# HOLLOW LWE: A NEW SPIN

## UNBOUNDED UPDATABLE ENCRYPTION FROM LWE AND PCE

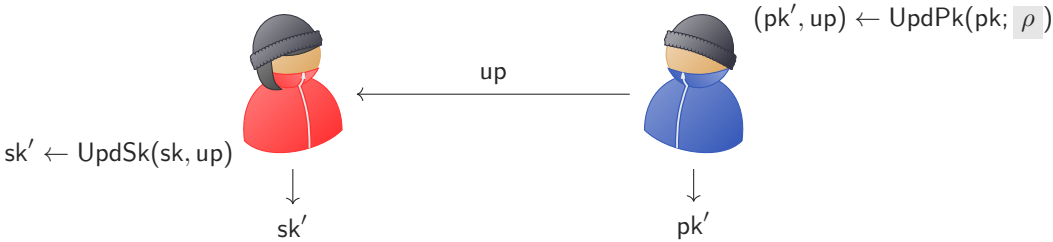
M. R. Albrecht, B. Benčina, and R. W. F. Lai

King's College London and SanboxAQ, Royal Holloway, Univesity of London, and Aalto University

EUROCRYPT 2025, eprint: ia.cr/2025/340

# UPDATABLE PUBLIC-KEY ENCRYPTION (UPKE)

Let $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be a correct PKE scheme.



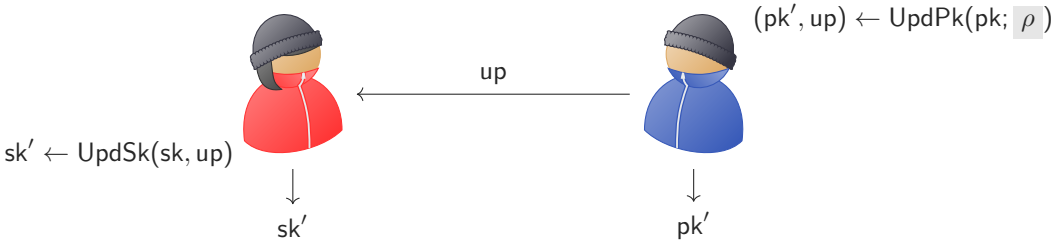

- Update correctness: Dec. cor. holds for updated keys $(\mathsf{pk}', \mathsf{sk}')$.
- IND-CR-CPA Security:

$$(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{msg}_0), \mathsf{pk}', \mathsf{sk}', \mathsf{up}) \approx_c (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{msg}_1), \mathsf{pk}', \mathsf{sk}', \mathsf{up})$$

$\implies$ "forward secrecy."

# UPDATABLE PUBLIC-KEY ENCRYPTION (UPKE)

Let $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be a correct PKE scheme.




- Update correctness: Dec. cor. holds for updated keys $(\mathsf{pk}', \mathsf{sk}')$.
- IND-CR-CPA Security:

$$(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{msg}_0), \mathsf{pk}', \mathsf{sk}', \mathsf{up}) \overset{c}{\approx} (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mathsf{msg}_1), \mathsf{pk}', \mathsf{sk}', \mathsf{up})$$

$\implies$ "forward secrecy."

# DUAL-REGEV ENCRYPTION [REG05, GPV08]

| $\mathsf{KGen}(1^\lambda)$ | $\mathsf{Enc}(\mathsf{pk}, \mathsf{msg} \in \{0,1\})$ |
|---|---|
| $\mathbf{A} \leftarrow\$ \mathbb{Z}_q^{n \times k}$ | $\mathbf{x} \leftarrow\$ \mathbb{Z}_q^k; \quad \mathbf{e} \leftarrow\$ \chi^n; \quad e' \leftarrow\$ \chi$ |
| $\mathbf{r} \leftarrow\$ \{\pm 1\}^n$ | $\mathbf{c}_0 := \mathbf{A} \cdot \mathbf{x} + \mathbf{e} \bmod q$ |
| $\mathbf{u}^\mathsf{T} := \mathbf{r}^\mathsf{T} \cdot \mathbf{A} \bmod q$ | $c_1 := \langle \mathbf{u}, \mathbf{x} \rangle + e' + \lfloor \frac{q}{2} \rceil \cdot \mathsf{msg} \bmod q$ |
| $\mathsf{pk} := (\mathbf{A}, \mathbf{u})$ | **return** $\mathsf{ctxt} := (\mathbf{c}_0, c_1)$ |
| $\mathsf{sk} := \mathbf{r}$ | |
| **return** $(\mathsf{pk}, \mathsf{sk})$ | $\mathsf{Dec}(\mathsf{sk}, \mathsf{ctxt})$ |
| | **return** $\left\lfloor \frac{2}{q} \cdot (c_1 - \langle \mathbf{r}, \mathbf{c}_0 \rangle \bmod q) \right\rceil$ |

- Correctness: $\mathbf{r}$, $\mathbf{e}$, $e'$ are short enough $\implies$ Dual-Regev has decryption correctness.
- Security: LWE assumption $\implies$ Dual-Regev is IND-CPA secure.

# DUAL-REGEV ENCRYPTION [REG05, GPV08]

| $\mathsf{KGen}(1^\lambda)$ | $\mathsf{Enc}(\mathsf{pk}, \mathsf{msg} \in \{0,1\})$ |
|---|---|
| $\mathbf{A} \leftarrow\$ \mathbb{Z}_q^{n \times k}$ | $\mathbf{x} \leftarrow\$ \mathbb{Z}_q^k; \quad \mathbf{e} \leftarrow\$ \chi^n; \quad e' \leftarrow\$ \chi$ |
| $\mathbf{r} \leftarrow\$ \{\pm 1\}^n$ | $\mathbf{c}_0 := \mathbf{A} \cdot \mathbf{x} + \mathbf{e} \bmod q$ |
| $\mathbf{u}^\mathsf{T} := \mathbf{r}^\mathsf{T} \cdot \mathbf{A} \bmod q$ | $c_1 := \langle \mathbf{u}, \mathbf{x} \rangle + e' + \lfloor \frac{q}{2} \rceil \cdot \mathsf{msg} \bmod q$ |
| $\mathsf{pk} := (\mathbf{A}, \mathbf{u})$ | **return** $\mathsf{ctxt} := (\mathbf{c}_0, c_1)$ |
| $\mathsf{sk} := \mathbf{r}$ | |
| **return** $(\mathsf{pk}, \mathsf{sk})$ | $\mathsf{Dec}(\mathsf{sk}, \mathsf{ctxt})$ |
| | **return** $\left\lfloor \frac{2}{q} \cdot (c_1 - \langle \mathbf{r}, \mathbf{c}_0 \rangle \bmod q) \right\rceil$ |

- Correctness: $\mathbf{r}$, $\mathbf{e}$, $e'$ are short enough $\implies$ Dual-Regev has decryption correctness.
- Security: LWE assumption $\implies$ Dual-Regev is IND-CPA secure.

# PRIOR KEY-UPDATE MECHANISM [DKW21]

| UpdPk(pk) | UpdSk(sk, up) |
|---|---|
| $(\mathbf{A}, \mathbf{u}) \leftarrow \mathsf{pk}$ | $\mathbf{r} \leftarrow \mathsf{sk}$ |
| $\delta \leftarrow\!\$\ \chi_{\mathbf{r}}^{n}$ | $\delta \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{up})$ |
| $\mathsf{pk}' := (\mathbf{A}, \mathbf{u}^{\mathsf{T}} + \delta^{\mathsf{T}} \cdot \mathbf{A})$ | $\mathsf{sk}' := \mathbf{r} + \delta$ |
| $\mathsf{up} \leftarrow \mathsf{Enc}(\mathsf{pk}, \delta)$ | **return** $\mathsf{sk}'$ |
| **return** $(\mathsf{pk}', \mathsf{up})$ | |

**Issues:**

- Updated secret key $\mathbf{r}' = \mathbf{r} + \delta$ has increased norm.
- To maintain correctness with many updates, either
  - restrict number of updates to be fixed a-priori, or
  - for poly($\lambda$) many updates, set super-poly. modulus $q > \lambda^{\omega(1)} \implies$ large ctxt.

# PRIOR KEY-UPDATE MECHANISM [DKW21]

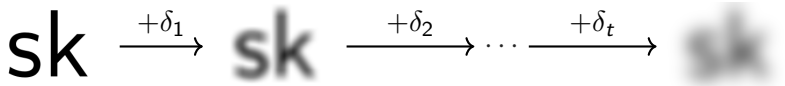
| UpdPk(pk) | UpdSk(sk, up) |
|---|---|
| $(\mathbf{A}, \mathbf{u}) \leftarrow \mathsf{pk}$ | $\mathbf{r} \leftarrow \mathsf{sk}$ |
| $\delta \leftarrow\!\$\ \chi_{\mathbf{r}}^{n}$ | $\delta \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{up})$ |
| $\mathsf{pk}' := (\mathbf{A}, \mathbf{u}^{\mathsf{T}} + \delta^{\mathsf{T}} \cdot \mathbf{A})$ | $\mathsf{sk}' := \mathbf{r} + \delta$ |
| $\mathsf{up} \leftarrow \mathsf{Enc}(\mathsf{pk}, \delta)$ | **return** $\mathsf{sk}'$ |
| **return** $(\mathsf{pk}', \mathsf{up})$ | |

**Issues:**

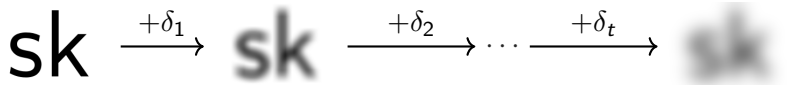
- Updated secret key $\mathbf{r}' = \mathbf{r} + \delta$ has increased norm.
- To maintain correctness with many updates, either
  - restrict number of updates to be fixed a-priori, or
  - for poly($\lambda$) many updates, set super-poly. modulus $q > \lambda^{\omega(1)} \implies$ large ctxt.

**What if we rotate keys instead?**

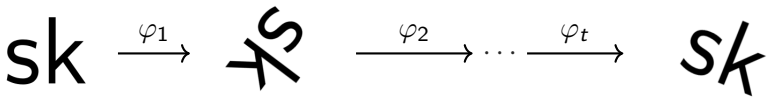
**Prior method: Adding noise**



**Prior method: Adding noise**




**Our Approach: Rotating keys**



# $q$-ARY LATTICES

A lattice $\Lambda \subseteq \mathbb{R}^n$ is a discrete additive subgroup of $\mathbb{R}^n$, i.e.

$$\Lambda = \mathbf{B} \cdot \mathbb{Z}^k$$

for some basis $\mathbf{B} \in \mathbb{R}^{n \times k}$ where $k \leq n$. All bases $\mathbf{B}, \mathbf{B}' \in \mathbb{R}^{n \times k}$ are related by unimodular $\mathbf{U} \in \mathbb{Z}^{k \times k}$ via $\mathbf{B}' = \mathbf{B} \cdot \mathbf{U}$.

Define the Construction A lattice of a full-rank $\mathbf{A} \in \mathbb{Z}_q^{n \times k}$ as

$$\Lambda_q(\mathbf{A}) = \mathbf{A} \cdot \mathbb{Z}^k + q \cdot \mathbb{Z}^n.$$

Note that $\Lambda_q(\mathbf{A})$ is $q$-ary, i.e.

$$q \cdot \mathbb{Z}^n \subseteq \Lambda_q(\mathbf{A}) \subseteq \mathbb{Z}^n.$$

## $q$-ARY LATTICES

A lattice $\Lambda \subseteq \mathbb{R}^n$ is a discrete additive subgroup of $\mathbb{R}^n$, i.e.

$$\Lambda = \mathbf{B} \cdot \mathbb{Z}^k$$

for some basis $\mathbf{B} \in \mathbb{R}^{n \times k}$ where $k \leq n$. All bases $\mathbf{B}, \mathbf{B}' \in \mathbb{R}^{n \times k}$ are related by unimodular $\mathbf{U} \in \mathbb{Z}^{k \times k}$ via $\mathbf{B}' = \mathbf{B} \cdot \mathbf{U}$.

Define the Construction A lattice of a full-rank $\mathbf{A} \in \mathbb{Z}_q^{n \times k}$ as

$$\Lambda_q(\mathbf{A}) = \mathbf{A} \cdot \mathbb{Z}^k + q \cdot \mathbb{Z}^n.$$

Note that $\Lambda_q(\mathbf{A})$ is $q$-ary, i.e.

$$q \cdot \mathbb{Z}^n \subseteq \Lambda_q(\mathbf{A}) \subseteq \mathbb{Z}^n.$$

# LWE AND DUAL-REGEV: LATTICE POINT OF VIEW

LWE assumption: for $\mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n\times k}$, $\mathbf{x} \leftarrow\!\$\ \mathbb{Z}_q^k$, $\mathbf{e} \leftarrow\!\$\ \chi^n$, $\mathbf{u} \leftarrow\!\$\ \mathbb{Z}_q^n$ we have

$$(\mathbf{A}, \mathbf{A}\cdot\mathbf{x} + \mathbf{e} \bmod q) \stackrel{c}{\approx} (\mathbf{A}, \mathbf{u})\,.$$

Lattice point of view: $(\mathbf{A}, \mathcal{U}(\Lambda_q(\mathbf{A})) + \chi^n) \stackrel{c}{\approx} (\mathbf{A}, \mathcal{U}(\mathbb{Z}_q^n))\,.$

A Dual-Regev secret key is a short vector

$$\mathbf{r} \in \Lambda_q^{\mathbf{u}}(\mathbf{A}) \coloneqq \left\{\mathbf{x} \in \mathbb{Z}^n : \mathbf{x}^{\mathrm{T}}\cdot\mathbf{A} = \mathbf{u}^{\mathrm{T}} \bmod q\right\}$$

which is a random lattice coset (defined by $\mathbf{u}$) of the kernel lattice

$$\Lambda_q^{\perp}(\mathbf{A}) \coloneqq \left\{\mathbf{x} \in \mathbb{Z}^n : \mathbf{x}^{\mathrm{T}}\cdot\mathbf{A} = \mathbf{0}^{\mathrm{T}} \bmod q\right\}.$$

# LATTICE ISOMORPHISM PROBLEM (LIP)

**Lattice Isomorphism:** Lattices $\Lambda, \Lambda'$ are isomorphic, denoted $\Lambda \sim \Lambda'$, if there exists an orthogonal matrix $\mathbf{O} \in \mathcal{O}_n(\mathbb{R})$, i.e.

$$\mathbf{O} \in \mathbb{R}^{n \times n} \qquad \text{with} \qquad \mathbf{O}^{\mathrm{T}} \cdot \mathbf{O} = \mathbf{I}_n,$$

such that

$$\Lambda' = \mathbf{O} \cdot \Lambda,$$

i.e. $\Lambda'$ can be obtained by rotating and reflecting $\Lambda$. If $\mathbf{B}$ and $\mathbf{B}'$ are bases of $\Lambda$ and $\Lambda'$, then it means $\mathbf{B}' = \mathbf{O} \cdot \mathbf{B} \cdot \mathbf{U}$ for some unimodular $\mathbf{U} \in \mathbb{Z}^{k \times k}$.

**Lattice Isomorphism Problem ($\Delta$LIP) [DvW22]:** Given lattices $\Lambda_0, \Lambda_1, \Lambda \subseteq \mathbb{R}^n$, decide if

$$\Lambda \sim \Lambda_0 \qquad \text{or} \qquad \Lambda \sim \Lambda_1.$$

# LATTICE ISOMORPHISM PROBLEM (LIP)

**Lattice Isomorphism:** Lattices $\Lambda, \Lambda'$ are isomorphic, denoted $\Lambda \sim \Lambda'$, if there exists an orthogonal matrix $\mathbf{O} \in \mathcal{O}_n(\mathbb{R})$, i.e.

$$\mathbf{O} \in \mathbb{R}^{n \times n} \qquad \text{with} \qquad \mathbf{O}^{\mathrm{T}} \cdot \mathbf{O} = \mathbf{I}_n,$$

such that

$$\Lambda' = \mathbf{O} \cdot \Lambda,$$

i.e. $\Lambda'$ can be obtained by rotating and reflecting $\Lambda$. If $\mathbf{B}$ and $\mathbf{B}'$ are bases of $\Lambda$ and $\Lambda'$, then it means $\mathbf{B}' = \mathbf{O} \cdot \mathbf{B} \cdot \mathbf{U}$ for some unimodular $\mathbf{U} \in \mathbb{Z}^{k \times k}$.

**Lattice Isomorphism Problem ($\Delta$LIP) [DvW22]:** Given lattices $\Lambda_0, \Lambda_1, \Lambda \subseteq \mathbb{R}^n$, decide if

$$\Lambda \sim \Lambda_0 \qquad \text{or} \qquad \Lambda \sim \Lambda_1.$$

# ROTATE KEYS WITH LIP?

**The idea, more concretely:**

- Rotate the lattice: $\mathbf{A} \mapsto \mathbf{A}' \coloneqq \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U} \bmod q$.
- Rotate the key: $\mathbf{r} \mapsto \mathbf{r}' \coloneqq \mathbf{O} \cdot \mathbf{r} \bmod q$.
- Update the syndrome: $\mathbf{u} \mapsto \mathbf{u}' \coloneqq \mathbf{U}^{\mathrm{T}} \cdot \mathbf{u} \bmod q$, so that:

$$\mathbf{r}^{\mathrm{T}} \cdot \mathbf{A} = \mathbf{u}^{\mathrm{T}} \qquad \Longrightarrow \qquad \mathbf{r}'^{\mathrm{T}} \cdot \mathbf{A}' = \mathbf{u}'^{\mathrm{T}}$$

One can think of it as re-randomising a SIS commitment.

**Upshot:** $\|\mathbf{r}'\|_2 = \sqrt{\langle \mathbf{O} \cdot \mathbf{r}, \mathbf{O} \cdot \mathbf{r} \rangle} = \sqrt{\langle \mathbf{r}, \mathbf{r} \rangle} = \|\mathbf{r}\|_2$.

**Issue:** Orthogonal $\mathbf{O} \in \mathcal{O}_n(\mathbb{R})$ are real-valued $\Longrightarrow$ $\mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$ and $\mathbf{O} \cdot \mathbf{r}$ may not be integral.

# ROTATE KEYS WITH LIP?

**The idea, more concretely:**

- Rotate the lattice: $\mathbf{A} \mapsto \mathbf{A}' \coloneqq \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U} \bmod q$.
- Rotate the key: $\mathbf{r} \mapsto \mathbf{r}' \coloneqq \mathbf{O} \cdot \mathbf{r} \bmod q$.
- Update the syndrome: $\mathbf{u} \mapsto \mathbf{u}' \coloneqq \mathbf{U}^{\mathrm{T}} \cdot \mathbf{u} \bmod q$, so that:

$$\mathbf{r}^{\mathrm{T}} \cdot \mathbf{A} = \mathbf{u}^{\mathrm{T}} \qquad \Longrightarrow \qquad \mathbf{r}'^{\mathrm{T}} \cdot \mathbf{A}' = \mathbf{u}'^{\mathrm{T}}$$

One can think of it as re-randomising a SIS commitment.

**Upshot:** $\|\mathbf{r}'\|_2 = \sqrt{\langle \mathbf{O} \cdot \mathbf{r}, \mathbf{O} \cdot \mathbf{r} \rangle} = \sqrt{\langle \mathbf{r}, \mathbf{r} \rangle} = \|\mathbf{r}\|_2$.

**Issue:** Orthogonal $\mathbf{O} \in \mathcal{O}_n(\mathbb{R})$ are real-valued $\Longrightarrow$ $\mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$ and $\mathbf{O} \cdot \mathbf{r}$ may not be integral.

# ROTATE KEYS WITH LIP?

**The idea, more concretely:**

- Rotate the lattice: $\mathbf{A} \mapsto \mathbf{A}' \coloneqq \mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U} \bmod q$.
- Rotate the key: $\mathbf{r} \mapsto \mathbf{r}' \coloneqq \mathbf{O} \cdot \mathbf{r} \bmod q$.
- Update the syndrome: $\mathbf{u} \mapsto \mathbf{u}' \coloneqq \mathbf{U}^{\mathrm{T}} \cdot \mathbf{u} \bmod q$, so that:

$$\mathbf{r}^{\mathrm{T}} \cdot \mathbf{A} = \mathbf{u}^{\mathrm{T}} \qquad \Longrightarrow \qquad \mathbf{r}'^{\mathrm{T}} \cdot \mathbf{A}' = \mathbf{u}'^{\mathrm{T}}$$

One can think of it as re-randomising a SIS commitment.

**Upshot:** $\|\mathbf{r}'\|_2 = \sqrt{\langle \mathbf{O} \cdot \mathbf{r}, \mathbf{O} \cdot \mathbf{r} \rangle} = \sqrt{\langle \mathbf{r}, \mathbf{r} \rangle} = \|\mathbf{r}\|_2$.

**Issue:** Orthogonal $\mathbf{O} \in \mathcal{O}_n(\mathbb{R})$ are real-valued $\implies$ $\mathbf{O} \cdot \mathbf{A} \cdot \mathbf{U}$ and $\mathbf{O} \cdot \mathbf{r}$ may not be integral.

# LATTICE AUTOMORPHISM OF $\mathbb{Z}^n$

- The automorphism group $\mathrm{Aut}(\Lambda)$ of a lattice $\Lambda$ is the group of all isomorphisms from $\Lambda$ to itself.

- It is well-known that $\mathrm{Aut}(\mathbb{Z}^n) = \mathcal{O}_n(\mathbb{Z})$, i.e. the group of signed permutations

$$\mathcal{O}_n(\mathbb{Z}) = \{\mathbf{D} \cdot \mathbf{P} \ ; \ \mathbf{D} \in \mathrm{diag}(\{\pm 1\}^n), \ \mathbf{P} \in \mathcal{P}_n\}.$$

- Since

$$q \cdot \mathbb{Z}^n \subseteq \Lambda_q(\mathbf{A}) \subseteq \mathbb{Z}^n,$$

we have

$$q \cdot \mathbb{Z}^n \subseteq \mathbf{O} \cdot \Lambda_q(\mathbf{A}) = \Lambda_q(\mathbf{O} \cdot \mathbf{A}) \subseteq \mathbb{Z}^n,$$

i.e. rotating $\Lambda_q(\mathbf{A})$ by $\mathbf{O} \in \mathcal{O}_n(\mathbb{Z})$ gives another $q$-ary lattice.

# LATTICE AUTOMORPHISM OF $\mathbb{Z}^n$

- The automorphism group $\mathrm{Aut}(\Lambda)$ of a lattice $\Lambda$ is the group of all isomorphisms from $\Lambda$ to itself.

- It is well-known that $\mathrm{Aut}(\mathbb{Z}^n) = \mathcal{O}_n(\mathbb{Z})$, i.e. the group of signed permutations

$$\mathcal{O}_n(\mathbb{Z}) = \{\mathbf{D} \cdot \mathbf{P} \ ; \ \mathbf{D} \in \mathrm{diag}(\{\pm 1\}^n), \ \mathbf{P} \in \mathcal{P}_n\}.$$

- Since

$$q \cdot \mathbb{Z}^n \subseteq \Lambda_q(\mathbf{A}) \subseteq \mathbb{Z}^n,$$

  we have

$$q \cdot \mathbb{Z}^n \subseteq \mathbf{O} \cdot \Lambda_q(\mathbf{A}) = \Lambda_q(\mathbf{O} \cdot \mathbf{A}) \subseteq \mathbb{Z}^n,$$

  i.e. rotating $\Lambda_q(\mathbf{A})$ by $\mathbf{O} \in \mathcal{O}_n(\mathbb{Z})$ gives another $q$-ary lattice.

# CODING THEORY POINT OF VIEW

- The Construction A lattice of $\mathbf{A} \in \mathbb{Z}_q^{n \times k}$ defined by $\Lambda_q(\mathbf{A}) = \mathbf{A} \cdot \mathbb{Z}^k + q \cdot \mathbb{Z}^n$ is isomorphic to the $[n, k]$-linear code $\mathcal{C} = \mathbf{A} \cdot \mathbb{Z}_q^k$ over $\mathbb{Z}_q$ generated by $\mathbf{A}$.

- The (Signed) Permutation Code Equivalence ((S)PCE) problem is to decide if two codes $\mathfrak{C}$ and $\mathfrak{C}'$ are (signed) permutation equivalent, i.e. whether

$$\mathfrak{C}' = \mathbf{O} \cdot \mathfrak{C}$$

  for some (signed) permutation matrix $\mathbf{O} \in \mathcal{O}_n(\mathbb{Z})$.

- SPCE is essentially decision LIP with $\Lambda$'s restricted to $q$-ary lattices and $\mathbf{O}$'s restricted to signed permutations.

# CODING THEORY POINT OF VIEW

- The Construction A lattice of $\mathbf{A} \in \mathbb{Z}_q^{n \times k}$ defined by $\Lambda_q(\mathbf{A}) = \mathbf{A} \cdot \mathbb{Z}^k + q \cdot \mathbb{Z}^n$ is isomorphic to the $[n, k]$-linear code $\mathcal{C} = \mathbf{A} \cdot \mathbb{Z}_q^k$ over $\mathbb{Z}_q$ generated by $\mathbf{A}$.

- The (Signed) Permutation Code Equivalence ((S)PCE) problem is to decide if two codes $\mathfrak{C}$ and $\mathfrak{C}'$ are (signed) permutation equivalent, i.e. whether

$$\mathfrak{C}' = \mathbf{O} \cdot \mathfrak{C}$$

  for some (signed) permutation matrix $\mathbf{O} \in \mathcal{O}_n(\mathbb{Z})$.

- SPCE is essentially decision LIP with $\Lambda$'s restricted to $q$-ary lattices and $\mathbf{O}$'s restricted to signed permutations.

# CODING THEORY POINT OF VIEW

- The Construction A lattice of $\mathbf{A} \in \mathbb{Z}_q^{n \times k}$ defined by $\Lambda_q(\mathbf{A}) = \mathbf{A} \cdot \mathbb{Z}^k + q \cdot \mathbb{Z}^n$ is isomorphic to the $[n, k]$-linear code $\mathcal{C} = \mathbf{A} \cdot \mathbb{Z}_q^k$ over $\mathbb{Z}_q$ generated by $\mathbf{A}$.

- The (Signed) Permutation Code Equivalence ((S)PCE) problem is to decide if two codes $\mathfrak{C}$ and $\mathfrak{C}'$ are (signed) permutation equivalent, i.e. whether

$$\mathfrak{C}' = \mathbf{O} \cdot \mathfrak{C}$$

  for some (signed) permutation matrix $\mathbf{O} \in \mathcal{O}_n(\mathbb{Z})$.

- SPCE is essentially decision LIP with $\Lambda$'s restricted to $q$-ary lattices and $\mathbf{O}$'s restricted to signed permutations.

# PCE-BASED KEY-UPDATE MECHANISM

| UpdPk(pk) | UpdSk(sk, up) |
|---|---|
| $(\mathbf{A}, \mathbf{u}) \leftarrow \mathsf{pk}$ | $\mathbf{r} \leftarrow \mathsf{sk}$ |
| $\mathbf{O} \leftarrow\!\!\$\ \mathcal{O}_n(\mathbb{Z})$ | $\mathbf{O} \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{up})$ |
| $\mathbf{A}', \mathbf{U} := \mathsf{SF}(\mathbf{O} \cdot \mathbf{A})$ | $\mathsf{sk}' := \mathbf{O} \cdot \mathbf{r}$ |
| $\mathsf{pk}' := (\mathbf{A}', \mathbf{u}^{\mathrm{T}} \cdot \mathbf{U})$ | **return** $\mathsf{sk}'$ |
| $\mathsf{up} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathbf{O})$ | |
| **return** $(\mathsf{pk}', \mathsf{up})$ | |

Update correctness:

$$\mathbf{r'}^{\mathrm{T}} \cdot \mathbf{A}' = \mathbf{r}^{\mathrm{T}} \cdot \underbrace{\mathbf{O}^{\mathrm{T}} \cdot \mathbf{O}}_{\mathbf{I}_n} \cdot \mathbf{A} \cdot \mathbf{U} = \underbrace{\mathbf{r}^{\mathrm{T}} \cdot \mathbf{A}}_{\mathbf{u}^{\mathrm{T}}} \cdot \mathbf{U} = \mathbf{u}^{\mathrm{T}} \cdot \mathbf{U} = \mathbf{u'}^{\mathrm{T}} \pmod{q}.$$

## CAUTION – MIND THE HULL

- The hardness of (S)PCE, depends on the hull of the code $\mathfrak{C} = \mathbf{A} \cdot \mathbb{Z}_q^k$.
- The hull $\mathcal{H}(\mathfrak{C}) := \mathfrak{C} \cap \mathfrak{C}^\perp = \{\mathbf{x} \in \mathfrak{C} : \mathbf{x}^\mathsf{T} \cdot \mathcal{C} = \mathbf{0}\}$ is a subcode of $\mathfrak{C}$.
- Random codes have small hull dimension [Sen97], most likely 0.
- Existing attacks against (S)PCE run in time $\mathcal{O}(q^h \cdot \mathrm{poly}(n,k))$ or $\mathcal{O}(n^h \cdot \mathrm{poly}(n,k,q))$, i.e. efficient when $h$ is small [Sen00, BOST19].

**SampleCode$(n,k,h,q)$**

We give an algorithm SampleCode$(n,k,h,q)$ that samples $\mathbf{A}$ generating a uniformly random $[n,k]$-linear code over $\mathbb{Z}_q$ with hull dimension $h$. We call such codes and matrices "$h$-hollow".

## CAUTION – MIND THE HULL

- The hardness of (S)PCE, depends on the hull of the code $\mathfrak{C} = \mathbf{A} \cdot \mathbb{Z}_q^k$.
- The hull $\mathcal{H}(\mathfrak{C}) := \mathfrak{C} \cap \mathfrak{C}^\perp = \{\mathbf{x} \in \mathfrak{C} : \mathbf{x}^\mathsf{T} \cdot \mathcal{C} = \mathbf{0}\}$ is a subcode of $\mathfrak{C}$.
- Random codes have small hull dimension [Sen97], most likely 0.
- Existing attacks against (S)PCE run in time $\mathcal{O}(q^h \cdot \mathrm{poly}(n,k))$ or $\mathcal{O}(n^h \cdot \mathrm{poly}(n,k,q))$, i.e. efficient when $h$ is small [Sen00, BOST19].

**SampleCode$(n,k,h,q)$**

We give an algorithm SampleCode$(n,k,h,q)$ that samples $\mathbf{A}$ generating a uniformly random $[n,k]$-linear code over $\mathbb{Z}_q$ with hull dimension $h$. We call such codes and matrices "$h$-hollow".

## HOLLOW LATTICE PROBLEMS

**Hollow-LWE:** $\mathbf{A} \leftarrow\!\$\ \mathbb{Z}_q^{n \times k}$ $h$-hollow, $\mathbf{x} \leftarrow\!\$\ \mathbb{Z}_q^k$, $\mathbf{e} \leftarrow\!\$\ \chi^n$, $\mathbf{u} \leftarrow\!\$\ \mathbb{Z}_q^n$, distinguish

$$(\mathbf{A}, \mathbf{A} \cdot \mathbf{x} + \mathbf{e}) \qquad \text{from} \qquad (\mathbf{A}, \mathbf{u}) .$$

**Theorem (LWE $\rightarrow$ Hollow-LWE)**

*If there exists a $(t, \varepsilon)$-algorithm $\mathcal{A}$ for $LWE^h_{k,n,q,\chi}$ then there exists a $(t + \mathrm{poly}(\lambda), \varepsilon')$-algorithm $\mathcal{B}$ for $LWE_{k-h,n,q,\chi}$ where*

$$\epsilon' \geq \varepsilon \cdot \underbrace{\left(1 - \frac{1}{q} - \frac{1}{q^2}\right)}_{\text{triv. hull}} \cdot \underbrace{\left(1 - \frac{h}{e^n}\right)}_{\text{sub-sampler}} \cdot \underbrace{\prod_{i=0}^{k-h} \left(1 - q^{i-n}\right)}_{\text{full rank}} \cdot \underbrace{\prod_{i=1}^{h} \left(1 - q^{k+i-n}\right)}_{\text{lin. dep. in hull}} .$$

**Theorem (Hollow-LHL)**

*Let $n, k, h, q$ integers with*

$$n \geq \underbrace{(1+c) \cdot k \cdot \log_2(q)}_{LHL} + \underbrace{k+h}_{extra}$$

*for a positive real constant $c > 0$, $h \leq \frac{k}{2}$, and $q$ an odd prime. Let $\mathbf{A} \hookleftarrow\$ \mathbb{Z}_q^{n \times k}$ h-hollow matrix, $\mathbf{r} \hookleftarrow\$ \{\pm 1\}^n$, and $\mathbf{u} \hookleftarrow\$ \mathbb{Z}_q^k$. Then the pairs*

$$(\mathbf{A}, \mathbf{r}^{\mathrm{T}} \cdot \mathbf{A}) \qquad \textit{and} \qquad (\mathbf{A}, \mathbf{u}^{\mathrm{T}})$$

*are statistically close in $k$.*

## A NEW UPKE SCHEME

Our construction is the Dual-Regev PKE with

- $\mathbf{A} \leftarrow\!\$\ \mathsf{SampleCode}(n, k, h, q)$,
- $\mathbf{r} \leftarrow\!\$\ \{\pm 1\}^n$, and
- the above PCE-based update mechanism.

**Theorem**

*Let $n, k, h, q$ be positive integers parametrised by $\lambda$ with $n \geq (1 + c) \cdot k \cdot \log_2(q) + k + h$ for a positive real constant $c > 0$, $2 \cdot h \leq k$ and $q$ prime.*
*Assuming the advantage of any* PPT *adversary in distinguishing $LWE^h_{k,n,q,\chi}$ and in distinguishing $PCE^h_{n,k,q}$ is negligible in $\lambda$, our construction is IND-CR-CPA secure in the ROM.*

**Table 1:** Parameters for the given $\lambda$ and $p$ with $c = 0.25$ and $s = 8$.

| $\lambda$ | $p$ | $n$ | $k$ | $\log_2(q)$ | $h$ | $\lvert\text{ctxt}\rvert$ | $\lvert\text{up}\rvert$ |
|---|---|---|---|---|---|---|---|
| 128 | 2 | 7313 | 450 | 13 | 27 | 11.6 KiB | 1485.7 KiB |
| 128 | 16 | 11000 | 550 | 16 | 26 | 21.5 KiB | 687.6 KiB |
| 192 | 32 | 20250 | 900 | 18 | 37 | 44.5 KiB | 1708.7 KiB |
| 256 | 32 | 29688 | 1250 | 19 | 48 | 68.9 KiB | 3525.6 KiB |
| [HPS23] with $2^{20}$ updates | | | | | | | |
| 128 | – | – | – | 36 | – | 9.1 KiB | 27 KiB |

# FUTURE WORK

- Replace the Hollow LHL with a computational assumption.
- Switch from LWE to MLWE.
- Consider the model from [AFM24].
- ...

# FUTURE WORK

- Replace the Hollow LHL with a computational assumption.
- Switch from LWE to MLWE.
- Consider the model from [AFM24].
- ...

# FUTURE WORK

- Replace the Hollow LHL with a computational assumption.
- Switch from LWE to MLWE.
- Consider the model from [AFM24].
- ...

# FUTURE WORK

- Replace the Hollow LHL with a computational assumption.
- Switch from LWE to MLWE.
- Consider the model from [AFM24].
- ...

Thank you! Read the full version at ia.cr/2025/340:

Joël Alwen, Georg Fuchsbauer, and Marta Mularczyk.
Updatable public-key encryption, revisited.
In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part VII*, volume 14657 of *LNCS*, pages 346–376. Springer, Cham, May 2024.

Magali Bardet, Ayoub Otmani, and Mohamed Saeed-Taha.
Permutation Code Equivalence is Not Harder Than Graph Isomorphism When Hulls Are Trivial.
In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 2464–2468. IEEE, 2019.

Yevgeniy Dodis, Harish Karthikeyan, and Daniel Wichs.
Updatable public key encryption in the standard model.
In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part III*, volume 13044 of *LNCS*, pages 254–285. Springer, Cham, November 2021.

Léo Ducas and Wessel P. J. van Woerden.
On the lattice isomorphism problem, quadratic forms, remarkable lattices, and cryptography.

In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part III*, volume 13277 of *LNCS*, pages 643–673. Springer, Cham, May / June 2022.


Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan.
Trapdoors for hard lattices and new cryptographic constructions.
In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008.


Calvin Abou Haidar, Alain Passelègue, and Damien Stehlé.
Efficient updatable public-key encryption from lattices.
In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part V*, volume 14442 of *LNCS*, pages 342–373. Springer, Singapore, December 2023.


Oded Regev.
On lattices, learning with errors, random linear codes, and cryptography.
In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005.


Nicolas Sendrier.
On the Dimension of the Hull.

*SIAM Journal on Discrete Mathematics*, 10(2):282–293, 1997.


Nicolas Sendrier.
Finding the permutation between equivalent linear codes: the support splitting algorithm.
*IEEE Transactions on Information Theory*, 46(4):1193–1203, 2000.