 = A^T r + z$ and $c_2 = b^T r + z' + \lceil q/2 \rfloor m$.
4. Output $c = (c_1, c_2)$.

Note: $c \in R_q^k \times R_q$.

closest integer to q/2 but ties broken upward

$$r = \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_k \quad z = \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_k \quad z' = R_q$$



$$c1 = A^T r + e_1 = \begin{bmatrix} R_q & R_q & R_q \\ R_q & R_q & R_q \\ R_q & R_q & R_q \end{bmatrix}_{k \times k} \times \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1} + \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1}$$

A^T r z

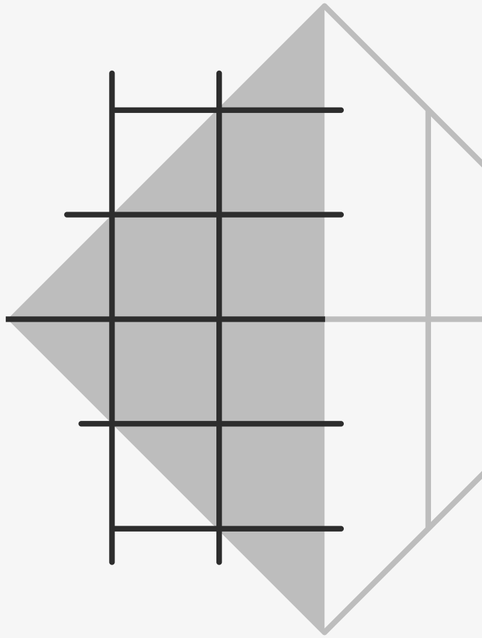
$$c2 = [R_q \quad R_q \quad R_q]_{1 \times k} \times \begin{bmatrix} R_q \\ R_q \\ R_q \end{bmatrix}_{k \times 1} + R_q + R_q$$

t^T r z'

Future work

Literature review: implementation of Kyber KEM

- Verilog
- Start looking at other implementation’s code



Reference

- https://cryptography101.ca/lattice-based-cryptography/
- https://people.csail.mit.edu/vinodv/CS294/lecture2.pdf
- https://eccc.weizmann.ac.il/eccc-reports/1997/TR97-047/index.html