Lattice-Based Cryptography

Criptografía basada en retículos

where to start and where to go next

Katharina Boudgoust

until 2023
Postdoc
Aarhus University
Denmark

from 2024
Researcher CNRS
LIRMM Montpellier
France

Overview of Today's Lecture

Questions we are trying to answer today:

- Part 1: *What are lattices?*
- Part 2: *What are lattice problems?*
- Part 3: *What is lattice-based cryptography?*

where to start

- Part 4: *What are the current challenges?*

whete to go next

References:

- Crash Course Spring 2022 [lecture notes]
- The Lattice Club [link]

Context

👉 The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:

- Discrete logarithm $\rightarrow$ Arantxa's proof system
- Factoring

Context

The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:

- ~~Discrete logarithm~~ → Arantxa's proof system
- ~~Factoring~~

⚠ ∃ poly-time quantum algorithm [Sho97]

Quantum-resistant candidates:

- Codes
- Lattices
- Isogenies
- Multivariate systems
- **?**

Context

The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:

- ~~Discrete logarithm~~ → Arantxa's proof system
- ~~Factoring~~

⚠ ∃ poly-time quantum algorithm [Sho97]

Quantum-resistant candidates:

- Codes
- Lattices → now
- Isogenies → later with Chloe
- Multivariate systems
- **?**

Fernando (INCA)

US National Institute of Standards and Technology (NIST) Project

- 2016: start of NIST's post-quantum cryptography project[*]
- 2022: selection of 4 schemes, 3 of them relying on lattice problems

Public Key Encryption:

- Kyber

Digital Signature:

- Dilithium
- Falcon
- SPHINCS+

Lattice-based cryptography plays a leading role in designing post-quantum cryptography.

[*] https://csrc.nist.gov/projects/post-quantum-cryptography

Part 1:

What is a lattice?

Euclidean Lattices

An Euclidean lattice Λ is a discrete additive subgroup of $\mathbb{R}^n$.

Euclidean Lattices

An Euclidean lattice Λ is a discrete additive subgroup of $\mathbb{R}^n$.

- additive subgroup: $\mathbf{0} \in \Lambda$, and for all $\mathbf{x}, \mathbf{y} \in \Lambda$ it holds $\mathbf{x} + \mathbf{y}, -\mathbf{x} \in \Lambda$;
- discrete: every $\mathbf{x} \in \Lambda$ has a neighborhood in which $\mathbf{x}$ is the only lattice point.
  $\exists \varepsilon > 0$ such that $\mathcal{B}(\mathbf{x}, \varepsilon) \cap \Lambda = \{ \mathbf{x} \}$

Euclidean Lattices

An Euclidean lattice Λ is a discrete additive subgroup of $\mathbb{R}^n$.

- additive subgroup: $\mathbf{0} \in \Lambda$, and for all $\mathbf{x}, \mathbf{y} \in \Lambda$ it holds $\mathbf{x} + \mathbf{y}, -\mathbf{x} \in \Lambda$;
- discrete: every $\mathbf{x} \in \Lambda$ has a neighborhood in which $\mathbf{x}$ is the only lattice point.
  $\exists \varepsilon > 0$ such that $\mathcal{B}(\mathbf{x}, \varepsilon) \cap \Lambda = \{ \mathbf{x} \}$

There exists a finite basis $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \subset \mathbb{R}^n$ such that

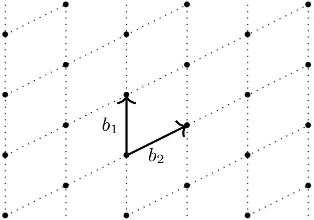
$$\Lambda(\mathbf{B}) = \left\{ \sum_{i=1}^{n} z_i \mathbf{b}_i \colon z_i \in \mathbb{Z} \right\}.$$

- n is the rank of Λ

Euclidean Lattices

Let $\mathbf{B} \in \mathbb{R}^{n \times n}$ be a basis for Λ, i.e.,

$$\Lambda(\mathbf{B}) = \left\{ \sum_{i=1}^{n} z_i \mathbf{b}_i \colon z_i \in \mathbb{Z} \right\} = \{ \mathbf{B}\mathbf{z} \colon \mathbf{z} \in \mathbb{Z}^n \} .$$

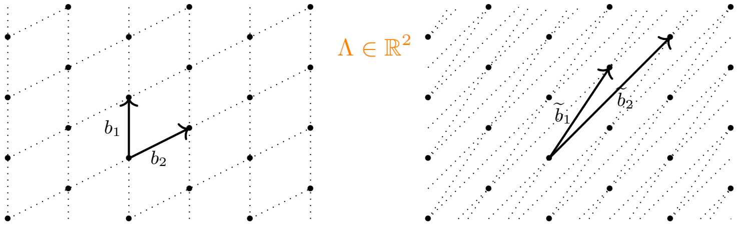


$\Lambda \in \mathbb{R}^2$

Euclidean Lattices

Let $\mathbf{B} \in \mathbb{R}^{n \times n}$ be a basis for Λ, i.e.,

$$\Lambda(\mathbf{B}) = \left\{ \sum_{i=1}^{n} z_i \mathbf{b}_i \colon z_i \in \mathbb{Z} \right\} = \{\mathbf{B}\mathbf{z} \colon \mathbf{z} \in \mathbb{Z}^n\}.$$



- $\mathbf{U} \in \mathbb{Z}^{n \times n}$ unimodular, then $\widetilde{\mathbf{B}} = \mathbf{B} \cdot \mathbf{U}$ also a basis of Λ $\quad$ $\det(\mathbf{U}) = \pm 1$
- $\det(\Lambda) := |\det(\mathbf{B})|$

Dual Lattices

The **dual** of a lattice $\Lambda \subset \mathbb{R}^n$ is defined as

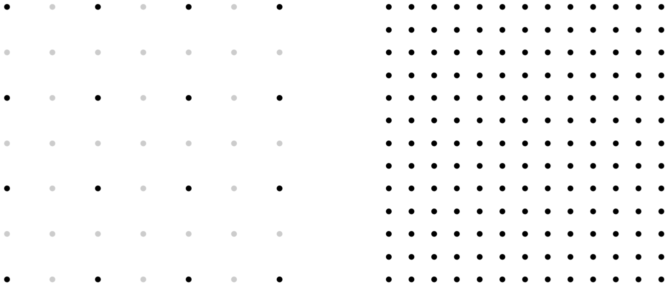
$$\Lambda^\vee = \{\mathbf{w} \in \mathbb{R}^n \colon \langle \mathbf{w}, \mathbf{x} \rangle \in \mathbb{Z} \ \forall \mathbf{x} \in \Lambda\}\,.$$

Dual Lattices

The **dual** of a lattice $\Lambda \subset \mathbb{R}^n$ is defined as

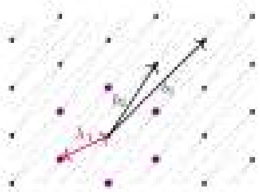
$$\Lambda^\vee = \{\mathbf{w} \in \mathbb{R}^n \colon \langle \mathbf{w}, \mathbf{x} \rangle \in \mathbb{Z} \ \forall \mathbf{x} \in \Lambda\}\,.$$

- if $\mathbf{B}$ a basis for Λ, then $(\mathbf{B}^T)^{-1}$ a basis for $\Lambda^\vee$
- $\det(\Lambda^\vee) = \det(\Lambda)^{-1}$



$2\mathbb{Z}^2$ and its dual $\frac{1}{2}\mathbb{Z}^2$

Lattice Minimum & Special Lattices

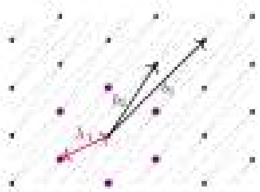


The **minimum** of a lattice $\Lambda \subset \mathbb{R}^n$ is defined as

$$\lambda_1(\Lambda) = \min_{\mathbf{x} \in \Lambda \setminus \{\mathbf{0}\}} \|\mathbf{x}\|_2.$$

- Minkowski: $\lambda_1(\Lambda) \leq \sqrt{n} \cdot \det(\Lambda)^{1/n}$
- Exercise: $\lambda_1(\Lambda) \cdot \lambda_1(\Lambda^\vee) \leq n$

Lattice Minimum & Special Lattices



The **minimum** of a lattice $\Lambda \subset \mathbb{R}^n$ is defined as

$$\lambda_1(\Lambda) = \min_{\mathbf{x} \in \Lambda \setminus \{\mathbf{0}\}} \|\mathbf{x}\|_2.$$

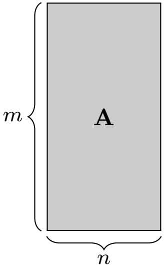
- Minkowski: $\lambda_1(\Lambda) \leq \sqrt{n} \cdot \det(\Lambda)^{1/n}$
- Exercise: $\lambda_1(\Lambda) \cdot \lambda_1(\Lambda^\vee) \leq n$

Let $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ for some $n, m, q \in \mathbb{N}$ with $n \leq m$

$\mathbb{Z}_q$ integers modulo q

$$\Lambda_q(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m \colon \mathbf{y} = \mathbf{A}\mathbf{s} \bmod q \text{ for some } \mathbf{s} \in \mathbb{Z}^n\}$$
$$\Lambda_q^\perp(\mathbf{A}) = \left\{\mathbf{y} \in \mathbb{Z}^m \colon \mathbf{A}^T\mathbf{y} = \mathbf{0} \bmod q\right\}$$

- Exercise: $\Lambda_q^\perp(\mathbf{A}) = q \cdot \Lambda_q(\mathbf{A})^\vee$



Part 2:

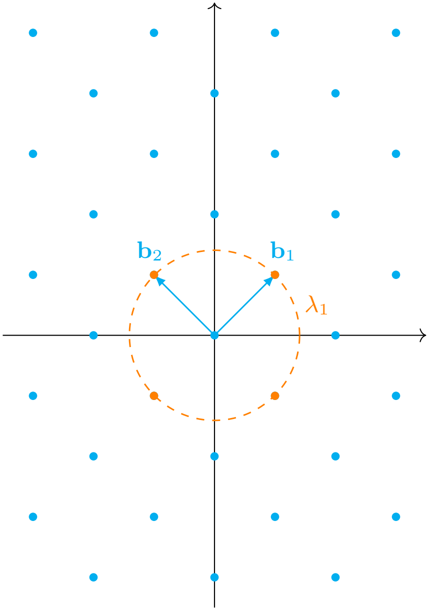
What are lattice problems?

Shortest Vector Problem

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n.

The **shortest vector problem** (SVP) asks to find a vector $\mathbf{w} \in \Lambda$ such that

$$\|\mathbf{w}\|_2 = \lambda_1(\Lambda).$$

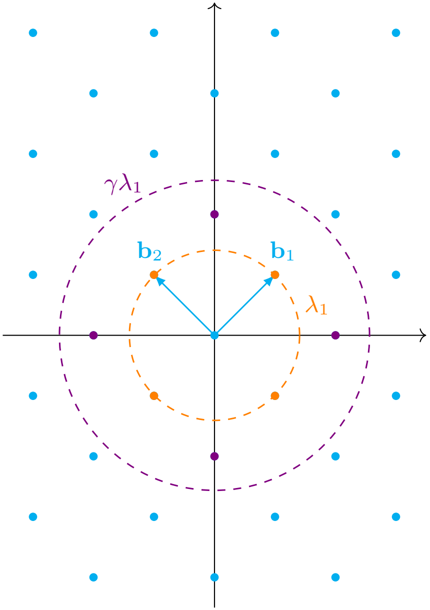


Shortest Vector Problem

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n.

The **approximate shortest vector problem** (SVP$_\gamma$) for $\gamma \geq 1$ asks to find a vector $\mathbf{w} \in \Lambda$ such that

$$\|\mathbf{w}\|_2 \leq \gamma\lambda_1(\Lambda).$$



Shortest Vector Problem

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n.

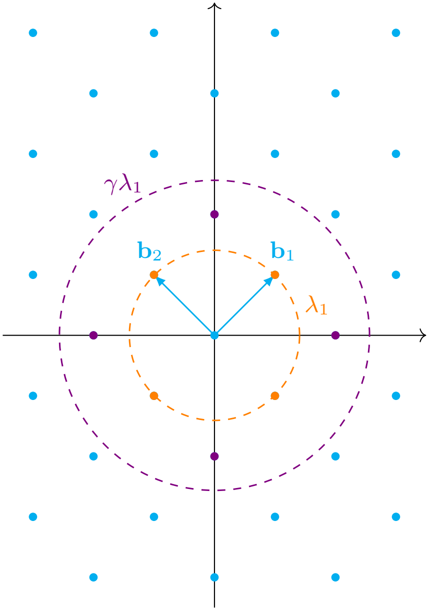
The **approximate shortest vector problem** (SVP$_\gamma$) for $\gamma \geq 1$ asks to find a vector $\mathbf{w} \in \Lambda$ such that

$$\|\mathbf{w}\|_2 \leq \gamma\lambda_1(\Lambda).$$

The complexity of SVP$_\gamma$ increases with n, but decreases with γ.

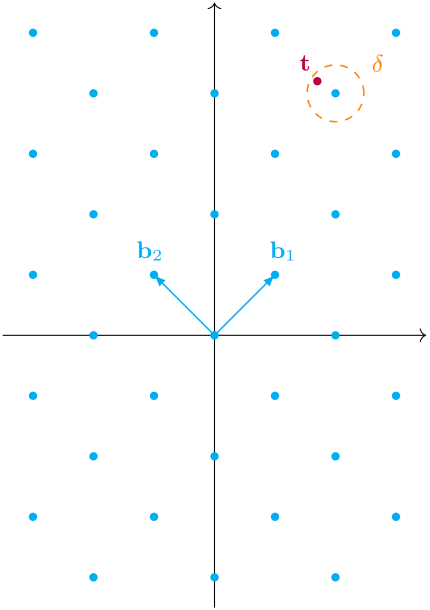
Conjecture:

There is no polynomial-time classical or quantum algorithm that solves SVP$_\gamma$ to within polynomial factors.



Bounded Distance Decoding

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n and a target $\mathbf{t} \in \mathbb{R}^n$ such $\mathrm{dist}(\Lambda, \mathbf{t}) \leq \delta < \lambda_1(\Lambda)$.

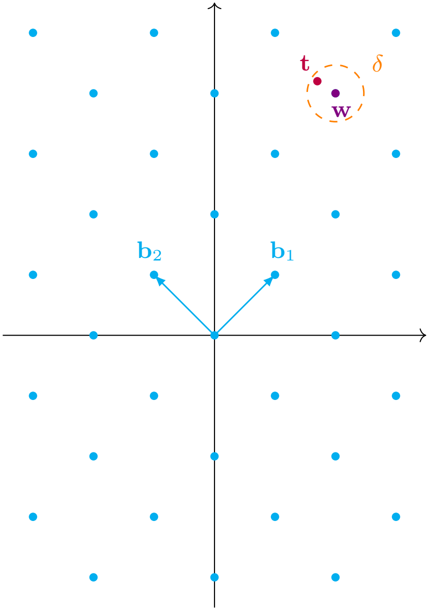


Bounded Distance Decoding

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n and a target $\mathbf{t} \in \mathbb{R}^n$ such $\mathrm{dist}(\Lambda, \mathbf{t}) \leq \delta < \lambda_1(\Lambda)$.

The **bounded distance decoding** (BDD_δ) problem asks to find the unique vector $\mathbf{w} \in \Lambda$ such that

$$\|\mathbf{w} - \mathbf{t}\|_2 \leq \delta.$$



Bounded Distance Decoding

Given a lattice $\Lambda \in \mathbb{R}^n$ of rank n and a target $\mathbf{t} \in \mathbb{R}^n$ such $\text{dist}(\Lambda, \mathbf{t}) \leq \delta < \lambda_1(\Lambda)$.

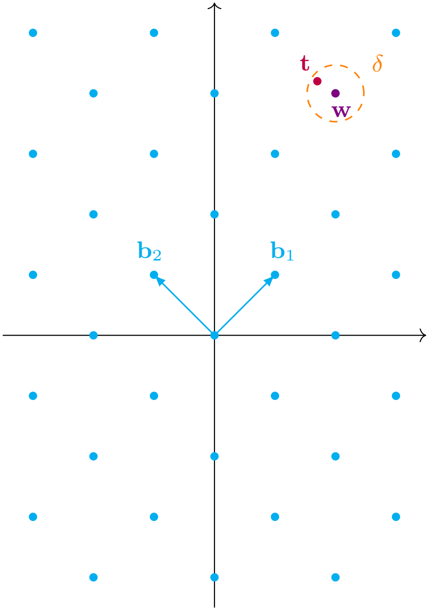
The **bounded distance decoding** (BDD_δ) problem asks to find the unique vector $\mathbf{w} \in \Lambda$ such that

$$\|\mathbf{w} - \mathbf{t}\|_2 \leq \delta.$$

The complexity of BDD_δ increases with n and with δ.

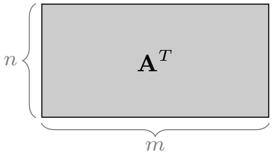
Conjecture:

There is no polynomial-time classical or quantum algorithm that solves BDD_δ to within polynomial factors.



Short Integer Solution [Ajt96]

Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ sampled uniformly at random and bound $\beta > 0$.

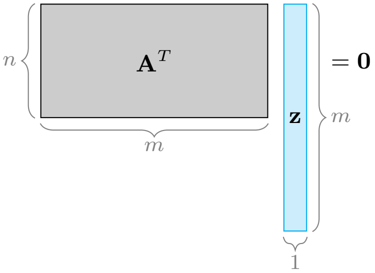


Short Integer Solution [Ajt96]

Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ sampled uniformly at random and bound $\beta > 0$.

The **short integer solution** (SIS$_\beta$) problem asks to find a vector $\mathbf{z} \in \mathbb{Z}^m$ of norm $0 < \|\mathbf{z}\|_2 \leq \beta$ such that

$$\mathbf{A}^T \mathbf{z} = \mathbf{0} \bmod q.$$



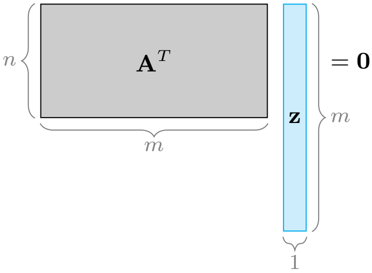
Short Integer Solution [Ajt96]

Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ sampled uniformly at random and bound $\beta > 0$.

The **short integer solution** (SIS$_\beta$) problem asks to find a vector $\mathbf{z} \in \mathbb{Z}^m$ of norm $0 < \|\mathbf{z}\|_2 \leq \beta$ such that

$$\mathbf{A}^T \mathbf{z} = \mathbf{0} \bmod q.$$

⚠ The norm restriction makes it a hard problem!

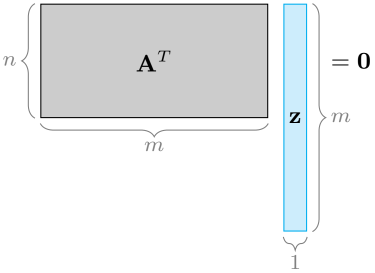


Short Integer Solution [Ajt96]

Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ sampled uniformly at random and bound $\beta > 0$.

The **short integer solution** (SIS$_\beta$) problem asks to find a vector $\mathbf{z} \in \mathbb{Z}^m$ of norm $0 < \|\mathbf{z}\|_2 \leq \beta$ such that

$$\mathbf{A}^T \mathbf{z} = \mathbf{0} \bmod q.$$



⚠ The norm restriction makes it a hard problem!

Recall:

$$\Lambda_q^{\perp}(\mathbf{A}) = \left\{ \mathbf{y} \in \mathbb{Z}^m : \mathbf{A}^T \mathbf{y} = \mathbf{0} \bmod q \right\}$$

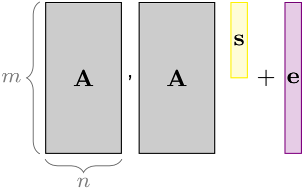
☞ SIS$_\beta$ equals SVP$_\gamma$ in the special lattice $\Lambda_q^{\perp}(\mathbf{A})$ for $\beta = \gamma \cdot \lambda_1(\Lambda_q^{\perp}(\mathbf{A}))$

Learning With Errors [Reg05]

Given a matrix $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{m \times n})$.

Given a vector $\mathbf{b} \in \mathbb{Z}_q^m$, where $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$ for

- secret $\mathbf{s} \in \mathbb{Z}_q^n$ sampled from distribution D_s and
- noise/error $\mathbf{e} \in \mathbb{Z}^m$ sampled from distribution D_e such that $\|\mathbf{e}\|_2 \leq \delta \ll q$.

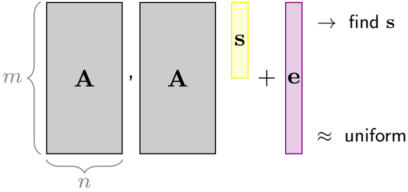


Learning With Errors [Reg05]

Given a matrix $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{m\times n})$.

Given a vector $\mathbf{b} \in \mathbb{Z}_q^m$, where $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$ for

- secret $\mathbf{s} \in \mathbb{Z}_q^n$ sampled from distribution D_s and
- noise/error $\mathbf{e} \in \mathbb{Z}^m$ sampled from distribution D_e such that $\|\mathbf{e}\|_2 \leq \delta \ll q$.



Search learning with errors (S-LWE$_\delta$) asks to find $\mathbf{s}$.

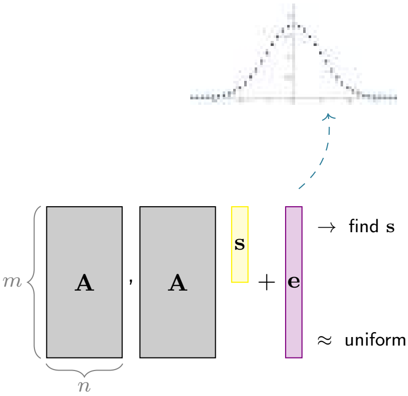
Decision learning with errors (D-LWE$_\delta$) asks to distinguish $(\mathbf{A}, \mathbf{b})$ from the uniform distribution over $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$.

Learning With Errors [Reg05]

Given a matrix $\mathbf{A} \leftarrow \text{Unif}(\mathbb{Z}_q^{m\times n})$.

Given a vector $\mathbf{b} \in \mathbb{Z}_q^m$, where $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$ for

- secret $\mathbf{s} \in \mathbb{Z}_q^n$ sampled from distribution D_s and
- noise/error $\mathbf{e} \in \mathbb{Z}^m$ sampled from distribution D_e such that $\|\mathbf{e}\|_2 \leq \delta \ll q$.



Search learning with errors (S-LWE$_\delta$) asks to find $\mathbf{s}$.

Decision learning with errors (D-LWE$_\delta$) asks to distinguish $(\mathbf{A}, \mathbf{b})$ from the uniform distribution over $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$.

⚠ The present noise makes S-LWE a hard problem.

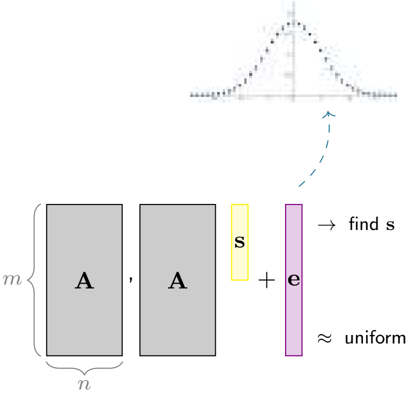
⚠ The norm restriction on $\mathbf{e}$ makes D-LWE a hard problem!

Learning With Errors [Reg05]

Given a matrix $\mathbf{A} \leftarrow \text{Unif}(\mathbb{Z}_q^{m\times n})$.

Given a vector $\mathbf{b} \in \mathbb{Z}_q^m$, where $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$ for

- secret $\mathbf{s} \in \mathbb{Z}_q^n$ sampled from distribution D_s and
- noise/error $\mathbf{e} \in \mathbb{Z}^m$ sampled from distribution D_e such that $\|\mathbf{e}\|_2 \leq \delta \ll q$.



Search learning with errors (S-LWE$_\delta$) asks to find $\mathbf{s}$.

Decision learning with errors (D-LWE$_\delta$) asks to distinguish $(\mathbf{A}, \mathbf{b})$ from the uniform distribution over $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$.

⚠ The present noise makes S-LWE a hard problem.

⚠ The norm restriction on $\mathbf{e}$ makes D-LWE a hard problem!

Exercise: S-LWE$_\delta$ equals BDD$_\delta$ in the special lattice $\Lambda_q(\mathbf{A})$.

Connection between LWE and SIS

If there is an efficient solver for SIS_β, then there is an efficient solver for $\mathsf{D\text{-}LWE}_\delta$, assuming $\delta \cdot \beta \ll q$.

Connection between LWE and SIS

If there is an efficient solver for SIS_β, then there is an efficient solver for $\mathsf{D\text{-}LWE}_\delta$, assuming $\delta \cdot \beta \ll q$.

Proof.

Given $(\mathbf{A}, \mathbf{b})$, our goal is to decide whether 1) $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$ for $\|\mathbf{e}\|_2 \leq \delta$ or 2) $\mathbf{b} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^m)$.

□

Connection between LWE and SIS

If there is an efficient solver for SIS_β, then there is an efficient solver for D-LWE_δ, assuming $\delta \cdot \beta \ll q$.

Proof.

Given $(\mathbf{A}, \mathbf{b})$, our goal is to decide whether 1) $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$ for $\|\mathbf{e}\|_2 \leq \delta$ or
2) $\mathbf{b} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^m)$.

Forward $\mathbf{A}$ to SIS-solver and receive back $\mathbf{z}$ such that $\mathbf{A}^T\mathbf{z} = \mathbf{0} \bmod q$ and $\|\mathbf{z}\|_2 \leq \beta$.

$\square$

Connection between LWE and SIS

If there is an efficient solver for SIS_β, then there is an efficient solver for D-LWE_δ, assuming $\delta \cdot \beta \ll q$.

Proof.

Given $(\mathbf{A}, \mathbf{b})$, our goal is to decide whether 1) $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$ for $\|\mathbf{e}\|_2 \leq \delta$ or 2) $\mathbf{b} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^m)$.

Forward $\mathbf{A}$ to SIS-solver and receive back $\mathbf{z}$ such that $\mathbf{A}^T\mathbf{z} = \mathbf{0} \bmod q$ and $\|\mathbf{z}\|_2 \leq \beta$.

Compute $\left\|\mathbf{b}^T\mathbf{z}\right\|_\infty$. If the norm is $\ll q$, claim that we are in case 1). Else, claim that we are in case 2).

$\square$

Connection between LWE and SIS

If there is an efficient solver for SIS_β, then there is an efficient solver for D-LWE_δ, assuming $\delta \cdot \beta \ll q$.

Proof.

Given $(\mathbf{A}, \mathbf{b})$, our goal is to decide whether 1) $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$ for $\|\mathbf{e}\|_2 \leq \delta$ or 2) $\mathbf{b} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^m)$.

Forward $\mathbf{A}$ to SIS-solver and receive back $\mathbf{z}$ such that $\mathbf{A}^T\mathbf{z} = \mathbf{0} \bmod q$ and $\|\mathbf{z}\|_2 \leq \beta$.

Compute $\left\|\mathbf{b}^T\mathbf{z}\right\|_\infty$. If the norm is $\ll q$, claim that we are in case 1). Else, claim that we are in case 2).

Case 1) $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}$, thus $\mathbf{b}^T\mathbf{z} = \mathbf{s}^T\mathbf{A}^T\mathbf{z} + \mathbf{e}^T\mathbf{z} = \mathbf{e}^T\mathbf{z} \bmod q$. Thus $\left\|\mathbf{b}^T\mathbf{z}\right\|_\infty \leq \left\|\mathbf{e}^T\right\|_\infty \cdot \|\mathbf{z}\|_\infty \leq \delta \cdot \beta \ll q$.

Case 2) $\mathbf{b}$ uniform, so is $\mathbf{b}^T\mathbf{z}$ and hence $\left\|\mathbf{b}^T\mathbf{z}\right\|_\infty$ with high chances larger than $\delta\beta$. □

Part 3:

What is lattice-based cryptography?

Collision-Resistant Hash Function from SIS [Ajt96]

A function $f\colon$ Domain $\to$ Range is called **collision-resistant** if it is hard to output two elements $\mathbf{x}, \mathbf{x}' \in$ Domain such that

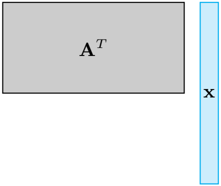
$$f(\mathbf{x}) = f(\mathbf{x}') \text{ and } \mathbf{x} \neq \mathbf{x}'.$$

Collision-Resistant Hash Function from SIS [Ajt96]

A function $f\colon \text{Domain} \to \text{Range}$ is called **collision-resistant** if it is hard to output two elements $\mathbf{x}, \mathbf{x}' \in \text{Domain}$ such that

$$f(\mathbf{x}) = f(\mathbf{x}') \text{ and } \mathbf{x} \neq \mathbf{x}'.$$

Set $f_{\mathbf{A}}\colon \{0,1\}^m \to \mathbb{Z}_q^n$ with $f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}^T\mathbf{x} \bmod q$ for $\mathbf{A} \leftarrow \text{Unif}(\mathbb{Z}_q^{m\times n})$.

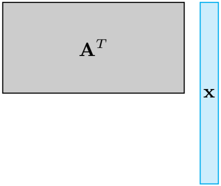


Collision-Resistant Hash Function from SIS [Ajt96]

A function $f\colon \text{Domain} \to \text{Range}$ is called **collision-resistant** if it is hard to output two elements $\mathbf{x}, \mathbf{x}' \in \text{Domain}$ such that

$$f(\mathbf{x}) = f(\mathbf{x}') \text{ and } \mathbf{x} \neq \mathbf{x}'.$$

Set $f_{\mathbf{A}}\colon \{0,1\}^m \to \mathbb{Z}_q^n$ with $f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}^T\mathbf{x} \bmod q$ for $\mathbf{A} \leftarrow \text{Unif}(\mathbb{Z}_q^{m\times n})$.



Exercise: Assuming SIS is hard to solve for $\beta = \sqrt{m}$, then $f_{\mathbf{A}}$ is collision-resistant

Hint: $\mathbf{x} \neq \mathbf{x}' \in \{0,1\}^m \Leftrightarrow \mathbf{0} \neq \mathbf{x} - \mathbf{x}' \in \{-1,0,1\}^m$

$\mathbf{A}^T\mathbf{x} = \mathbf{A}^T\mathbf{x}' \Leftrightarrow \mathbf{A}^T(\mathbf{x} - \mathbf{x}') = 0$

Reminder: Public-Key Encryption (PKE)

A public-key encryption scheme $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ consists of three algorithms:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$ — λ security parameter
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = m'$

Correctness: $\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = m$ during an honest execution

Semantic Security: $\mathsf{Enc}(\mathsf{pk}, m_0)$ is indistinguishable from $\mathsf{Enc}(\mathsf{pk}, m_1)$
(IND-CPA)

Public-Key Encryption from LWE [Reg05]

Let χ be distribution on $\mathbb{Z}$.

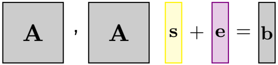
- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$

$$\boxed{\mathbf{A}}\ ,\ \boxed{\mathbf{A}}\,\boxed{\mathbf{s}} + \boxed{\mathbf{e}} = \boxed{\mathbf{b}}$$

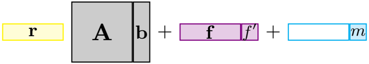
Public-Key Encryption from LWE [Reg05]

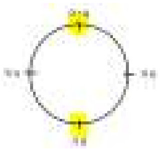
Let χ be distribution on $\mathbb{Z}$.

- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n \times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$



- $\mathsf{Enc}(\mathsf{pk}, m \in \{0,1\})$:
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output $\mathsf{ct} = (\mathbf{u}, v)$

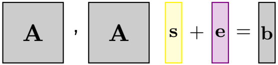




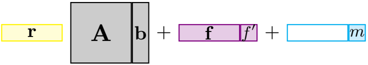
Public-Key Encryption from LWE [Reg05]

Let χ be distribution on $\mathbb{Z}$.

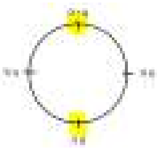
- KGen(1^λ):
  - $\mathbf{A} \leftarrow$ Unif$(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output sk $= \mathbf{s}$ and pk $= (\mathbf{A}, \mathbf{b})$



- Enc(pk, $m \in \{0,1\}$):
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output ct $= (\mathbf{u}, v)$

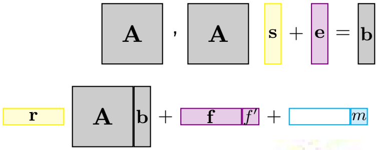


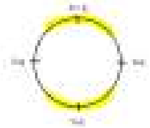
- Dec(sk, ct):
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$



Public-Key Encryption from LWE [Reg05]

- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$
- $\mathsf{Enc}(\mathsf{pk}, m \in \{0,1\})$:
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rceil \cdot m$
  - Output $\mathsf{ct} = (\mathbf{u}, v)$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$





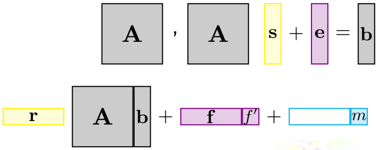
Correctness:

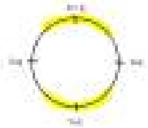
$$v - \mathbf{u}\mathbf{s} = \mathbf{r}(\mathbf{A}\mathbf{s} + \mathbf{e}) + f' + \lfloor q/2 \rceil \cdot m - (\mathbf{r}\mathbf{A} + \mathbf{f})\mathbf{s}$$
$$= \underbrace{\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}}_{*\ \text{ciphertext noise}} + \lfloor q/2 \rceil m$$

Decryption succeeds if $|*| < q/8$

Public-Key Encryption from LWE [Reg05]

- KGen(1^λ):
  - $\mathbf{A} \leftarrow$ Unif$(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output sk $= \mathbf{s}$ and pk $= (\mathbf{A}, \mathbf{b})$
- Enc(pk, $m \in \{0,1\}$):
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output ct $= (\mathbf{u}, v)$
- Dec(sk, ct):
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$





Correctness: Let χ be B-bounded with $2nB^2 + B < q/8$

$$v - \mathbf{u}\mathbf{s} = \mathbf{r}(\mathbf{A}\mathbf{s} + \mathbf{e}) + f' + \lfloor q/2 \rfloor \cdot m - (\mathbf{r}\mathbf{A} + \mathbf{f})\mathbf{s}$$

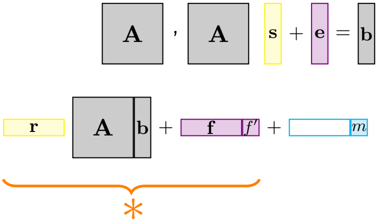
$$= \underbrace{\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}}_{*\ \text{ciphertext noise}} + \lfloor q/2 \rfloor m$$

Decryption succeeds if $|*| < q/8$

$$|*| = |\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}| \leq \|\mathbf{r}\|_2 \cdot \|\mathbf{e}\|_2 + \|\mathbf{f}\|_2 \cdot \|\mathbf{s}\|_2 + |f'| \leq 2(\sqrt{n}B \cdot \sqrt{n}B) + B < q/8$$

Public-Key Encryption from LWE [Reg05]

- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$
- $\mathsf{Enc}(\mathsf{pk}, m \in \{0,1\})$:
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output $\mathsf{ct} = (\mathbf{u}, v)$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$



Semantic Security: Assume hardness of decision LWE

1. replace $\mathbf{b}$ by uniform random vector
2. replace non-message part ($*$) by uniform random vector
3. then the message is completely hidden

Kyber - Selected for Standardization by NIST

Kyber = the previous construction + several improvements

Main improvements:

1. Structured LWE variant (most important)
2. LWE secret and noise from centered binomial distribution
3. Pseudorandomness for distributions
4. Ciphertext compression

Sources:

- Website of Kyber: https://pq-crystals.org/kyber/
- Latest specifications [link]
- Tutorial by V. Lyubashevsky [link]

5 Min

Part 4:

What are (my) current challenges?

Re-Reminder: Public Key Encryption (PKE)

PKE scheme:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$ — λ security parameter
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to m'$

Properties:

- Correctness
- Semantic security

Re-Reminder: Public Key Encryption (PKE)

PKE scheme:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$ — λ security parameter
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to m'$

Properties:

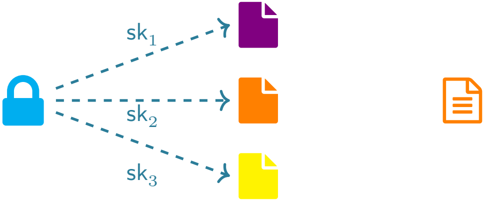
- Correctness
- Semantic security

Single Point of Failure

Threshold Public Key Encryption (TPKE)

t-out-of-n Threshold PKE scheme:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_n)$ secret sharing
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{PartDec}(\mathsf{sk}_i, \mathsf{ct}') \to d_i$ $i \in \{1, \ldots, n\}$
- $\mathsf{Combine}(\{d_i\}_{i \in S}) \to m'$ $S \subset \{1, \ldots, n\}$

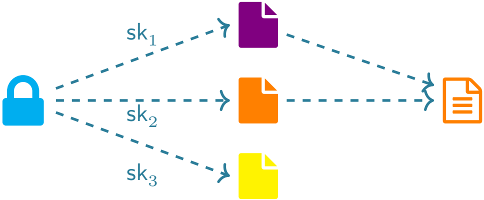


*https://csrc.nist.gov/projects/threshold-cryptography

Threshold Public Key Encryption (TPKE)

t-out-of-n Threshold PKE scheme:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_n)$ secret sharing
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{PartDec}(\mathsf{sk}_i, \mathsf{ct}') \to d_i$ $\quad i \in \{1, \ldots, n\}$
- $\mathsf{Combine}(\{d_i\}_{i \in S}) \to m'$ $\quad S \subset \{1, \ldots, n\}$



*https://csrc.nist.gov/projects/threshold-cryptography

Threshold Public Key Encryption (TPKE)

t-out-of-n Threshold PKE scheme:

- $\mathsf{KGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_n)$ — secret sharing
- $\mathsf{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$
- $\mathsf{PartDec}(\mathsf{sk}_i, \mathsf{ct}') \to d_i$ — $i \in \{1, \ldots, n\}$
- $\mathsf{Combine}(\{d_i\}_{i \in S}) \to m'$ — $S \subset \{1, \ldots, n\}$

Properties:

- Correctness — for $|S| > t$ recover correct message
- Partial decryption security — for $|S| \leq t$ no information is leaked
- Semantic security

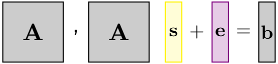
Applications:

- Storing sensitive data — NIST's call*
- Electronic voting protocols
- Multiparty computations — $\to$ Chris yesterday, Daniel later

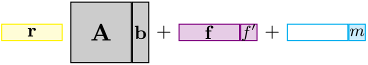
* https://csrc.nist.gov/projects/threshold-cryptography

Reminder: PKE from LWE

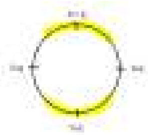
- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$



- $\mathsf{Enc}(\mathsf{pk}, m \in \{0,1\})$:
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output $\mathsf{ct} = (\mathbf{u}, v)$

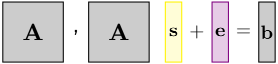


- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$

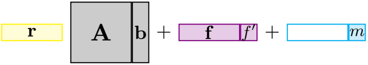


Reminder: PKE from LWE

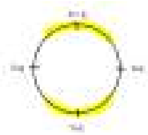
- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - Output $\mathsf{sk} = \mathbf{s}$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$



- $\mathsf{Enc}(\mathsf{pk}, m \in \{0,1\})$:
  - $\mathbf{r}, \mathbf{f} \leftarrow \chi^n$ and $f' \leftarrow \chi$
  - $\mathbf{u} = \mathbf{r}\mathbf{A} + \mathbf{f}$
  - $v = \mathbf{r}\mathbf{b} + f' + \lfloor q/2 \rfloor \cdot m$
  - Output $\mathsf{ct} = (\mathbf{u}, v)$



- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$:
  - If $v - \mathbf{u}\mathbf{s}$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$



In order to thresholdize it:

modify KGen and replace Dec by PartDec and Combine

(Enc stays the same)

Full-Threshold PKE from LWE, First Trial

(n-out-of-n)

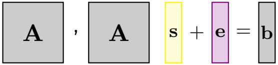
- KGen(1^λ):
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - $\mathbf{s}_1, \dots, \mathbf{s}_{n-1} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^n)$
  - $\mathbf{s}_n = \mathbf{s} - \sum_{i=1}^{n-1} \mathbf{s}_i$
  - Output $\mathsf{sk}_i = \mathbf{s}_i$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$

$$\mathbf{A} \,,\, \mathbf{A}\,\mathbf{s} + \mathbf{e} = \mathbf{b}$$

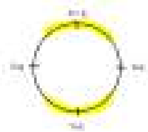
- PartDec$(\mathsf{sk}_i, (\mathbf{u}, v))$:
  - Output $d_i = \mathbf{u}\mathbf{s}_i$
- Combine$(d_1, \dots, d_n)$:
  - $d = \sum_{i=1}^{n} d_i$
  - If $v - d$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$

Full-Threshold PKE from LWE, First Trial

- KGen(1^λ):
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - $\mathbf{s}_1, \ldots, \mathbf{s}_{n-1} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^n)$
  - $\mathbf{s}_n = \mathbf{s} - \sum_{i=1}^{n-1} \mathbf{s}_i$
  - Output $\mathsf{sk}_i = \mathbf{s}_i$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$



- PartDec$(\mathsf{sk}_i, (\mathbf{u}, v))$:
  - Output $d_i = \mathbf{u}\mathbf{s}_i$



- Combine$(d_1, \ldots, d_n)$:
  - $d = \sum_{i=1}^{n} d_i$
  - If $v - d$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$

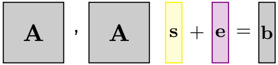
Correctness: given $d_1, \ldots, d_n$

$$v - \sum_{i=1}^{n} \mathbf{u}\mathbf{s}_i = v - \mathbf{u}\sum_{i=1}^{n} \mathbf{s}_i = v - \mathbf{u}\mathbf{s}$$

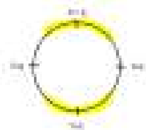
$$= \underbrace{\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}}_{*\text{ ciphertext noise}} + \lfloor q/2 \rceil m$$

Decryption succeeds if $|*| < q/8$

Full-Threshold PKE from LWE, First Trial



- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - $\mathbf{s}_1, \ldots, \mathbf{s}_{n-1} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^n)$
  - $\mathbf{s}_n = \mathbf{s} - \sum_{i=1}^{n-1} \mathbf{s}_i$
  - Output $\mathsf{sk}_i = \mathbf{s}_i$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$
- PartDec($\mathsf{sk}_i, (\mathbf{u}, v)$):
  - Output $d_i = \mathbf{u}\mathbf{s}_i$
- Combine($d_1, \ldots, d_n$):
  - $d = \sum_{i=1}^{n} d_i$
  - If $v - d$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$



Correctness: given $d_1, \ldots, d_n$

$$v - \sum_{i=1}^{n} \mathbf{u}\mathbf{s}_i = v - \mathbf{u}\sum_{i=1}^{n} \mathbf{s}_i = v - \mathbf{u}\mathbf{s}$$

$$= \underbrace{\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}}_{*\ \text{ciphertext noise}} + \lfloor q/2 \rfloor m$$

⚠ But $(*)$ leaks information about $\mathsf{sk} = \mathbf{s}$!

Full-Threshold PKE from LWE [BD10]

- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - $\mathbf{s} = \sum_{i=1}^{n} \mathbf{s}_i$
  - Output $\mathsf{sk}_i = \mathbf{s}_i$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$

$$\boxed{\mathbf{A}}\ ,\ \boxed{\mathbf{A}}\ \boxed{\mathbf{s}} + \boxed{\mathbf{e}} = \boxed{\mathbf{b}}$$

- $\mathsf{PartDec}(\mathsf{sk}_i, \mathsf{ct})$:
  - Sample $e_i \leftarrow D_{flood}$
  - Output $d_i = \mathbf{u}\mathbf{s}_i + e_i$

- $\mathsf{Combine}(d_1, \ldots, d_n)$:
  - $d = \sum_{i=1}^{n} d_i$
  - If $v - d$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$

Full-Threshold PKE from LWE [BD10]

- $\mathsf{KGen}(1^\lambda)$:
  - $\mathbf{A} \leftarrow \mathsf{Unif}(\mathbb{Z}_q^{n\times n})$ and $\mathbf{s}, \mathbf{e} \leftarrow \chi^n$
  - $\mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \bmod q$
  - $\mathbf{s} = \sum_{i=1}^{n} \mathbf{s}_i$
  - Output $\mathsf{sk}_i = \mathbf{s}_i$ and $\mathsf{pk} = (\mathbf{A}, \mathbf{b})$

$$\boxed{\mathbf{A}}\ ,\ \boxed{\mathbf{A}}\,\boxed{\mathbf{s}} + \boxed{\mathbf{e}} = \boxed{\mathbf{b}}$$

- $\mathsf{PartDec}(\mathsf{sk}_i, \mathsf{ct})$:
  - Sample $e_i \leftarrow D_{flood}$
  - Output $d_i = \mathbf{u}\mathbf{s}_i + e_i$

- $\mathsf{Combine}(d_1, \ldots, d_n)$:
  - $d = \sum_{i=1}^{n} d_i$
  - If $v - d$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$

Correctness:

$$v - \sum_{i=1}^{n} \mathbf{u}\mathbf{s}_i + e_i = v - \mathbf{u}\sum_{i=1}^{n} \mathbf{s}_i + e_i = v - \mathbf{u}\mathbf{s} + \sum_{i=1}^{n} e_i$$

$$= \underbrace{\mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s} + \sum_{i=1}^{n} e_i}_{*} + \lfloor q/2 \rfloor m$$

Decryption succeeds if $|*| < q/8$

Put under the carpet for today ...

⚠ It is non-trivial to go from full-threshold to arbitrary threshold PKE
if you are working with lattices ;-)

n-out-of-n threshold

$$\sum_{i=1}^{n} e_i$$

t-out-of-n threshold

$$\underbrace{\sum_{i \in S} \lambda_i e_i}_{\text{still needs to be small}}$$

? There are solutions, but not very efficient for large n.

Partial Decryption Security

Two worlds:

- Real: $e_{\mathsf{ct}} = \mathbf{re} + f' - \mathbf{fs}$ and $e_{flood} = \sum_i e_i$
- Simulated: only $e_{flood} = \sum_i e_i$

How close are they? [BD10] measures with statistical distance Δ

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + e_{\mathsf{ct}}, e_{flood}) \leq \mathsf{negl}(\lambda)$$

Partial Decryption Security

Two worlds:

- Real: $e_{\text{ct}} = \mathbf{re} + f' - \mathbf{fs}$ and $e_{flood} = \sum_i e_i$
- Simulated: only $e_{flood} = \sum_i e_i$

How close are they? [BD10] measures with statistical distance Δ

$$\Delta(\text{Real}, \text{Sim}) \leq \Delta(e_{flood} + e_{\text{ct}}, e_{flood}) \leq \text{negl}(\lambda)$$

Problem:

- $\|e_{flood}\|$ needs to be super-polynomially larger than $\|e_{\text{ct}}\|$
- LWE-based constructions: $\|e_{flood}\| \sim$ LWE modulus q and $\|e_{\text{ct}}\| \sim$ LWE noise $\mathbf{e}$, thus super-polynomial modulus-noise ratio
  - Larger parameters
  - Easier problem

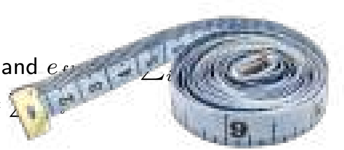
$$\mathbf{A}, \ \mathbf{A}\mathbf{s} + \mathbf{e} \mod q$$

Partial Decryption Security

💡 Idea: change the measure! [BLR+18]

Two worlds:

- Real: $e_{\text{ct}} = \mathbf{r}\mathbf{e} + f' - \mathbf{f}\mathbf{s}$ and e_{fl} ... $\sum_i$
- Simulated: only $e_{flood} = \sum$



How close are they? [BD10] measures with statistical distance Δ

$$\Delta(\text{Real}, \text{Sim}) \leq \Delta(e_{flood} + e_{\text{ct}}, e_{flood}) \leq \text{negl}(\lambda)$$

Problem:

- $\|e_{flood}\|$ needs to be super-polynomially larger than $\|e_{\text{ct}}\|$
- LWE-based constructions: $\|e_{flood}\| \sim$ LWE modulus q and $\|e_{\text{ct}}\| \sim$ LWE noise $\mathbf{e}$, thus super-polynomial modulus-noise ratio
  - Larger parameters
  - Easier problem

$$\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e} \mod q$$

Improved Noise Flooding via Rényi Divergence 1/2

Let P, Q be discrete probability distributions

In [BD10]: Statistical Distance $\Delta(P,Q) = \frac{1}{2}\sum_{x\in\mathrm{Supp}(P)}|P(x)-Q(x)|$

In [BS23]: Rényi Divergence

$$\mathsf{RD}(P,Q) = \sum_{\substack{x\in\mathrm{Supp}(P)\\ \subset \mathrm{Supp}(Q)}} \frac{P(x)^2}{Q(x)}$$

Improved Noise Flooding via Rényi Divergence 1/2

Let P, Q be discrete probability distributions

In [BD10]: Statistical Distance $\Delta(P,Q) = \frac{1}{2}\sum_{x \in \mathrm{Supp}(P)} |P(x) - Q(x)|$

In [BS23]: Rényi Divergence

$$\mathsf{RD}(P,Q) = \sum_{\substack{x \in \mathrm{Supp}(P) \\ \subset \mathrm{Supp}(Q)}} \frac{P(x)^2}{Q(x)}$$

Both fulfill the **probability preservation property** for an event E:

[BD10]:	$P(E)$	$\leq$	$\Delta(P,Q) + Q(E)$	(additive)
Our work:	$P(E)^2$	$\leq$	$\mathsf{RD}(P,Q) \cdot Q(E)$	(multiplicative)

- $Q(E)$ negligible $\Rightarrow P(E)$ negligible
- $\Delta(P,Q) \stackrel{!}{=}$ negligible and $\mathsf{RD}(P,Q) \stackrel{!}{=}$ **constant**

Improved Noise Flooding via Rényi Divergence 2/2

Two worlds:

- Real: e_{ct} and e_{flood}
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + e_{\mathsf{ct}}, e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + e_{\mathsf{ct}}, e_{flood}) \leq \text{constant}$$

Advantage:

- $\|e_{flood}\|$ only needs to be polynomially larger than $\|e_{\mathsf{ct}}\|$
- LWE-based constructions: polynomial modulus-noise ratio

Improved Noise Flooding via Rényi Divergence 2/2

Two worlds:

- Real: e_{ct} and e_{flood}
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + e_{\mathsf{ct}}, e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + e_{\mathsf{ct}}, e_{flood}) \leq \text{constant}$$

Advantage:

- $\|e_{flood}\|$ only needs to be polynomially larger than $\|e_{\mathsf{ct}}\|$
- LWE-based constructions: polynomial modulus-noise ratio

Disadvantage:

1) Rényi divergence depends on the number of issued partial decryptions
   $\rightarrow$ from simulation-based to game-based security notion
2) Works well with search problems, not so well with decision problems

Zooming out - leakage on secret key

Two worlds:

- Real: $f(\mathsf{sk})$ and e_{flood} — f some function
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \text{constant}$$

Zooming out - leakage on secret key

Two worlds:

- Real: $f(\mathsf{sk})$ and e_{flood} — f some function
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \text{constant}$$

Examples:

- Threshold decryption: $f(\mathsf{sk})$ is the ciphertext noise [BS23]
- Signatures schemes: $f(\mathsf{sk})$ is part of a signature [Raccoon]

Zooming out - leakage on secret key

Two worlds:

- Real: $f(\mathsf{sk})$ and e_{flood} $\qquad$ f some function
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \text{constant}$$

Examples:

- Threshold decryption: $f(\mathsf{sk})$ is the ciphertext noise [BS23]
- Signatures schemes: $f(\mathsf{sk})$ is part of a signature [Raccoon]

Alternative Approaches:

- Rejection Sampling $\rightarrow$ Dilithium
- LWE with hints aka just accept the leakage

Zooming out - leakage on secret key

Two worlds:

- Real: $f(\mathsf{sk})$ and e_{flood} — f some function
- Simulated: only e_{flood}

How close are they?

$$\Delta(\mathsf{Real}, \mathsf{Sim}) \leq \Delta(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \mathsf{negl}(\lambda)$$

$$\mathsf{RD}(\mathsf{Real}, \mathsf{Sim}) \leq \mathsf{RD}(e_{flood} + f(\mathsf{sk}), e_{flood}) \leq \text{constant}$$

Examples:

- Threshold decryption: $f(\mathsf{sk})$ is the ciphertext noise [BS23]
- Signatures schemes: $f(\mathsf{sk})$ is part of a signature [Raccoon]

“ We don't yet understand very well when which approach is **optimal** ”

Alternative Approaches:

- Rejection Sampling → Dilithium
- LWE with hints aka just accept the leakage

Wrap-Up

Hopefully you have now a rough idea:

- Part 1: *What lattices are!*
- Part 2: *What lattice problems are!*
- Part 3: *What lattice-based cryptography is!*
- Part 4: *What particular challenges are!*

Any questions or interested in my research?

- Reach out to me today or at Latincrypt
- Write me an e-mail

Wrap-Up

Hopefully you have now a rough idea:

- Part 1: *What lattices are!*
- Part 2: *What lattice problems are!*
- Part 3: *What lattice-based cryptography is!*
- Part 4: *What particular challenges are!*

¡Muchas Gracias!

Any questions or interested in my research?

- Reach out to me today or at Latincrypt
- Write me an e-mail

Miklós Ajtai.
Generating hard instances of lattice problems (extended abstract).
In *STOC*, pages 99–108. ACM, 1996.

Rikke Bendlin and Ivan Damgård.
Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems.
In *TCC*, volume 5978 of *Lecture Notes in Computer Science*, pages 201–218. Springer, 2010.

Shi Bai, Tancrède Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld.
Improved security proofs in lattice-based cryptography: Using the rényi divergence rather than the statistical distance.
J. Cryptol., 31(2):610–640, 2018.

Katharina Boudgoust and Peter Scholl.
Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus.
IACR Cryptol. ePrint Arch., page 16, 2023.

Oded Regev.
On lattices, learning with errors, random linear codes, and cryptography.
In *STOC*, pages 84–93. ACM, 2005.

Peter W. Shor.

Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.
SIAM J. Comput., 26(5):1484–1509, 1997.