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?
- Part 4: What are the current challenges?

References:

- Crash Course Spring 2022 [lecture notes]
- The Lattice Club [link]
The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:
- Discrete logarithm → Arantxa’s proof system
- Factoring
The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:
- Discrete logarithm
- Factoring

.quantum algorithm [Sho97]

Quantum-resistant candidates:
- Codes
- Lattices
- Isogenies
- Multivariate systems
- ?
The security in public-key cryptography relies on presumably hard mathematical problems.

Currently used problems:
- Discrete logarithm
- Factoring

Quantum-resistant candidates:
- Codes
- Lattices → now
- Isogenies → later with Chloe
- Multivariate systems
- ?
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.

Part 1:

What is a lattice?
An Euclidean lattice $\Lambda$ is a discrete additive subgroup of $\mathbb{R}^n$. 
An Euclidean lattice \( \Lambda \) is a \textit{discrete additive subgroup} of \( \mathbb{R}^n \).

- \textbf{additive subgroup}: \( 0 \in \Lambda \), and for all \( x, y \in \Lambda \) it holds \( x + y, -x \in \Lambda \);
- \textbf{discrete}: every \( x \in \Lambda \) has a neighborhood in which \( x \) is the only lattice point.

\[ \exists \varepsilon > 0 \text{ such that } B(x, \varepsilon) \cap \Lambda = \{ x \} \]
An Euclidean lattice $\Lambda$ is a discrete additive subgroup of $\mathbb{R}^n$.

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

There exists a finite basis $B = (b_1, \ldots, b_n) \subset \mathbb{R}^n$ such that

$$\Lambda(B) = \left\{ \sum_{i=1}^{n} z_i b_i : z_i \in \mathbb{Z} \right\}.$$  

$n$ is the rank of $\Lambda$.
Euclidean Lattices

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

$$\Lambda(B) = \left\{ \sum_{i=1}^{n} z_i b_i : z_i \in \mathbb{Z} \right\} = \{Bz : z \in \mathbb{Z}^n\}.$$
Euclidean Lattices

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

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

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

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

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

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

\[
\Lambda^\vee = \{ w \in \mathbb{R}^n : \langle w, x \rangle \in \mathbb{Z} \ \forall x \in \Lambda \}.
\]

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

\[
2\mathbb{Z}^2 \text{ and its dual } \frac{1}{2}\mathbb{Z}^2
\]
Lattice Minimum & Special Lattices

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

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

- **Minkowski:** $\lambda_1(\Lambda) \leq \sqrt{n} \cdot \det(\Lambda)^{1/n}$
- **Exercise:** $\lambda_1(\Lambda) \cdot \lambda_1(\Lambda^\vee) \leq n$
The **minimum** of a lattice $\Lambda \subset \mathbb{R}^n$ is defined as

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

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

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

$$\Lambda_q(A) = \{ y \in \mathbb{Z}^m : y = As \text{ mod } q \text{ for some } s \in \mathbb{Z}^n \}$$

$$\Lambda_q^\perp(A) = \{ y \in \mathbb{Z}^m : A^T y = 0 \text{ mod } q \}$$

- **Exercise**: $\Lambda_q^\perp(A) = q \cdot \Lambda_q(A)^\perp$
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 $w \in \Lambda$ such that

$$\|w\|_2 = \lambda_1(\Lambda).$$
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 $w \in \Lambda$ such that

$$\|w\|_2 \leq \gamma \lambda_1(\Lambda).$$
Shortest Vector Problem

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

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

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

The complexity of $\text{SVP}_\gamma$ increases with $n$, but decreases with $\gamma$.

**Conjecture:**
There is no polynomial-time classical or quantum algorithm that solves $\text{SVP}_\gamma$ to within polynomial factors.
Bounded Distance Decoding

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

Conjecture: There is no polynomial-time classical or quantum algorithm that solves BDD$_{\delta}$ to within polynomial factors.
Bounded Distance Decoding

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

The bounded distance decoding (BDD$_\delta$) problem asks to find the unique vector $w \in \Lambda$ such that

$$\|w - t\|_2 \leq \delta.$$
Bounded Distance Decoding

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

The bounded distance decoding (BDD$_\delta$) problem asks to find the unique vector $w \in \Lambda$ such that

$$\|w - t\|_2 \leq \delta.$$

The complexity of BDD$_\delta$ increases with $n$ and with $\delta$.

**Conjecture:**
There is no polynomial-time classical or quantum algorithm that solves BDD$_\delta$ to within polynomial factors.
Short Integer Solution [Ajt96]

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

\[ \Lambda_\perp q(A) = \{ y \in \mathbb{Z}^m : A^T y = 0 \text{ mod } q \} \]

$\text{SIS}_\beta$ equals $\text{SVP}_\gamma$ in the special lattice $\Lambda_\perp q(A)$ for $\beta = \gamma \cdot \lambda_1(\Lambda_\perp q(A))$.
Short Integer Solution [Ajt96]

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

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

$$A^T z = 0 \mod q.$$
Short Integer Solution [Ajt96]

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

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

$$A^T z = 0 \mod q.$$ 

⚠️ The norm restriction makes it a hard problem!
Short Integer Solution [Ajt96]

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

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

$$A^T z = 0 \mod q.$$ 

⚠️ The norm restriction makes it a hard problem!

Recall:

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

👍 SIS$_\beta$ equals SVP$_\gamma$ in the special lattice $\Lambda_q^\perp(A)$ for $\beta = \gamma \cdot \lambda_1(\Lambda_q^\perp(A))$
Given a matrix $A \leftarrow \text{Unif}(\mathbb{Z}_q^{m \times n})$.

Given a vector $b \in \mathbb{Z}_q^m$, where $b = As + e \mod q$ for

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

$A$, $A$ + $e$, $s$
Learning With Errors [Reg05]

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

Given a vector $b \in \mathbb{Z}_q^m$, where $b = As + e \mod q$ for

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

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

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

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

\[ m \quad \text{A} \quad A \quad e \quad s \quad \rightarrow \text{find } s \]

\[ \approx \text{uniform} \]
Learning With Errors [Reg05]

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

Given a vector $b \in \mathbb{Z}_q^m$, where $b = As + e \mod q$ for

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

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

Decision learning with errors (D-LWE$_\delta$) asks to distinguish $(A, b)$ 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 $e$ makes D-LWE a hard problem!
Learning With Errors [Reg05]

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

Given a vector $b \in \mathbb{Z}_q^m$, where $b = As + e \mod q$ for

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

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

Decision learning with errors (D-LWE$_{\delta}$) asks to distinguish $(A, b)$ 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 $e$ makes D-LWE a hard problem!

Exercise: S-LWE$_{\delta}$ equals BDD$_{\delta}$ in the special lattice $\Lambda_q(A)$. 
If there is an efficient solver for SIS$_\beta$, then there is an efficient solver for D-LWE$_\delta$, assuming $\delta \cdot \beta \ll q$. 
If there is an efficient solver for SIS\(_\beta\), then there is an efficient solver for D-LWE\(_\delta\), assuming \(\delta \cdot \beta \ll q\).

**Proof.**

Given \((A, b)\), our goal is to decide whether 1) \(b = As + e\) for \(||e||_2 \leq \delta\) or 2) \(b \leftarrow \text{Unif}(\mathbb{Z}_q^m)\).
If there is an efficient solver for SIS\(\beta\), then there is an efficient solver for D-LWE\(\delta\), assuming \(\delta \cdot \beta \ll q\).

Proof.

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

Forward \(A\) to SIS-solver and receive back \(z\) such that \(A^Tz = 0 \mod q\) and \(\|z\|_2 \leq \beta\).
Connection between LWE and SIS

If there is an efficient solver for SIS\(_\beta\), then there is an efficient solver for D-LWE\(_\delta\), assuming \(\delta \cdot \beta \ll q\).

Proof.

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

Forward \(A\) to SIS-solver and receive back \(z\) such that \(A^Tz = 0 \mod q\) and \(\|z\|_2 \leq \beta\).

Compute \(\|b^Tz\|_\infty\). If the norm is \(\ll q\), claim that we are in case 1). Else, claim that we are in case 2).
Connection between LWE and SIS

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

Proof.

Given $(A, b)$, our goal is to decide whether 1) $b = As + e$ for $\|e\|_2 \leq \delta$ or 2) $b \leftarrow \text{Unif}(Z^m_q)$.

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

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

Case 1) $b = As + e$, thus $b^T z = s^T A^T z + e^T z = e^T z \mod q$. Thus $\|b^T z\|_\infty \leq \|e^T\|_\infty \cdot \|z\|_\infty \leq \delta \cdot \beta \ll q$.

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

Part 3:

What is lattice-based cryptography?
Collision-Resistant Hash Function from SIS [Ajt96]

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

$$f(x) = f(x') \text{ and } x \neq x'.$$
Collision-Resistant Hash Function from SIS [Ajt96]

A function \( f : \text{Domain} \rightarrow \text{Range} \) is called \textit{collision-resistant} if it is hard to output two elements \( x, x' \in \text{Domain} \) such that

\[
f(x) = f(x') \quad \text{and} \quad x \neq x'.
\]

Set \( f_A : \{0, 1\}^m \rightarrow \mathbb{Z}_q^n \) with \( f_A(x) = A^T x \mod q \) for \( A \leftarrow \text{Unif}(\mathbb{Z}_q^{m \times n}) \).
Collision-Resistant Hash Function from SIS [Ajt96]

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

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

Set $f_A : \{0, 1\}^m \rightarrow \mathbb{Z}_q^n$ with $f_A(x) = A^T x \mod q$ for $A \leftarrow \text{Unif}(\mathbb{Z}_q^{m \times n})$.

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

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

$$A^T x = A^T x' \iff A^T (x - x') = 0$$
Reminder: Public-Key Encryption (PKE)

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

- $\text{KGen}(1^\lambda) \rightarrow (sk, pk)$  
  $\lambda$ security parameter
- $\text{Enc}(pk, m) \rightarrow ct$
- $\text{Dec}(sk, ct) = m'$

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

**Semantic Security:** $\text{Enc}(pk, m_0)$ is indistinguishable from $\text{Enc}(pk, m_1)$  
  (IND-CPA)
Public-Key Encryption from LWE [Reg05]

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

- **KGen($1^\lambda$):**
  - $A \leftarrow \text{Unif}(\mathbb{Z}^n_q \times n)$ and $s, e \leftarrow \chi^n$
  - $b = As + e \mod q$
  - Output sk = $s$ and pk = $(A, b)$
Public-Key Encryption from LWE [Reg05]

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

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

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

![Diagram showing the encryption process]
Public-Key Encryption from LWE [Reg05]

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

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

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

- **Dec($sk, ct$):**
  - If $v - us$ is closer to 0 than to $q/2$, output $m' = 0$
  - Else output $m' = 1$
Public-Key Encryption from LWE [Reg05]

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

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

- **Dec(sk, ct):**
  - If \( v - us \) is closer to 0 than to \( q/2 \), output \( m' = 0 \)
  - Else output \( m' = 1 \)

**Correctness:**

\[
v - us = r(As + e) + f' + \lfloor q/2 \rfloor \cdot m - (rA + f)s
= re + f' - fs + \lfloor q/2 \rfloor m
\]

Decryption succeeds if \(|*| < q/8\)
Public-Key Encryption from LWE [Reg05]

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

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

- **Dec(sk, ct):**
  - If \( v - us \) is closer to 0 than to \( q/2 \), output \( m' = 0 \)
  - Else output \( m' = 1 \)

**Correctness:** Let \( \chi \) be \( B \)-bounded with \( 2nB^2 + B < q/8 \)

\[
v - us = r(As + e) + f' + \lfloor q/2 \rfloor \cdot m - (rA + f)s
\]

\[
= re + f' - fs + \lfloor q/2 \rfloor m
\]

\* ciphertext noise

Decryption succeeds if \(|*| < q/8\)

\[
|*| = |re + f' - fs| \leq \|r\|_2 \cdot \|e\|_2 + \|f\|_2 \cdot \|s\|_2 + |f'| \leq 2(\sqrt{n}B \cdot \sqrt{n}B) + B < q/8
\]
Public-Key Encryption from LWE [Reg05]

- **KGen(1^λ):**
  - A ← Unif(\(\mathbb{Z}_q^{n\times n}\)) and s, e ← \(\chi^n\)
  - b = As + e mod q
  - Output sk = s and pk = (A, b)

- **Enc(pk, m ∈ \{0, 1\}):**
  - r, f ← \(\chi^n\) and f’ ← \(\chi\)
  - u = rA + f
  - v = rb + f’ + [q/2] \cdot m
  - Output ct = (u, v)

- **Dec(sk, ct):**
  - If v − us is closer to 0 than to q/2, output \(m' = 0\)
  - Else output \(m' = 1\)

---

**Semantic Security:** Assume hardness of decision LWE

1. replace 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]
Part 4:

What are (my) current challenges?
Re-Reminder: Public Key Encryption (PKE)

PKE scheme:
- $KGen(1^\lambda) \rightarrow (pk, sk)$
- $Enc(pk, m) \rightarrow ct$
- $Dec(sk, ct) \rightarrow m'$

Properties:
- Correctness
- Semantic security
Re-Reminder: Public Key Encryption (PKE)

PKE scheme:
- $\text{KGen}(1^\lambda) \rightarrow (\text{pk}, \text{sk})$
- $\text{Enc}(\text{pk}, m) \rightarrow \text{ct}$
- $\text{Dec}(\text{sk}, \text{ct}) \rightarrow m'$

Properties:
- Correctness
- Semantic security

⚠️ Single Point of Failure
Threshold Public Key Encryption (TPKE)

$t$-out-of-$n$ Threshold PKE scheme:

- $\text{KGen}(1^\lambda) \rightarrow (pk, sk_1, \ldots, sk_n)$
- $\text{Enc}(pk, m) \rightarrow ct$
- $\text{PartDec}(sk_i, ct') \rightarrow d_i$
- $\text{Combine}(\{d_i\}_{i \in S}) \rightarrow m'$

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

*https://csrc.nist.gov/projects/threshold-cryptography*
Threshold Public Key Encryption (TPKE)

$t$-out-of-$n$ Threshold PKE scheme:
- $KGen(1^\lambda) \rightarrow (pk, sk_1, \ldots, sk_n)$
- $Enc(pk, m) \rightarrow ct$
- $PartDec(sk_i, ct') \rightarrow d_i$
- $Combine(\{d_i\}_{i \in S}) \rightarrow m'$

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
- $\cdots$
- Chris yesterday, Daniel later

secret sharing

\[ i \in \{1, \ldots, n\} \]
\[ S \subset \{1, \ldots, n\} \]

*https://csrc.nist.gov/projects/threshold-cryptography*
Threshold Public Key Encryption (TPKE)

t-out-of-n Threshold PKE scheme:

- **KGen**(1^λ) → (pk, sk_1, …, sk_n)  
  secret sharing

- **Enc**(pk, m) → ct

- **PartDec**(sk_i, ct') → d_i  
  i ∈ {1, …, n}

- **Combine**(\{d_i\}_{i∈S}) → m'  
  S ⊂ \{1, …, n\}

Properties:

- Correctness  
  for |S| > t recover correct message

- Partial decryption security  
  for |S| ≤ t no information is leaked

- Semantic security

Applications:

- Storing sensitive data
  
  NIST’s call*

- Electronic voting protocols

- Multiparty computations  
  → Chris yesterday, Daniel later

---

*https://csrc.nist.gov/projects/threshold-cryptography
Reminder: PKE from LWE

- **KGen\(1^\lambda\):**
  - \(A \leftarrow \text{Unif}(\mathbb{Z}_q^{n \times n})\) and \(s, e \leftarrow \chi^n\)
  - \(b = As + e \mod q\)
  - Output \(sk = s\) and \(pk = (A, b)\)

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

- **Dec(sk, ct):**
  - If \(v - us\) is closer to 0 than to \(q/2\), output \(m' = 0\)
  - Else output \(m' = 1\)
Reminder: PKE from LWE

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

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

- **Dec(sk, ct):**
  - If $v - us$ 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

\((n\text{-out-of-}n)\)

- **KGen\(1^\lambda\):**
  - \(A \leftarrow \text{Unif}(\mathbb{Z}_q^{n \times n})\) and \(s, e \leftarrow \chi^n\)
  - \(b = As + e \mod q\)
  - \(s_1, \ldots, s_{n-1} \leftarrow \text{Unif}(\mathbb{Z}_q^n)\)
  - \(s_n = s - \sum_{i=1}^{n-1} s_i\)
  - Output \(sk_i = s_i\) and \(pk = (A, b)\)

- **PartDec**(\(sk_i, (u, v)\)):
  - Output \(d_i = us_i\)

- **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 \(m' = 1\)
Full-Threshold PKE from LWE, First Trial

- **KGen($1^\lambda$):**
  - $A \leftarrow \text{Unif}(\mathbb{Z}_q^{n \times n})$ and $s, e \leftarrow \chi^n$
  - $b = As + e \mod q$
  - $s_1, \ldots, s_{n-1} \leftarrow \text{Unif}(\mathbb{Z}_q^n)$
  - $s_n = s - \sum_{i=1}^{n-1} s_i$
  - Output $sk_i = s_i$ and $pk = (A, b)$

- **PartDec($sk_i, (u, v)$):**
  - Output $d_i = us_i$

- **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 $d_1, \ldots, d_n$

\[
v - \sum_{i=1}^n us_i = v - u \sum_{i=1}^n s_i = v - us = re + f' - fs + [q/2]m\]

Decryption succeeds if $|\ast| < q/8$
Full-Threshold PKE from LWE, First Trial

KGen(1^λ):

- A ← Unif(\mathbb{Z}_q^{n\times n}) and s, e ← \chi^n
- b = As + e \mod q
- s_1, \ldots, s_{n-1} ← Unif(\mathbb{Z}_q^n)
- s_n = s - \sum_{i=1}^{n-1} s_i
- Output sk_i = s_i and pk = (A, b)

PartDec(sk_i, (u, v)):

- Output d_i = us_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

\[
v - \sum_{i=1}^{n} us_i = v - u \sum_{i=1}^{n} s_i = v - us
\]

\[
= re + f' - fs + \lfloor q/2 \rfloor m
\]

⚠️ But (*) leaks information about sk = s!
Full-Threshold PKE from LWE [BD10]

KGen(1^λ):
- A ← Unif(Z_q^{n×n}) and s, e ← χ^n
- b = As + e mod q
- s = ∑_{i=1}^n s_i
- Output sk_i = s_i and pk = (A, b)

PartDec(sk_i, ct):
- Sample e_i ← D_flood
- Output d_i = us_i + e_i

Combine(d_1, . . . , d_n):
- d = ∑_{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]

- **KGen(1^λ):**
  - \( A \leftarrow \text{Unif}(\mathbb{Z}_q^{\times n}) \) and \( s, e \leftarrow \chi^n \)
  - \( b = As + e \mod q \)
  - \( s = \sum_{i=1}^{n} s_i \)
  - Output \( sk_i = s_i \) and \( pk = (A, b) \)

- **PartDec(sk_i, ct):**
  - Sample \( e_i \leftarrow D_{\text{flood}} \)
  - Output \( d_i = us_i + e_i \)

- **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:**

\[
v - \sum_{i=1}^{n} us_i + e_i = v - u \sum_{i=1}^{n} s_i + e_i = v - us + \sum_{i=1}^{n} e_i = re + f' - fs + \sum_{i=1}^{n} e_i + \lfloor q/2 \rfloor m
\]

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 ;-) 

\[
\begin{align*}
n\text{-out-of-}n \text{ threshold} & \quad t\text{-out-of-}n \text{ threshold} \\
\sum_{i=1}^{n} e_i & \quad \sum_{i \in S} \lambda_i e_i \\
\end{align*}
\]

still needs to be small

❓ There are solutions, but not very efficient for large $n$. 
Partial Decryption Security

Two worlds:
- Real: \( e_{ct} = re + f' - fs \) and \( e_{flood} = \sum_i e_i \)
- Simulated: only \( e_{flood} = \sum_i e_i \)

How close are they? [BD10] measures with statistical distance \( \Delta \)

\[
\Delta(\text{Real, Sim}) \leq \Delta(e_{flood} + e_{ct}, e_{flood}) \leq \text{negl}(\lambda)
\]
Partial Decryption Security

Two worlds:
- Real: \( e_{ct} = re + f' - fs \) and \( e_{flood} = \sum_i e_i \)
- Simulated: only \( e_{flood} = \sum_i e_i \)

How close are they? [BD10] measures with statistical distance \( \Delta \)

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

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

\[
\begin{align*}
A & , \quad A \\
+ & \quad \text{mod } q
\end{align*}
\]
Partial Decryption Security

Two worlds:
- **Real:** \( e_{ct} = re + f^\prime - fs \) and \( e_{flood} = \triangle \)
- **Simulated:** only \( e_{flood} = \triangle \)

How close are they? [BD10] measures with statistical distance \( \Delta \)

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

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

\[
\begin{array}{ccc}
A & & A \\
\downarrow & & \downarrow \\
& s & + e \mod q
\end{array}
\]
Let $P, Q$ be discrete probability distributions

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

In [BS23]: Rényi Divergence

$$\text{RD}(P, Q) = \sum_{x \in \text{Supp}(P)} \frac{P(x)^2}{Q(x)}$$
Improved Noise Flooding via Rényi Divergence 1/2

Let $P, Q$ be discrete probability distributions.

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

In [BS23]: Rényi Divergence

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

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

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

- $Q(E)$ negligible $\Rightarrow P(E)$ negligible
- $\Delta(P, Q) =^!$ negligible and $\text{RD}(P, Q) =^!$ 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?

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

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

Advantage:

- $\|e_{flood}\|$ only needs to be polynomially larger than $\|e_{ct}\|$.
- 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?

\[ \Delta(\text{Real}, \text{Sim}) \leq \Delta(e_{flood} + e_{ct}, e_{flood}) \leq \text{negl}(\lambda) \]
\[ \text{RD}(\text{Real}, \text{Sim}) \leq \text{RD}(e_{flood} + e_{ct}, e_{flood}) \leq \text{constant} \]

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

Disadvantage:
1) Rényi divergence depends on the number of issued partial decryptions  
   → 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: $f(\text{sk})$ and $e_{\text{flood}}$
- Simulated: only $e_{\text{flood}}$

How close are they?

\[
\Delta(\text{Real, Sim}) \leq \Delta(e_{\text{flood}} + f(\text{sk}), e_{\text{flood}}) \leq \text{negl}(\lambda)
\]

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

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

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

We don’t yet understand very well when which approach is optimal
Zooming out - leakage on secret key

Two worlds:
- Real: $f(\text{sk})$ and $e_{\text{flood}}$
- Simulated: only $e_{\text{flood}}$

How close are they?

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

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

Examples:
- Threshold decryption: $f(\text{sk})$ is the ciphertext noise
  [BS23]
- Signatures schemes: $f(\text{sk})$ is part of a signature
  [Raccoon]
Zooming out - leakage on secret key

Two worlds:
- Real: \( f(sk) \) and \( e_{flood} \)
- Simulated: only \( e_{flood} \)

How close are they?

\[
\Delta(\text{Real}, \text{Sim}) \leq \Delta(e_{flood} + f(sk), e_{flood}) \leq \text{negl}(\lambda)
\]
\[
\text{RD}(\text{Real}, \text{Sim}) \leq \text{RD}(e_{flood} + f(sk), e_{flood}) \leq \text{constant}
\]

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

Alternative Approaches:
- Rejection Sampling \(\rightarrow\) Dilithium
- LWE with hints aka just accept the leakage
Zooming out - leakage on secret key

Two worlds:
- Real: $f(sk)$ and $e_{flood}$
- Simulated: only $e_{flood}$

How close are they?

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

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

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

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

"We don't yet understand very well when which approach is optimal"
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?

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

¡Muchas Gracias!
Miklós Ajtai.
Generating hard instances of lattice problems (extended abstract).

Rikke Bendlin and Ivan Damgård.
Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems.

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.

Katharina Boudgoust and Peter Scholl.
Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus.

Oded Regev.
On lattices, learning with errors, random linear codes, and cryptography.

Peter W. Shor.
Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.