Update on the lattice-code relationship ♡

Antoine Douteau
CNRS, GREYC, Caen

Journée Crypto Caen/Rouen

September 12, 2025

This work explains -not in details- the paper [Albrecht, Benčina, and Lai, 2025]

PKE and Updatable PKE

Public Key Encryption

Encrypt messages with a public key, and decrypt using the associated private key

IND-CPA

Indistinguishability of chosen-plaintext-messages

PKE and Updatable PKE

Updatable Public Key Encryption

Encrypt messages with a public key, and decrypt using the associated private key

Public users can update at any time the keys used

IND-CPA-CR

Indistinguishability of chosen-plaintext-messages using chosen randomness

PKE and Updatable PKE

Updatable Public Key Encryption

Encrypt messages with a public key, and decrypt using the associated private key

Public users can update at any time the keys used

IND-CPA-CR

Indistinguishability of chosen-plaintext-messages using chosen randomness

Why ?

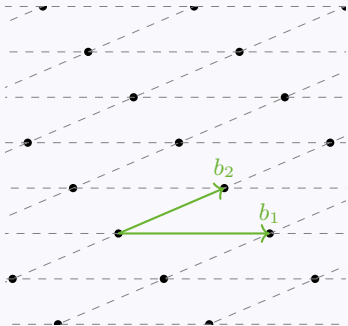
Allows public users to update the public key if not convinced on the security

Previous ciphertexts remains secure even if the updated secret is leaked

Lattices

Definitions

Lattice



$\Lambda = \mathcal{L}(\mathcal{B})$

Finite discrete subgroup of $\mathbb{R}^n$

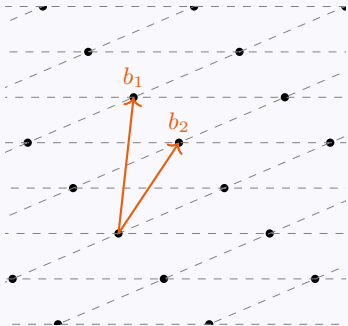
- Evenly spaced elements
- Can be defined by a base $\mathcal{B}$

$$\mathcal{B} = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Lattices

Definitions

Lattice



$\Lambda = \mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{B})$

Finite discrete subgroup of $\mathbb{R}^n$

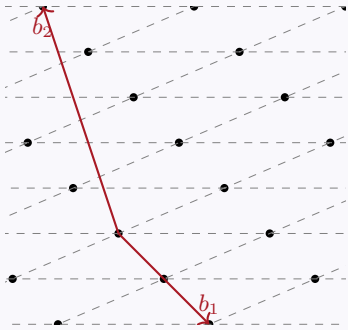
- Evenly spaced elements
- Can be defined by a base $\mathcal{B}$

$$\mathcal{B} = \begin{bmatrix} -4 & -2 \\ 6 & 4 \end{bmatrix}$$

Lattices

Definitions

Lattice



$\Lambda = \mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{B})$

Finite discrete subgroup of $\mathbb{R}^n$

- Evenly spaced elements
- Can be defined by a base $\mathcal{B}$

$$\mathcal{B} = \begin{bmatrix} 4 & -9 \\ -4 & 10 \end{bmatrix}$$

Lattices

Assumptions

Learning With Errors

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}$

$\mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \mathcal{D}_\sigma^m$

$\mathbf{b}_0 \leftarrow \mathbb{Z}_q^m$

$\mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}$

Knowing $\mathbf{A}$ and b_i, find i

Lattices

Assumptions

Learning With Errors

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}$

$\mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \mathcal{D}_\sigma^m$

$\mathbf{b}_0 \leftarrow \mathbb{Z}_q^m$

$\mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}$

Knowing $\mathbf{A}$ and b_i, find i

Lattice Isomorphism Problem

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times n}$,

$\mathbf{O} \leftarrow \mathcal{O}_n(\mathbb{R}), \mathbf{U} \leftarrow \mathcal{GL}_n(\mathbb{Z})$

$\mathbf{B}_0 \leftarrow \mathbb{Z}_q^{n \times n}$

$\mathbf{B}_1 = \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$

Knowing $\mathbf{A}$ and $\mathbf{B}_i$, find i

. . . and codes

Definitions

Code

An $[n, k]$-linear code $\mathcal{C}$ over $\mathbb{Z}_q$ is a k-dimensional vector subspace of $\mathbb{Z}_q^n$.

$\mathcal{C}$ is represented by a generator $\mathbf{G} \in \mathbb{Z}_q^{n \times k}$ or a parity check matrix $\mathbf{H} \in \mathbb{Z}_q^{n \times (n-k)}$:

$$\mathbf{c} \in \mathcal{C} \iff \mathbf{c}^\top \cdot \mathbf{H} = \mathbf{0}$$

Then, $\mathbf{H}$ generates a $[n, n-k]$-linear code also known as the dual-code $\mathcal{C}^\perp$ of $\mathcal{C}$.

Hull and h-hollow

We define $\text{hull}(\mathcal{C}) := \mathcal{C} \cap \mathcal{C}^\perp$.

$\mathcal{C}$ (resp. $\mathbf{G}$) is called h-hollow if $\dim(\text{hull}(\mathcal{C})) = h$ (if $\mathbf{G}$ is full rank).

. . . and codes

Assumptions

Permutation Code Equivalence

$\mathbf{G} \xleftarrow{\$} \mathbb{Z}_q^{n \times k}$

$\mathbf{O} \leftarrow \mathcal{O}_n(\mathbb{Z})$, $\mathbf{U} \leftarrow \mathcal{GL}_k(\mathbb{Z}_q)$

$\mathbf{H}_0 \xleftarrow{\$} \mathbb{Z}_q^{n \times k}$

$\mathbf{H}_1 = \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$

Knowing $\mathbf{G}$ and $\mathbf{H}_i$, find i.

. . . and codes

Assumptions

Permutation Code Equivalence

$\mathbf{G} \xleftarrow{\$} \mathbb{Z}_q^{n \times k}$

$\mathbf{O} \leftarrow \mathcal{O}_n(\mathbb{Z}), \mathbf{U} \leftarrow \mathcal{GL}_k(\mathbb{Z}_q)$

$\mathbf{H}_0 \xleftarrow{\$} \mathbb{Z}_q^{n \times k}$

$\mathbf{H}_1 = \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$

Knowing $\mathbf{G}$ and $\mathbf{H}_i$, find i.

LIP

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times n}$,

$\mathbf{O} \leftarrow \mathcal{O}_n(\mathbb{R}), \mathbf{U} \leftarrow \mathcal{GL}_n(\mathbb{Z})$

$\mathbf{B}_0 \leftarrow \mathbb{Z}_q^{n \times n}$

$\mathbf{B}_1 = \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$

Knowing $\mathbf{A}$ and $\mathbf{B}_i$, find i

. . . and codes

Assumptions

Permutation Code Equivalence

$\mathbf{G} \xleftarrow{\$} \{\mathbf{G}\ h\text{-hollow}\} \subseteq \mathbb{Z}_q^{n \times k}$

$\mathbf{O} \leftarrow \mathcal{O}_n(\mathbb{Z}), \mathbf{U} \leftarrow \mathcal{GL}_k(\mathbb{Z}_q)$

$\mathbf{H}_0 \xleftarrow{\$} \{\mathbf{H}\ h\text{-hollow}\} \subseteq \mathbb{Z}_q^{n \times k}$

$\mathbf{H}_1 = \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$

Knowing $\mathbf{G}$ and $\mathbf{H}_i$, find i.

Property

PCE hard when h is big

[Sendrier and Simos, 2013]

Efficient h-hollow sampling and self-orthogonal sampling

[Albrecht, Benčina, and Lai, 2025]

LWE and Hollow-LWE

Learning With Errors

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}$

$\mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \mathcal{D}_\sigma^m$

$\mathbf{b}_0 \leftarrow \mathbb{Z}_q^m$

$\mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}$

Knowing $\mathbf{A}$ and b_i, find i

LWE and Hollow-LWE

Learning With Errors

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}$

$\mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \mathcal{D}_\sigma^m$

$\mathbf{b}_0 \leftarrow \mathbb{Z}_q^m$

$\mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}$

Knowing $\mathbf{A}$ and b_i, find i

Hollow-LWE

$\mathbf{A} \xleftarrow{\$} \{h\text{-hollow matrices}\} \subseteq \mathbb{Z}_q^{m \times n}$

$\mathbf{s} \leftarrow \mathcal{U}(\mathbb{Z}_q^n), \mathbf{e} \leftarrow \mathcal{D}_\sigma^m$

$\mathbf{b}_0 \leftarrow \mathbb{Z}_q^m$

$\mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}$

Knowing $\mathbf{A}$ and b_i, find i

Dual-Regev PKE Cryptosystem

Keygen

$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times k}$

$\mathbf{r} \xleftarrow{\$} \{-1, 1\}^n$

$\mathbf{u} := \mathbf{r}\mathbf{A} \mod q$

$\mathsf{pk} = (\mathbf{A}, \mathbf{u})$

$\mathsf{sk} = \mathbf{r}$

Encrypt$(\mathsf{pk}, m \in \{0,1\})$

$\mathbf{x} \xleftarrow{\$} \mathbb{Z}_q^k, \mathbf{e} \leftarrow \mathcal{D}_\sigma^n, e' \leftarrow \mathcal{D}_\sigma$

$\mathbf{c}_0 := \mathbf{A}\mathbf{x} + \mathbf{e} \mod q$

$c_1 = \langle \mathbf{u}, \mathbf{x} \rangle + e' + \lfloor \frac{q}{2} \cdot m \rceil$

Return $\mathbf{c} = (\mathbf{c}_0, c_1)$

Decrypt$(\mathsf{sk}, \mathbf{c} = (\mathbf{c}_0, c_1))$

Return

$\lfloor \frac{2}{q} \cdot (c_1 - \langle \mathbf{r}, \mathbf{c}_0 \mod q \rangle) \rceil$

Security

Smallness of secrets $\Rightarrow$ correctness

LWE $\Rightarrow$ IND-CPA

Dual-Regev UPKE Cryptosystem

UPKE ?

Updatable Public Key Encryption: PKE + Updatables keys

UpdatePK$(\text{pk} = (\mathbf{A}, \mathbf{u}))$

$\mathbf{O} \xleftarrow{\$} \mathcal{O}_n(\mathbb{Z})$

$(\mathbf{A}', \mathbf{U}) = \text{SF}(\mathbf{O} \cdot \mathbf{A})$

(such that $\mathbf{OAU} = \mathbf{A}'$)

pk' $:= (\mathbf{A}', \mathbf{u}^\top \cdot \mathbf{U})$

token $\leftarrow$ **Enc**$(\text{pk}, \mathbf{O})$

Return (pk',token)

UpdateSK$(\text{sk} = \mathbf{r}, \text{token})$

$\mathbf{O} \leftarrow$ **Dec**$(\text{sk}, \text{token})$

Return $\mathbf{O} \cdot \mathbf{r}$

Dual-Regev UPKE Cryptosystem

UPKE ?

Updatable Public Key Encryption: PKE + Updatables keys

UpdatePK$(\text{pk} = (\mathbf{A}, \mathbf{u}))$

$\mathbf{O} \xleftarrow{\$} \mathcal{O}_n(\mathbb{Z})$

$(\mathbf{A}', \mathbf{U}) = \text{SF}(\mathbf{O} \cdot \mathbf{A})$

(such that $\mathbf{OAU} = \mathbf{A}'$)

$\text{pk}' := (\mathbf{A}', \mathbf{u}^\top \cdot \mathbf{U})$

$\text{token} \leftarrow$ **Enc**$(\text{pk}, \mathbf{O})$

Return (pk′,token)

UpdateSK$(\text{sk} = \mathbf{r}, \text{token})$

$\mathbf{O} \leftarrow$ **Dec**$(\text{sk}, \text{token})$

Return $\mathbf{O} \cdot \mathbf{r}$

Security

Restrict to $\mathbf{A} \xleftarrow{\$} \{h\text{-hollow matrices}\}$

Hollow-LWE + SPCE $\Rightarrow$ IND-CR-CPA

Going further

Can we construct (other) useful lattice-based assumptions amplified by code ?

- Two approaches of constructing relaxed assumptions:
  1. Give additionnal information as hints.
  2. Modify the original assumption (distributions, sets, operations, ...)
- Do you have cool structures ? (like Hull in the previous example)
- Can we construct useful code-based assumptions amplified by lattice tools ?

References I

Albrecht, Martin R., Benjamin Benčina, and Russell W. F. Lai (2025). “Hollow LWE: A New Spin - Unbounded Updatable Encryption from LWE and PCE”. Pp. 363–392. DOI: 10.1007/978-3-031-91101-9_13.

Sendrier, Nicolas and Dimitris E. Simos (2013). “The Hardness of Code Equivalence over and Its Application to Code-Based Cryptography”. Pp. 203–216. DOI: 10.1007/978-3-642-38616-9_14.