Somewhat Homomorphic Encryption based on Random Ideal Codes

Carlos Aguilar-Melchor[1], **Victor Dyseryn**[2], Philippe Gaborit[2]

[1]Sandbox AQ
[2]XLIM, Université de Limoges, France

GT Codes-Crypto - November 20, 2023

Outline

Outline

What is Homomorphic Encryption?

Public-key version

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$
- $\mathsf{Enc}(m, \mathsf{pk}) \to \mathsf{ct}$
- $\mathsf{Dec}(\mathsf{ct}, \mathsf{sk}) \to m$
- $\mathsf{Eval}(f, \mathsf{ct}_1, \mathsf{ct}_2) \to \mathsf{ct}$

Proposition (Correctness)

$$\mathsf{Dec}(\mathsf{Eval}(f, \mathsf{Enc}(m_1, \mathsf{pk}), \mathsf{Enc}(m_2, \mathsf{pk})), \mathsf{sk}) = f(m_1, m_2)$$

What is Homomorphic Encryption?

- $f \in \{+, \times\} \to$ partial homomorphic encryption (RSA)
- $f \in \mathbb{F}_d[X] \to$ somewhat homomorphic encryption [BGN05]
- $f \in \mathbb{F}[X] \to$ fully homomorphic encryption [Gen09]

What is Homomorphic Encryption?

Secret-key version

- $\mathsf{KeyGen}(1^\lambda) \to \mathsf{sk}$
- $\mathsf{Enc}(m, \mathsf{sk}) \to \mathsf{ct}$
- $\mathsf{Dec}(\mathsf{ct}, \mathsf{sk}) \to m$
- $\mathsf{Eval}(f, \mathsf{ct}_1, \mathsf{ct}_2) \to \mathsf{ct}$

Proposition (Correctness)

$$\mathsf{Dec}(\mathsf{Eval}(f, \mathsf{Enc}(m_1, \mathsf{sk}), \mathsf{Enc}(m_2, \mathsf{sk})), \mathsf{sk}) = f(m_1, m_2)$$

Noisy ciphertexts

$$\mathrm{ct}_1 = \boldsymbol{m}_1 \boldsymbol{G} + \boldsymbol{e}_1$$
$$\mathrm{ct}_2 = \boldsymbol{m}_2 \boldsymbol{G} + \boldsymbol{e}_2$$

$$\mathrm{Eval}(+, \mathrm{ct}_1, \mathrm{ct}_2) = (\boldsymbol{m}_1 + \boldsymbol{m}_2)\boldsymbol{G} + \underbrace{\boldsymbol{e}_1 + \boldsymbol{e}_2}_{\text{weight} \approx 2w}$$

In general:

$$\mathrm{Eval}(f, \mathrm{Enc}(m_1, \mathrm{pk}), \mathrm{Enc}(m_2, \mathrm{pk})) \neq \mathrm{Enc}(f(m_1, m_2), \mathrm{pk}).$$

Bootstrapping: how to reduce ciphertext noise

$$(\mathsf{pk}_1, \mathsf{sk}_1) = \mathsf{KeyGen}(1^\lambda)$$
$$\mathsf{ct} = \mathsf{Enc}(m, \mathsf{pk}_1)$$

$$(\mathsf{pk}_2, \mathsf{sk}_2) = \mathsf{KeyGen}(1^\lambda)$$
$$\mathsf{ct}_1 = \mathsf{Enc}(\mathsf{Enc}(m, \mathsf{pk}_1), \mathsf{pk}_2)$$
$$\mathsf{ct}_2 = \mathsf{Enc}(\mathsf{sk}_1, \mathsf{pk}_2)$$

$$\mathsf{Eval}(\mathsf{Dec}(\cdot,\cdot), \mathsf{ct}_1, \mathsf{ct}_2) = \ \textbf{?}$$

Bootstrapping: how to reduce ciphertext noise

$$\mathsf{ct}_1 = \mathsf{Enc}(\mathsf{Enc}(m, \mathsf{pk}_1), \mathsf{pk}_2)$$
$$\mathsf{ct}_2 = \mathsf{Enc}(\mathsf{sk}_1, \mathsf{pk}_2)$$

$$\mathsf{Dec}(\mathsf{Eval}(\mathsf{Dec}(\cdot,\cdot), \mathsf{ct}_1, \mathsf{ct}_2), \mathsf{sk}_2) = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{pk}_1), \mathsf{sk}_1) = m$$

$$\big(\mathsf{Eval}(\mathsf{Dec}(\cdot,\cdot), \mathsf{ct}_1, \mathsf{ct}_2) \approx \mathsf{Enc}(m, \mathsf{pk}_2)\big)$$

History of fully homomorphic encryption

There has been a burst of activity in the last decade:

- 2009: Gentry's first FHE [Gen09]
- 2010-2015: Practical somewhat homomorphic encryption
- 2016: TFHE [CGGI16], bootstrapping below 100ms
- 2016-present: remarkable progress

... but most of existing constructions are based on **structured lattices**.

Outline

1. What is homomorphic encryption?
2. Existing homomorphic encryption with codes
3. New idea: homomorphic encryption with Alekhnovich framework
4. Our construction

Why homomorphic encryption with codes?

- An alternative to structured lattices
- Support and multi-dimensional approach
- Faster and simpler decryption circuit

Multi-dimensional approach for homomorphic encryption

$$\mathrm{ct}_1 = \boldsymbol{m}_1 \boldsymbol{G} \quad + \quad \boldsymbol{e}_1$$
$$\mathrm{ct}_2 = \boldsymbol{m}_2 \boldsymbol{G} \quad + \quad \underbrace{\boldsymbol{e}_2}_{\text{same support}}$$

$$\mathrm{Eval}(+, \mathrm{ct}_1, \mathrm{ct}_2) = (\boldsymbol{m}_1 + \boldsymbol{m}_2)\boldsymbol{G} + \underbrace{\boldsymbol{e}_1 + \boldsymbol{e}_2}_{\text{weight} \leq w}$$

Support Learning problem

Definition ([GHPT17])

Given a parity check matrix $\boldsymbol{H} \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ and ℓ syndromes $\boldsymbol{s}_i = \boldsymbol{e}_i \boldsymbol{H}^T$ for $\boldsymbol{e}_i$ errors of weight w in the same support E, find E.

$\Longrightarrow$ restricts the number of independent ciphertexts than can be published.

Multiplication operation with codes

Technique from [AAPS11]:

$$\begin{aligned} \mathrm{ct}_1 &= \boldsymbol{m}_1 \boldsymbol{G} &+& \quad \boldsymbol{e}_1 \\ \mathrm{ct}_2 &= \boldsymbol{m}_2 \boldsymbol{G} &+& \quad \underbrace{\boldsymbol{e}_2}_{\text{same support}} \end{aligned}$$

$$\begin{aligned} \mathrm{Eval}(\times, \mathrm{ct}_1, \mathrm{ct}_2) &= \mathrm{ct}_1 \odot \mathrm{ct}_2 \\ &= \boldsymbol{m}_1 \boldsymbol{G} \odot \boldsymbol{m}_2 \boldsymbol{G} + \underbrace{\boldsymbol{e}_1 \odot \mathrm{ct}_2 + \boldsymbol{e}_2 \odot \mathrm{ct}_1 - \boldsymbol{e}_1 \odot \boldsymbol{e}_2}_{\text{still in the same support}} \end{aligned}$$

Evaluation codes

Definition

Let $\boldsymbol{g} = (g_1, \ldots, g_n)$ a vector of evaluation points, the evaluation code on $\boldsymbol{g}$ is

$$\mathcal{C} = \{(P(g_1), \ldots, P(g_n)) | P \in \mathcal{L}\}$$

Multiplication operation with evaluation codes

$$\begin{aligned} \mathsf{ct}_1 &= P_1(\boldsymbol{g}) + \boldsymbol{e}_1 \\ \mathsf{ct}_2 &= P_2(\boldsymbol{g}) + \underbrace{\boldsymbol{e}_2}_{\text{same support}} \end{aligned}$$

$$\begin{aligned} \mathsf{Eval}(\times, \mathsf{ct}_1, \mathsf{ct}_2) &= \mathsf{ct}_1 \odot \mathsf{ct}_2 \\ &= (P_1 \cdot P_2)(\boldsymbol{g}) + \underbrace{\boldsymbol{e}_1 \odot \mathsf{ct}_2 + \boldsymbol{e}_2 \odot \mathsf{ct}_1 - \boldsymbol{e}_1 \odot \boldsymbol{e}_2}_{\text{still in the same support}} \end{aligned}$$

Evaluation codes

Examples are:

- Reed-Muller [AAPS11]
- Reed-Solomon [BL11] (broken by [GOT12])

$\Longrightarrow$ highly **structured** codes

Outline

Alekhnovich (secret key version)

$$\mathrm{sk} = \boldsymbol{s}$$

$$\mathrm{Enc}(\boldsymbol{m}, \mathrm{sk}) = (\boldsymbol{G}, \boldsymbol{v} = \boldsymbol{s}\boldsymbol{G} + \boldsymbol{e} + \mathit{Encode}(\boldsymbol{m}))$$
$$\mathrm{Dec}(\mathrm{ct}, \mathrm{sk}) = \mathit{Decode}(\boldsymbol{v} - \boldsymbol{s}\boldsymbol{G})$$

Usually: $\mathit{Encode}(\boldsymbol{m}) = \boldsymbol{m}\mathcal{G}$, with $\mathcal{G}$ highly structured code

Ideal Alekhnovich (secret key version)

$$\text{sk} = \boldsymbol{s}$$

$$\text{Enc}(\boldsymbol{m}, \text{sk}) = (\boldsymbol{u}, \boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{s} + \boldsymbol{e} + \mathit{Encode}(\boldsymbol{m}))$$
$$\text{Dec}(\text{ct}, \text{sk}) = \mathit{Decode}(\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{s})$$

Usually: $\mathit{Encode}(\boldsymbol{m}) = \boldsymbol{m}\mathcal{G}$, with $\mathcal{G}$ highly structured code

Security reduction (single ciphertext)

$$\mathrm{ct} = (\boldsymbol{u}, \boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{s} + \boldsymbol{e} + Encode(\boldsymbol{m}))$$

$$\begin{pmatrix} \boldsymbol{v} \end{pmatrix} = \left(\begin{array}{c|c} \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{u}) \end{array} \right) \begin{pmatrix} \boldsymbol{e} \\ \hline \boldsymbol{s} \end{pmatrix} + Encode(\boldsymbol{m})$$

Additive homomorphic properties

$$\text{ct}_1 = (\boldsymbol{u}_1, \boldsymbol{v}_1 = \boldsymbol{u}_1 \cdot \boldsymbol{s} + \boldsymbol{e}_1 + Encode(\boldsymbol{m}_1))$$

$$\text{ct}_2 = (\boldsymbol{u}_2, \boldsymbol{v}_2 = \boldsymbol{u}_2 \cdot \boldsymbol{s} + \boldsymbol{e}_2 + Encode(\boldsymbol{m}_2))$$

$$\text{ct}_+ = (\boldsymbol{u}_1 + \boldsymbol{u}_2, (\boldsymbol{u}_1 + \boldsymbol{u}_2) \cdot \boldsymbol{s} + \boldsymbol{e}_1 + \boldsymbol{e}_2 + Encode(\boldsymbol{m}_1 + \boldsymbol{m}_2))$$

Security reduction (two ciphertexts)

$$\mathrm{ct}_1 = (\boldsymbol{u}_1, \boldsymbol{v}_1 = \boldsymbol{u}_1 \cdot \boldsymbol{s} + \boldsymbol{e}_1 + Encode(\boldsymbol{m}_1))$$
$$\mathrm{ct}_2 = (\boldsymbol{u}_2, \boldsymbol{v}_2 = \boldsymbol{u}_2 \cdot \boldsymbol{s} + \boldsymbol{e}_2 + Encode(\boldsymbol{m}_2))$$

$$\begin{pmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{pmatrix} = \left(\begin{array}{cc|c} \boldsymbol{I}_n & & \mathcal{IM}(\boldsymbol{u}_1) \\ & \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{u}_2) \end{array}\right) \begin{pmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \\ \boldsymbol{s} \end{pmatrix} \begin{array}{c} + \; Encode(\boldsymbol{m}_1) \\ + \; Encode(\boldsymbol{m}_2) \end{array}$$

Security reduction (two ciphertexts)

$$\mathrm{ct}_1 = \boldsymbol{m}_1 \boldsymbol{G} + \boldsymbol{e}_1$$
$$\mathrm{ct}_2 = \boldsymbol{m}_2 \boldsymbol{G} + \boldsymbol{e}_2$$

$$\begin{pmatrix} \mathrm{ct}_1 & \mathrm{ct}_2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{G} \end{pmatrix}^{\begin{pmatrix} \boldsymbol{m}_1 & \boldsymbol{m}_2 \end{pmatrix}} + \begin{pmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 \end{pmatrix}$$

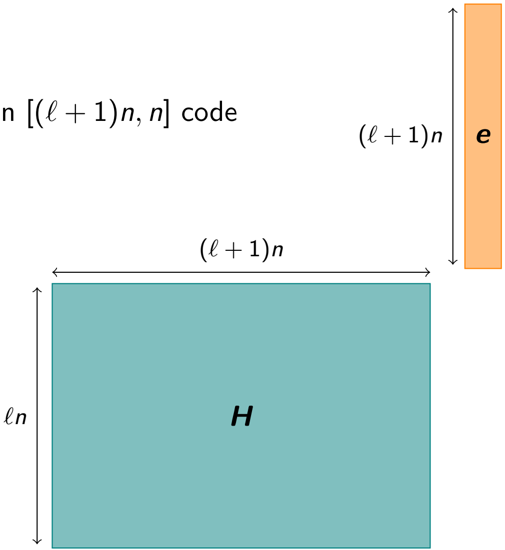
Security reduction (two ciphertexts)

$$\mathrm{ct}_1 = (\boldsymbol{u}_1, \boldsymbol{v}_1 = \boldsymbol{u}_1 \cdot \boldsymbol{s} + \boldsymbol{e}_1 + Encode(\boldsymbol{m}_1))$$
$$\mathrm{ct}_2 = (\boldsymbol{u}_2, \boldsymbol{v}_2 = \boldsymbol{u}_2 \cdot \boldsymbol{s} + \boldsymbol{e}_2 + Encode(\boldsymbol{m}_2))$$

$$\begin{pmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{pmatrix} = \left(\begin{array}{cc|c} \boldsymbol{I}_n & & \mathcal{IM}(\boldsymbol{u}_1) \\ & \boldsymbol{I}_n & \mathcal{IM}(\boldsymbol{u}_2) \end{array}\right) \begin{pmatrix} \boldsymbol{e}_1 \\ \boldsymbol{e}_2 \\ \boldsymbol{s} \end{pmatrix} \begin{array}{c} + \; Encode(\boldsymbol{m}_1) \\ + \; Encode(\boldsymbol{m}_2) \end{array}$$

Security reduction (ℓ ciphertexts)

Syndrome decoding problem in $[(\ell+1)n, n]$ code



Reminder: rank metric

	Hamming metric	Rank metric
Words	$(\mathbb{F}_q)^n$	$(\mathbb{F}_{q^m})^n$
Support	Indexes of non-zero coordinates	$\mathbb{F}_q$-subspace generated by coordinates
Small weight means...	Few non-zero coordinates	Each coordinate belongs to a small $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$

Ideal rank-metric Alekhnovich (secret key version)

$$\text{sk} = \boldsymbol{s} \text{ (of small support } E\text{)}$$

$$\text{Enc}(\boldsymbol{m}, \text{sk}) = (\boldsymbol{u}, \boldsymbol{v} = \boldsymbol{u} \cdot \overbrace{\boldsymbol{s} + \boldsymbol{e}}^{\text{same support } E} + Encode(\boldsymbol{m}))$$

$$\text{Dec}(\text{ct}, \text{sk}) = Decode(\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{s})$$

Usually: $Encode(\boldsymbol{m}) = \boldsymbol{m}\mathcal{G}$, with $\mathcal{G}$ highly structured code

Outline

Encoding the message perpendicular to the error

$$\mathsf{sk} = \boldsymbol{s}$$

$$\mathsf{Enc}(\boldsymbol{m}, \mathsf{sk}) = (\boldsymbol{u}, \boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{s} + \boldsymbol{e} + \mathit{Encode}(\boldsymbol{m}))$$
$$\mathsf{Dec}(\mathsf{ct}, \mathsf{sk}) = \mathit{Decode}(\boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{s})$$

Usually: $\mathit{Encode}(\boldsymbol{m}) = \boldsymbol{m}\mathcal{G}$, with $\mathcal{G}$ highly structured code
This work: $\mathit{Encode}(\boldsymbol{m}) = \mathsf{e}^{\perp} \cdot \boldsymbol{m}$ with $\mathsf{e}^{\perp} \in E^{\perp}$

Encryption / decryption algorithms

Rank Somewhat Homomorphic Encryption (RankSHE)

$$\mathsf{sk} = \boldsymbol{s} \in \mathbb{F}_{q^m}^n, \mathrm{Supp}(\boldsymbol{s}) = E, e^{\perp} \in E^{\perp}, \langle e^{\perp}, e^{\perp} \rangle = 1$$

$$\mathrm{Enc}(\boldsymbol{m} \in \mathbb{F}_q^n, \mathsf{sk}) = (\boldsymbol{u} \in \mathbb{F}_{q^m}^n, \boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{s} + \underbrace{\boldsymbol{e}}_{\boldsymbol{e} \in E} + e^{\perp} \cdot \boldsymbol{m})$$

$$\mathrm{Dec}((\boldsymbol{u}, \boldsymbol{v}), \mathsf{sk}) = \langle e^{\perp} \cdot \mathbf{1}, \boldsymbol{v} - \boldsymbol{u} \cdot \boldsymbol{s} \rangle$$

Proposition

The security of **RankSHE** *with ℓ independent ciphertexts is reduced to the $(\ell + 1)$-IRSD problem (decoding in an ideal $[(\ell + 1)n, n]_{q^m}$ code)*

Notation: scalar products

Let $(\gamma_1, \ldots, \gamma_m)$ be a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^m}$.

Definition (Scalar product in $\mathbb{F}_{q^m}$)

For $x = \sum_i x^{(i)}\gamma_i \in \mathbb{F}_{q^m}$, $y = \sum_j y^{(j)}\gamma_j \in \mathbb{F}_{q^m}$

$$\left\langle \sum_i x^{(i)}\gamma_i, \sum_j y^{(j)}\gamma_j \right\rangle := \sum_i x^{(i)} y^{(i)} \in \mathbb{F}_q$$

Definition (Scalar product in $\mathbb{F}_{q^m}^n$)

For $\boldsymbol{x} = (x_i)_i \in \mathbb{F}_{q^m}^n, \boldsymbol{y} = (y_i)_i \in \mathbb{F}_{q^m}^n$,

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle := (\langle x_i, y_i \rangle)_i \in \mathbb{F}_q^n$$

Notation: scalar products

Lemma

For $u \in \mathbb{F}_{q^m}$, $\boldsymbol{v} \in \mathbb{F}_{q^m}^n$ *and* $\boldsymbol{m} \in \mathbb{F}_q^n$,

$$\langle u\mathbf{1}, \boldsymbol{m} \cdot \boldsymbol{v} \rangle = \boldsymbol{m} \cdot \langle u\mathbf{1}, \boldsymbol{v} \rangle.$$

Addition

$$\mathrm{Enc}(\boldsymbol{m}_1, \mathrm{sk}) + \mathrm{Enc}(\boldsymbol{m}_2, \mathrm{sk}) = \mathrm{Enc}(\boldsymbol{m}_1 + \boldsymbol{m}_2, \mathrm{sk})$$

$$\boldsymbol{v}_1 = \boldsymbol{u}_1 \cdot \boldsymbol{s} + \boldsymbol{e}_1 + e^{\perp} \cdot \boldsymbol{m}_1$$

$$\boldsymbol{v}_2 = \boldsymbol{u}_2 \cdot \boldsymbol{s} + \boldsymbol{e}_2 + e^{\perp} \cdot \boldsymbol{m}_2$$

$$\boldsymbol{v}_1 + \boldsymbol{v}_2 = (\boldsymbol{u}_1 + \boldsymbol{u}_2) \cdot \boldsymbol{s} + \boldsymbol{e}_1 + \boldsymbol{e}_2 + e^{\perp} \cdot (\boldsymbol{m}_1 + \boldsymbol{m}_2)$$

Plaintext absorption

$$\boldsymbol{m}_1 \cdot \mathsf{Enc}(\boldsymbol{m}_2, \mathsf{sk}) = \mathsf{Enc}(\boldsymbol{m}_1 \cdot \boldsymbol{m}_2, \mathsf{sk})$$

$$\boldsymbol{v}_2 = \boldsymbol{u}_2 \cdot \boldsymbol{s} + \boldsymbol{e}_2 + e^{\perp} \cdot \boldsymbol{m}_2$$

$$\boldsymbol{m}_1 \cdot \boldsymbol{v}_2 = (\boldsymbol{m}_1 \cdot \boldsymbol{u}_2) \cdot \boldsymbol{s} + \boldsymbol{m}_1 \cdot \boldsymbol{e}_2 + e^{\perp} \cdot (\boldsymbol{m}_1 \cdot \boldsymbol{m}_2)$$

Multiplication

$$\mathrm{Eval}(\times, (\boldsymbol{u}_1, \boldsymbol{v}_1), (\boldsymbol{u}_2, \boldsymbol{v}_2)) = (\boldsymbol{u}_1 \cdot \boldsymbol{u}_2, \boldsymbol{u}_1 \cdot \boldsymbol{v}_2 + \boldsymbol{u}_2 \cdot \boldsymbol{v}_1, \boldsymbol{v}_1 \cdot \boldsymbol{v}_2)$$

$$\begin{aligned}
&\boldsymbol{u}_1 \cdot \boldsymbol{u}_2 \cdot \boldsymbol{s}^2 - (\boldsymbol{u}_1 \cdot \boldsymbol{v}_2 + \boldsymbol{u}_2 \cdot \boldsymbol{v}_1) \cdot \boldsymbol{s} + \boldsymbol{v}_1 \cdot \boldsymbol{v}_2 \\
&= (\boldsymbol{v}_1 - \boldsymbol{u}_1 \cdot \boldsymbol{s}) \cdot (\boldsymbol{v}_2 - \boldsymbol{u}_2 \cdot \boldsymbol{s}) \\
&= (\boldsymbol{e}_1 + e^{\perp} \cdot \boldsymbol{m}_1) \cdot (\boldsymbol{e}_2 + e^{\perp} \cdot \boldsymbol{m}_2) \\
&= \underbrace{\boldsymbol{e}_1 \cdot \boldsymbol{e}_2 + e^{\perp} \cdot (\boldsymbol{e}_1 \cdot \boldsymbol{m}_2 + \boldsymbol{e}_2 \cdot \boldsymbol{m}_1)}_{\boldsymbol{e}',\ \mathrm{Supp}(\boldsymbol{e}') \subset E^2 \oplus e^{\perp} E} + (e^{\perp})^2 \cdot \boldsymbol{m}_1 \cdot \boldsymbol{m}_2
\end{aligned}$$

$$\mathrm{Dec}\mathit{AfterMul}((\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), \mathrm{sk}) = \langle (e^{\perp})^2 \mathbf{1}, \boldsymbol{u} \cdot \boldsymbol{s}^2 - \boldsymbol{v} \cdot \boldsymbol{s} + \boldsymbol{w} \rangle$$

Summary

- Encryption scheme based on ideal random rank metric codes
- Unlimited additions
- Multiplication adds a component to the ciphertext and increases noise quadratically

Reducing ciphertext noise

$$(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) \text{ decryptable under } \mathsf{sk}_1$$

$$\Downarrow$$

$$(\boldsymbol{u}, \boldsymbol{v}) \text{ decryptable under } \mathsf{sk}_2$$

Bootstrapping

$$\mathsf{Eval}(\mathsf{Dec}\mathit{AfterMul}(\cdot,\cdot), \underbrace{\mathsf{ct}_1}_{\mathsf{Enc}((\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}),\mathsf{sk}_2)}, \underbrace{\mathsf{ct}_2}_{\mathsf{Enc}(\mathsf{sk}_1,\mathsf{sk}_2)}) \approx \mathsf{Enc}(\boldsymbol{m}, \mathsf{sk}_2)$$

$$\widehat{\mathsf{Eval}}(\mathsf{Dec}\mathit{AfterMul}(\cdot,\cdot), \underbrace{\mathsf{ct}_1}_{(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w})}, \underbrace{\mathsf{ct}_2}_{\mathsf{Enc}(\phi(\mathsf{sk}_1),\mathsf{sk}_2)}) \approx \mathsf{Enc}(\boldsymbol{m}, \mathsf{sk}_2)$$

Bootstrapping

$$\text{Dec}\mathit{AfterMul}((\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), \text{sk}_1) = \langle (e^{\perp})^2 \mathbf{1}, \boldsymbol{u} \cdot \boldsymbol{s}_1^2 - \boldsymbol{v} \cdot \boldsymbol{s}_1 + \boldsymbol{w} \rangle$$

$$\boldsymbol{u} = \sum_i \gamma_i \boldsymbol{u}^{(i)}$$

$$\boldsymbol{v} = \sum_i \gamma_i \boldsymbol{v}^{(i)}$$

$$\boldsymbol{w} = \sum_i \gamma_i \boldsymbol{w}^{(i)}$$

$$\text{Dec}\mathit{AfterMul}((\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), \text{sk}_1) = \sum_i \underbrace{\boldsymbol{u}^{(i)}}_{\in \mathbb{F}_q} \cdot \underbrace{\boldsymbol{a}^{(i)}}_{\in \mathbb{F}_q} - \boldsymbol{v}^{(i)} \cdot \boldsymbol{b}^{(i)} + \boldsymbol{w}^{(i)} \cdot \boldsymbol{c}^{(i)}$$

Bootstrapping

$$\begin{aligned}\langle (e^{\perp})^2\mathbf{1}, \boldsymbol{u}\cdot \boldsymbol{s}_1^2\rangle &= \sum_i \langle (e^{\perp})^2\mathbf{1}, \gamma_i \boldsymbol{u}^{(i)} \cdot \boldsymbol{s}_1^2\rangle \\ &= \sum_i \langle (e^{\perp})^2\mathbf{1}, \boldsymbol{u}^{(i)} \cdot \gamma_i \boldsymbol{s}_1^2\rangle \\ &= \sum_i \boldsymbol{u}^{(i)} \cdot \langle (e^{\perp})^2\mathbf{1}, \gamma_i \boldsymbol{s}_1^2\rangle \\ &= \sum_i \boldsymbol{u}^{(i)} \cdot \boldsymbol{a}^{(i)}\end{aligned}$$

Bootstrapping

$$\boldsymbol{m} = \mathrm{Dec}\mathit{AfterMul}((\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), \mathrm{sk}_1) = \sum_i \boldsymbol{u}^{(i)} \cdot \boldsymbol{a}^{(i)} - \boldsymbol{v}^{(i)} \cdot \boldsymbol{b}^{(i)} + \boldsymbol{w}^{(i)} \cdot \boldsymbol{c}^{(i)}$$

with

$$\boldsymbol{a}^{(i)} = \langle (e^{\perp})^2 \mathbf{1}, \gamma_i \boldsymbol{s}_1^2 \rangle$$
$$\boldsymbol{b}^{(i)} = \langle (e^{\perp})^2 \mathbf{1}, \gamma_i \boldsymbol{s}_1 \rangle$$
$$\boldsymbol{c}^{(i)} = \langle (e^{\perp})^2 \mathbf{1}, \gamma_i (1, 0, \ldots, 0) \rangle$$

Bootstrapping

$$\mathsf{ct}_{\boldsymbol{a}^{(i)}} = \mathsf{Enc}(\boldsymbol{a}^{(i)}, \mathsf{sk}_2) = \mathsf{Enc}(\langle (e^{\perp})^2 \mathbf{1}, \gamma_i \boldsymbol{s}_1^2 \rangle, \mathsf{sk}_2)$$

$$\mathsf{ct}_{\boldsymbol{b}^{(i)}} = \mathsf{Enc}(\boldsymbol{b}^{(i)}, \mathsf{sk}_2) = \mathsf{Enc}(\langle (e^{\perp})^2 \mathbf{1}, \gamma_i \boldsymbol{s}_1 \rangle, \mathsf{sk}_2)$$

$$\mathsf{ct}_{\boldsymbol{c}^{(i)}} = \mathsf{Enc}(\boldsymbol{c}^{(i)}, \mathsf{sk}_2) = \mathsf{Enc}(\langle (e^{\perp})^2 \mathbf{1}, \gamma_i (1, 0, \ldots, 0) \rangle, \mathsf{sk}_2)$$

$$\begin{aligned}
\widehat{\mathsf{Eval}}(\mathsf{Dec}\mathit{AfterMul}, (\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), \mathsf{ct}_{\boldsymbol{a}^{(i)}}, \mathsf{ct}_{\boldsymbol{b}^{(i)}}, \mathsf{ct}_{\boldsymbol{c}^{(i)}}) \\
:= \sum_i \boldsymbol{u}^{(i)} \cdot \mathsf{ct}_{\boldsymbol{a}^{(i)}} - \boldsymbol{v}^{(i)} \cdot \mathsf{ct}_{\boldsymbol{b}^{(i)}} + \boldsymbol{w}^{(i)} \cdot \mathsf{ct}_{\boldsymbol{c}^{(i)}} \\
= \mathsf{Enc}(\sum_i \boldsymbol{u}^{(i)} \cdot \boldsymbol{a}^{(i)} - \boldsymbol{v}^{(i)} \cdot \boldsymbol{b}^{(i)} + \boldsymbol{w}^{(i)} \cdot \boldsymbol{c}^{(i)}, \mathsf{sk}_2) \\
= \mathsf{Enc}(\boldsymbol{m}, \mathsf{sk}_2)
\end{aligned}$$

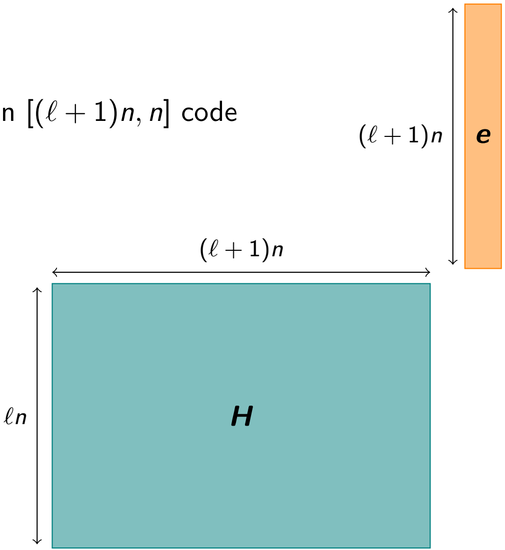
Bootstrapping

Our bootstrapping algorithm:

- Transforms a three-compenent ciphertext into a two-component ciphertext;
- reduces the noise from $\approx E_1^2$ to E_2;
- has no multiplicative cost;
- but... requires $3m$ independent ciphertexts under sk_2.

Security reduction (ℓ ciphertexts)

Syndrome decoding problem in $[(\ell+1)n, n]$ code



Security reduction (3m ciphertexts)

The attacker needs to solve the RSD problem in an ideal $[(3m+1)n, n]_{q^m}$ code.

There exists a polynomial attack [GRS13] in an $[n, k]_{q^m}$ code when

$$(k+1)(w+1) \leq n+1.$$

$\Longrightarrow$ maximal number of independent ciphertexts $\approx w$.

Reducing the number of bootstrapping ciphertexts

We pack several plaintexts into a single ciphertext:

$$\mathrm{Enc}((\boldsymbol{m}_1, \ldots, \boldsymbol{m}_p) \in (\mathbb{F}_q^n)^p, \mathrm{sk}) = (\boldsymbol{u} \in \mathbb{F}_{q^m}^n, \boldsymbol{v} = \boldsymbol{u} \cdot \boldsymbol{s} + \underbrace{\boldsymbol{e}}_{\|\boldsymbol{e}\| \leq w} + \sum_{i=1}^{p} \chi^i e^{\perp} \cdot \boldsymbol{m}_i)$$

with $\chi \in \mathbb{F}_{q^m}$ s.t. $\chi^p = 1$.

Maximal packing index $p = \frac{m}{w}$.

$\Longrightarrow$ reduces the number of bootstrapping plaintexts to $\frac{3m}{p} = 3w$.

Parameters

d	q	m	n	w	ℓ	Security	Key size	ct size	Add	Mul	Bootstrap
1	2	172	20	13	9	128	3.7 kB	**0.9 kB**	0.002 ms	**0.5 ms**	**2 ms**
2	2	367	183	7	5	128	17.0 kB	16.8 kB	0.04 ms	52 ms	374 ms
3	2	1296	314	6	4	128	210 kB	102 kB	0.3 ms	944 ms	11 s
4	2	3125	713	5	3	128	1.22 MB	557 kB	1 ms	14.3 s	239 s

Table: Example of paramaters for our SHE scheme, with associated sizes and execution timings. d is the number of possible multiplications. q, m and n are parameters of the rank linear code and w is the rank weight of the error. ℓ is the number of independant ciphertexts that can be published.

Comparison

Scheme	ct size	Bootstrap ct size	Mul time	Bootstrap time
TFHE [CGGI20]	2 kB	15.6 MB	0.03 ms	13 ms
[AAPS11]	18.5 kB	-	10 ms	-
This work	0.9 kB	35 kB	0.5 ms	2 ms

Table: Parameters are taken for 128-bit security, and for SHE schemes, with a single multiplication allowed.

Comparison

Scheme	ct size	Bootstrap ct size	Mul time	Bootstrap time
TFHE [CGGI20]	2 kB	15.6 MB	0.03 ms	13 ms
[AAPS11]	18.5 kB	-	10 ms	-
This work	0.9 kB	35 kB	0.5 ms	2 ms

Table: Parameters are taken for 128-bit security, and for SHE schemes, with a single multiplication allowed.

Thank you for your attention!

References I

Frederik Armknecht, Daniel Augot, Ludovic Perret, and Ahmad-Reza Sadeghi.
On constructing homomorphic encryption schemes from coding theory.
In IMA International Conference on Cryptography and Coding, pages 23–40.
Springer, 2011.

Dan Boneh, Eu-Jin Goh, and Kobbi Nissim.
Evaluating 2-DNF formulas on ciphertexts.
In Theory of Cryptography: Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005. Proceedings 2, pages 325–341.
Springer, 2005.

Andrej Bogdanov and Chin Ho Lee.
Homomorphic encryption from codes.
Cryptology ePrint Archive, Report 2011/622, 2011.
https://eprint.iacr.org/2011/622.

Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène.
Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds.
In Jung Hee Cheon and Tsuyoshi Takagi, editors, ASIACRYPT 2016, Part I, volume 10031 of LNCS, pages 3–33. Springer, Heidelberg, December 2016.

Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène.
TFHE: Fast fully homomorphic encryption over the torus.
Journal of Cryptology, 33(1):34–91, January 2020.

Craig Gentry.
Fully homomorphic encryption using ideal lattices.
In Proceedings of the forty-first annual ACM symposium on Theory of computing, pages 169–178, 2009.

References III

Philippe Gaborit, Adrien Hauteville, Duong Hieu Phan, and Jean-Pierre Tillich.
Identity-based encryption from codes with rank metric.
In Jonathan Katz and Hovav Shacham, editors, CRYPTO 2017, Part III, volume 10403 of LNCS, pages 194–224. Springer, Heidelberg, August 2017.

Valérie Gauthier, Ayoub Otmani, and Jean-Pierre Tillich.
A distinguisher-based attack of a homomorphic encryption scheme relying on reed-solomon codes.
Cryptology ePrint Archive, Report 2012/168, 2012.
https://eprint.iacr.org/2012/168.

Philippe Gaborit, Olivier Ruatta, and Julien Schrek.
On the complexity of the rank syndrome decoding problem.
CoRR, abs/1301.1026, 2013.