What is Lattice-Based Cryptography?

An Introduction to Lattice-Based Cryptography and the Connection to Coding Theory

Jessica Bariffi

Quantum-Resistant Cryptography Group

SAN-OSL PhD Seminar

September 12, 2022

Motivation - Cryptography

Childhood example

- Invent a secret language to communicate with your friends (e.g. shift the letters in the alphabet by n, use completely new alphabet, ...)
- In finite time (polynomial time) other class mates cracked the code and understood you.

Motivation - Cryptography

Childhood example

- Invent a secret language to communicate with your friends (e.g. shift the letters in the alphabet by n, use completely new alphabet, ...)
- In finite time (polynomial time) other class mates cracked the code and understood you.

Goal: Want a secret language which is “hard” to crack in finite time.

Motivation - Cryptography

Childhood example

- Invent a secret language to communicate with your friends (e.g. shift the letters in the alphabet by n, use completely new alphabet, ...)
- In finite time (polynomial time) other class mates cracked the code and understood you.

Goal: Want a secret language which is “hard” to crack in finite time.

In a more serious world:

- Many sensitive data is send via a computer
(online banking, passport information, medical data)

Motivation - Cryptography

Childhood example

- Invent a secret language to communicate with your friends (e.g. shift the letters in the alphabet by n, use completely new alphabet, ...)
- In finite time (polynomial time) other class mates cracked the code and understood you.

Goal: Want a secret language which is "hard" to crack in finite time.

In a more serious world:

- Many sensitive data is send via a computer (online banking, passport information, medical data)
- We use a "secret language" also there (RSA, ...). We call this *encryption*.

Motivation - Cryptography

Childhood example

- Invent a secret language to communicate with your friends (e.g. shift the letters in the alphabet by n, use completely new alphabet, ...)
- In finite time (polynomial time) other class mates cracked the code and understood you.

Goal: Want a secret language which is “hard” to crack in finite time.

In a more serious world:

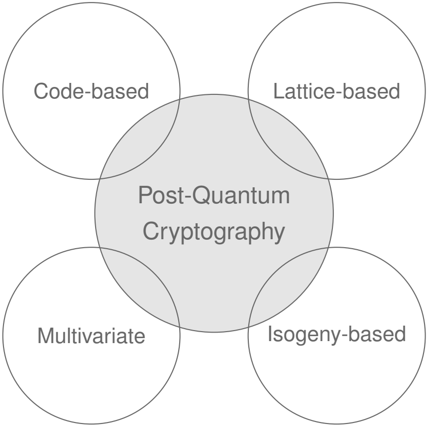
- Many sensitive data is send via a computer
(online banking, passport information, medical data)
- We use a “secret language” also there (RSA, ...). We call this *encryption*.
- Unauthorized parties use more and more powerful tools (soon probably quantum computers) that crack our encrypted data $\Longrightarrow$ *decryption*.

Motivation - Cryptography

- Current public key cryptosystems can be broken in polynomial time by a quantum computer $\Longrightarrow$ *Post-Quantum Cryptography*

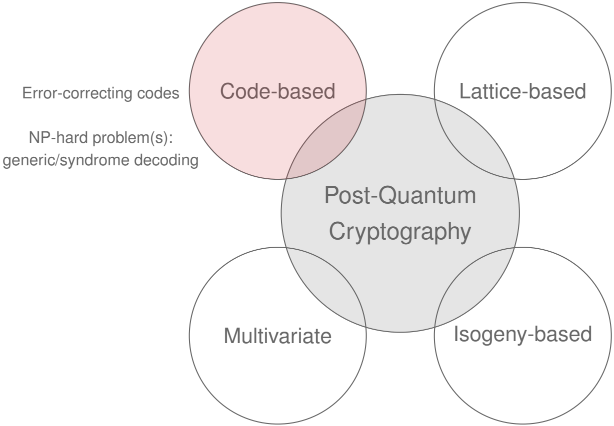
Motivation - Cryptography

- Current public key cryptosystems can be broken in polynomial time by a quantum computer $\Longrightarrow$ *Post-Quantum Cryptography*



Motivation - Cryptography

- Current public key cryptosystems can be broken in polynomial time by a quantum computer $\Longrightarrow$ *Post-Quantum Cryptography*

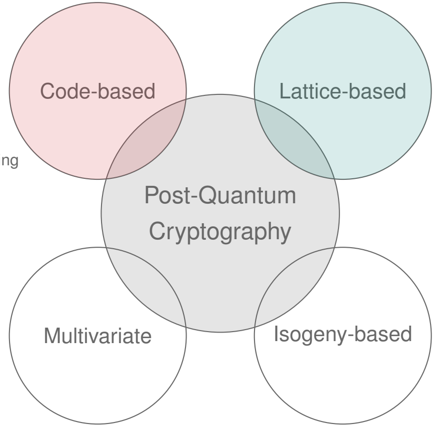


Motivation - Cryptography

- Current public key cryptosystems can be broken in polynomial time by a quantum computer $\Longrightarrow$ *Post-Quantum Cryptography*

Error-correcting codes

NP-hard problem(s):
generic/syndrome decoding

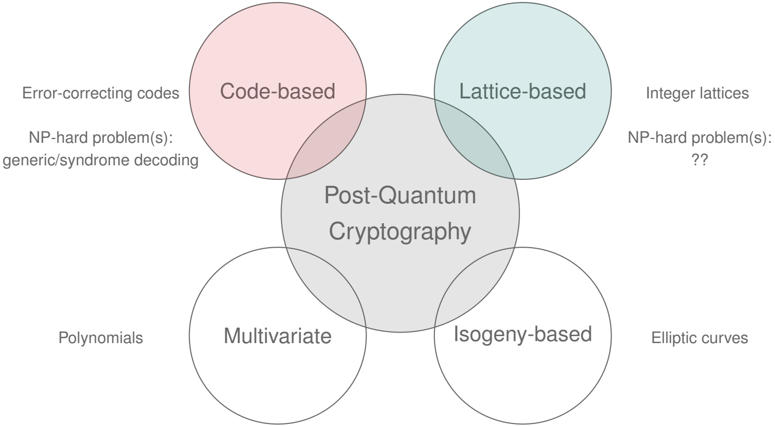


Integer lattices

NP-hard problem(s):
??

Motivation - Cryptography

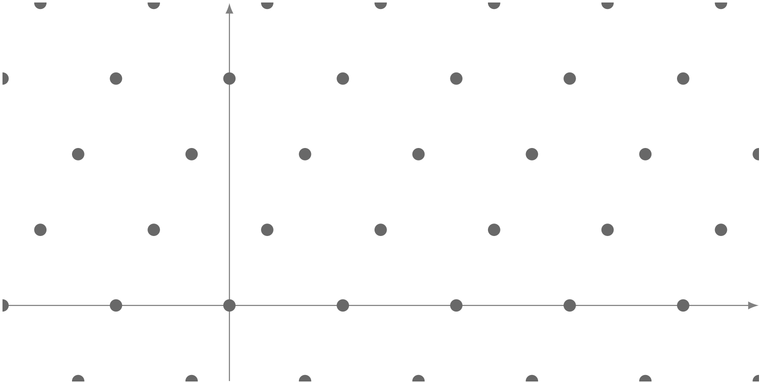
- Current public key cryptosystems can be broken in polynomial time by a quantum computer $\Longrightarrow$ *Post-Quantum Cryptography*



Outline

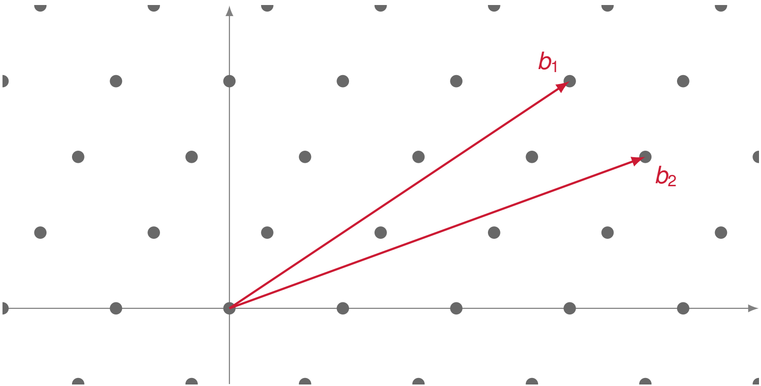
What is a lattice?

Let us consider the two-dimensional case



What is a lattice?

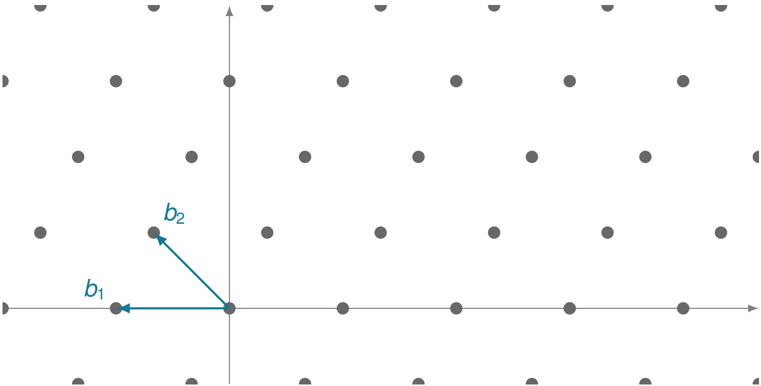
Let us consider the two-dimensional case



- Represent by basis vectors $\{b_1, b_2\} =: B \in \mathbb{Z}^{2\times 2}$.

What is a lattice?

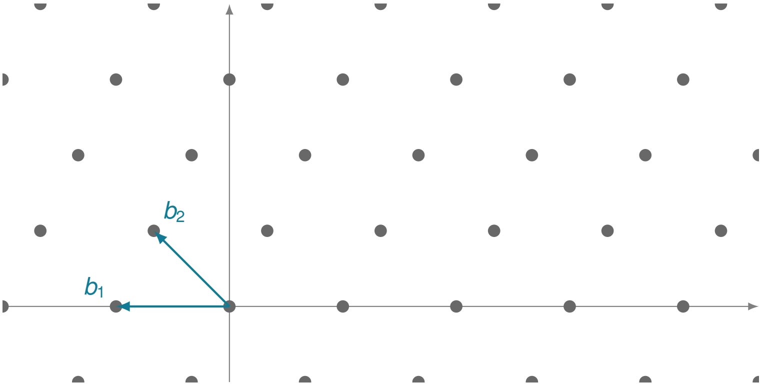
Let us consider the two-dimensional case



- Represent by basis vectors $\{b_1, b_2\} =: B \in \mathbb{Z}^{2\times 2}$.

What is a lattice?

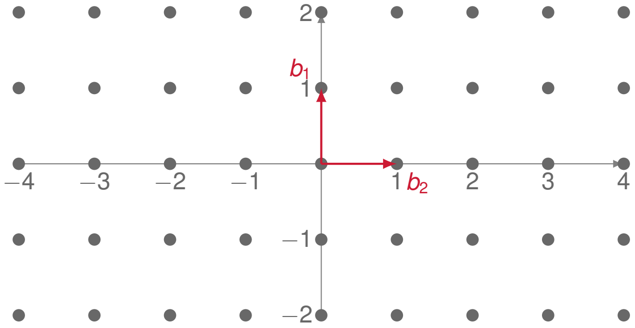
Let us consider the two-dimensional case



- Represent by basis vectors $\{b_1, b_2\} =: B \in \mathbb{Z}^{2\times 2}$.
- The set $\mathcal{L}(B) = \left\{ \sum_{i=1}^{2} b_i x_i \mid x_i \in \mathbb{Z}^2 \right\}$ is called a *lattice*.

Example of a lattice

Assume we have the basis $\left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$



The lattice generated by $B := \left\{ \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}$ is

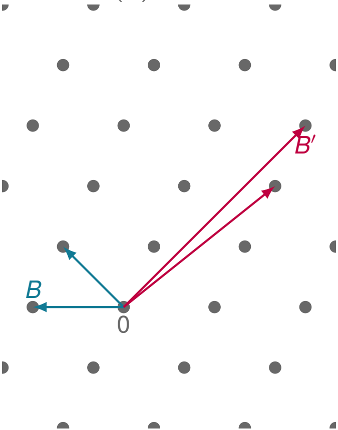
$$\mathcal{L}(B) = \left\{ \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} \middle| a_i, b_i \in \mathbb{Z} \right\} = \left\{ \begin{pmatrix} a_1 \\ b_2 \end{pmatrix} \middle| a_1, b_2 \in \mathbb{Z} \right\} = \mathbb{Z} \times \mathbb{Z}$$

Representation

We represent a lattice $\mathcal{L}$ by a matrix $B \in \mathbb{Z}^{n \times n}$ and write $\mathcal{L}(B)$.

- The matrix B is not unique.
- Some choices of B can make the algorithmic problems easier/harder.

Question: What is the "best" choice?



Representation

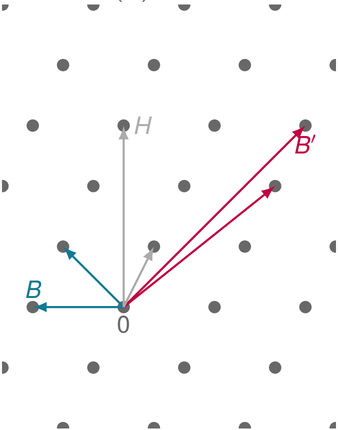
We represent a lattice $\mathcal{L}$ by a matrix $B \in \mathbb{Z}^{n \times n}$ and write $\mathcal{L}(B)$.

- The matrix B is not unique.
- Some choices of B can make the algorithmic problems easier/harder.

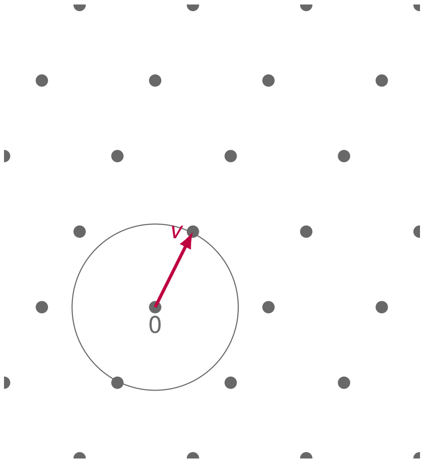
Question: What is the "best" choice?
$\implies$ *Hermite Normal Form* of any B.

This normal form is...

- unique (i.e., $\mathrm{HNF}(B) = \mathrm{HNF}(B')$)
- efficiently computable



Properties

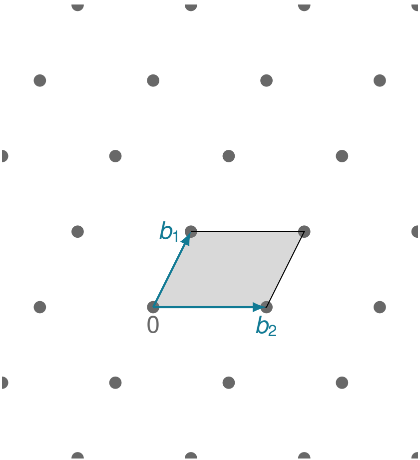


First minimum

$\lambda_1(\mathcal{L}) := \min_{x \in \mathcal{L} \setminus \{0\}} ||x||_2$

minimum distance between any two distinct lattice points.

Properties

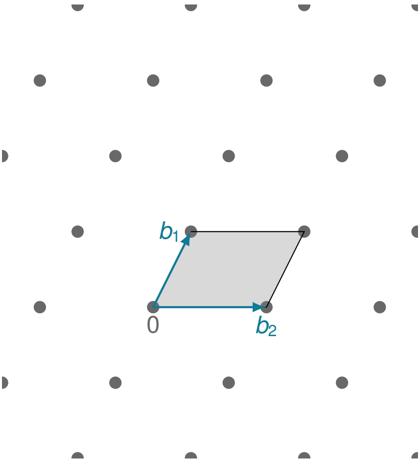


First minimum
$\lambda_1(\mathcal{L}) := \min_{x \in \mathcal{L} \setminus \{0\}} ||x||_2$

minimum distance between any two distinct lattice points.

Determinant
$\det(\mathcal{L}) := \text{vol}(\mathbb{R}^n/\mathcal{L}) = |\det(B)|$

Properties



First minimum

$\lambda_1(\mathcal{L}) := \min_{x \in \mathcal{L} \setminus \{0\}} ||x||_2$

minimum distance between any two distinct lattice points.

Determinant

$\det(\mathcal{L}) := \text{vol}(\mathbb{R}^n / \mathcal{L}) = |\det(B)|$

Minkowski's Theorem

$\lambda_1(\mathcal{L}) \leq \sqrt{n} \det(\mathcal{L})^{1/n}$

Some Algorithmic Problems on Lattices

1. Testing the equality (or inclusion) of lattices
2. Intersection of lattices
3. Computing a short vector of a lattice
4. Computing a lattice vector close to some target

Some Algorithmic Problems on Lattices

1. Testing the equality (or inclusion) of lattices

Equivalent lattices

For two matrices $B_1, B_2 \in \mathbb{Z}^{n \times n}$ it holds $\mathcal{L}(B_1) = \mathcal{L}(B_2)$ if and only if there is a unitary matrix $U \in \mathrm{GL}_n(\mathbb{Z})$ (i.e. $\det(U) = \pm 1$) such that $B_1 = B_2 U$.

Example

The following matrices generate the same lattice:

$$B_1 = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix} \quad \text{and} \quad B_2 = \begin{pmatrix} 1 & -3 \\ 1 & -4 \end{pmatrix}$$

because

$$B_1 = B_2 \begin{pmatrix} 11 & -9 \\ 16 & -13 \end{pmatrix} \quad \text{and} \quad \det\left(\begin{pmatrix} 11 & -9 \\ 16 & -13 \end{pmatrix}\right) = 11 \cdot (-13) - 16 \cdot (-9) = 1.$$

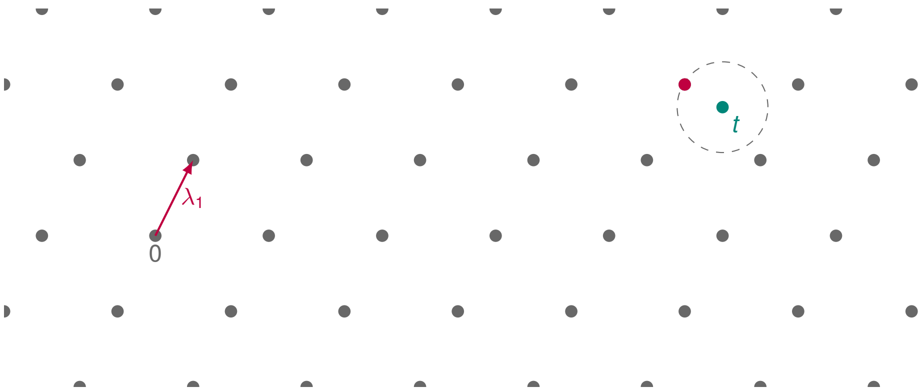
2. Intersection of lattices
3. Computing a short vector of a lattice
4. Computing a lattice vector close to some target

Some Algorithmic Problems on Lattices

1. Testing the equality (or inclusion) of lattices easy
2. Intersection of lattices easy
3. Computing a short vector of a lattice hard
4. Computing a lattice vector close to some target hard

(Hard) Lattice Problems

Shortest Vector Problem (SVP) | Closest Vector Problem (CVP)



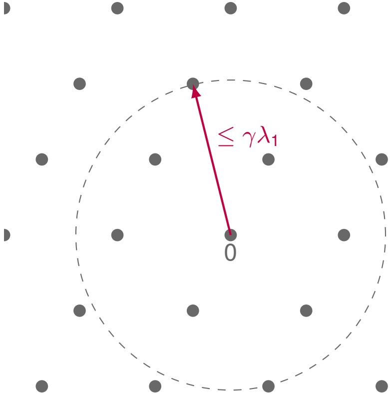
Input: HNF basis of $\mathcal{L}$

Input: HNF basis of $\mathcal{L}$ and target t

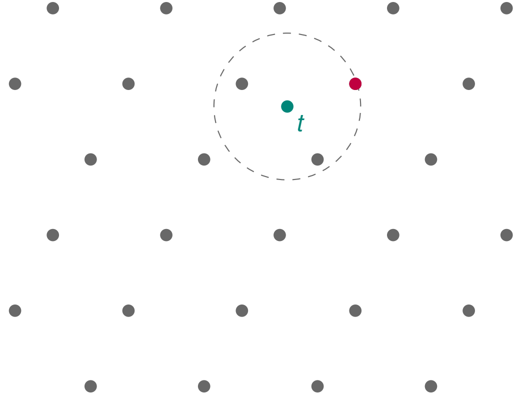
Supposedly hard to solve when n is large (even with a quantum computer)

(Hard) Approximate Lattice Problems

Approximate SVP



Approximate CVP



Supposedly hard to solve when n is large and when the approximation factor is small.

Learning with Errors

- **Parameters** dimension n, $\mathbb{Z}/q\mathbb{Z}$ and error distribution χ_α (often Gaussian)
- **Search** Find a *secret* $s \in (\mathbb{Z}/q\mathbb{Z})^n$ given many "noisy inner products", i.e.

$$
\begin{aligned}
a_1 &\xleftarrow{\$} (\mathbb{Z}/q\mathbb{Z})^n, & b_1 &= \langle a_1, s\rangle + e_1 \in \mathbb{Z}/q\mathbb{Z} \\
a_2 &\xleftarrow{\$} (\mathbb{Z}/q\mathbb{Z})^n, & b_2 &= \langle a_2, s\rangle + e_2 \in \mathbb{Z}/q\mathbb{Z} \\
&\vdots \\
a_m &\xleftarrow{\$} (\mathbb{Z}/q\mathbb{Z})^n, & b_m &= \langle a_m, s\rangle + e_m \in \mathbb{Z}/q\mathbb{Z}
\end{aligned}
$$

Learning with Errors

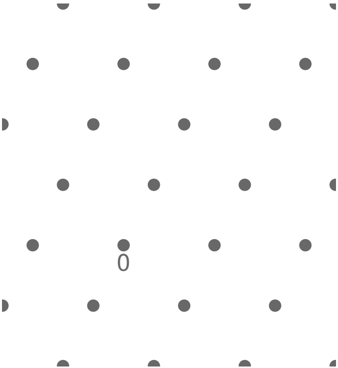
- **Parameters** dimension n, $\mathbb{Z}/q\mathbb{Z}$ and error distribution χ_α (often Gaussian)
- **Search** Find a *secret* $s \in (\mathbb{Z}/q\mathbb{Z})^n$ given many "noisy inner products", i.e.

$$b = A \cdot s + e \mod q$$

LWE as a Lattice Problem

LWE

Given a random matrix $A \in (\mathbb{Z}/q\mathbb{Z})^{m \times n}$ and the vector $b := As + e \in (\mathbb{Z}/q\mathbb{Z})^m$ where each coordinate e_i is chosen independently following a distribution χ_α, recover s or e.

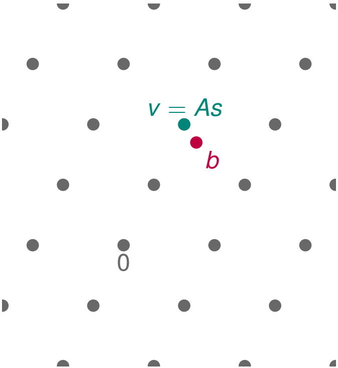


$$\mathcal{L} := \left\{ x \in \mathbb{Z}^n \mid s \in \mathbb{Z}^n,\ As = x \mod q \right\}$$

LWE as a Lattice Problem

LWE

Given a random matrix $A \in (\mathbb{Z}/q\mathbb{Z})^{m \times n}$ and the vector $b := As + e \in (\mathbb{Z}/q\mathbb{Z})^m$ where each coordinate e_i is chosen independently following a distribution χ_α, recover s or e.



$$\mathcal{L} := \{x \in \mathbb{Z}^n \mid s \in \mathbb{Z}^n,\ As = x \mod q\}$$

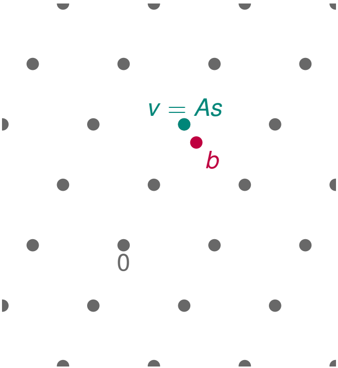
$$b = v + e,$$

where $v \in \mathcal{L}$ and e small.

LWE as a Lattice Problem

LWE

Given a random matrix $A \in (\mathbb{Z}/q\mathbb{Z})^{m\times n}$ and the vector $b := As + e \in (\mathbb{Z}/q\mathbb{Z})^m$ where each coordinate e_i is chosen independently following a distribution χ_α, recover s or e.



$$\mathcal{L} := \{x \in \mathbb{Z}^n \mid s \in \mathbb{Z}^n,\ As = x \mod q\}$$

$$b = v + e,$$

where $v \in \mathcal{L}$ and e small.

LWE $\approx$ CVP in $\mathcal{L}$

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$
  - Approximating nearest codeword problem is as hard as quantumly approximating worst-case lattice problems

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$
  - Approximating nearest codeword problem is as hard as quantumly approximating worst-case lattice problems
- In lattices we use the euclidean distance (L^2 norm)

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$
  - Approximating nearest codeword problem is as hard as quantumly approximating worst-case lattice problems
- In lattices we use the euclidean distance (L^2 norm)
  - Reducing the LWE problem from L^2 to L^1 does not reduce the security (still NP hard).

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$
  - Approximating nearest codeword problem is as hard as quantumly approximating worst-case lattice problems
- In lattices we use the euclidean distance (L^2 norm)
  - Reducing the LWE problem from L^2 to L^1 does not reduce the security (still NP hard).
  - The Lee metric can be interpreted as the L^1 norm modulo q

Concluding Remarks

- The LWE problem can equivalently be presented as the problem of decoding random linear codes.
  - The Hamming metric of the error vector follows from $\chi_\alpha(0)$
  - Approximating nearest codeword problem is as hard as quantumly approximating worst-case lattice problems
- In lattices we use the euclidean distance (L^2 norm)
  - Reducing the LWE problem from L^2 to L^1 does not reduce the security (still NP hard).
  - The Lee metric can be interpreted as the L^1 norm modulo q
- As n grows large, sampling an error term e of given Lee weight uniformly at random yields an exponential distribution for the entries of e.

Research Questions

- Defining codes over lattices what can we deduce from the Lee metric knowledge and LWE to coding theory?
- Does LWE in the Lee metric help to understand the limits of ISD?

Thank you for you attention.