Fully Homomorphic Encryption for Matrix Arithmetic

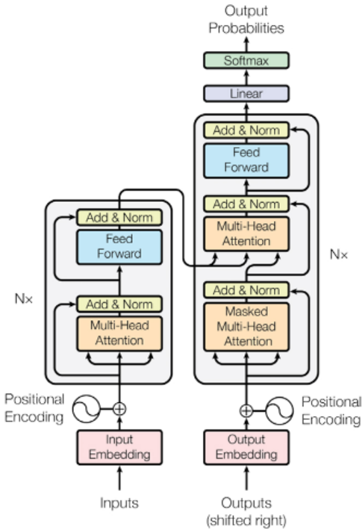
Craig Gentry[1]   **Yongwoo Lee**[2]

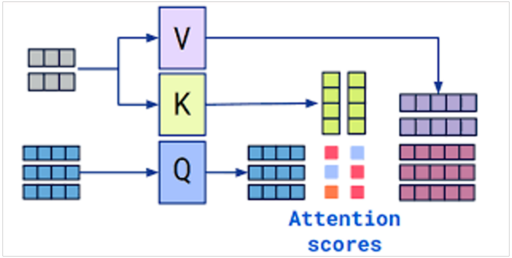
[1]Cornami

[2]DESILO Inc. / Inha University

Mar. 8, 2026

Motivation: Privacy-Preserving LLM





Goal

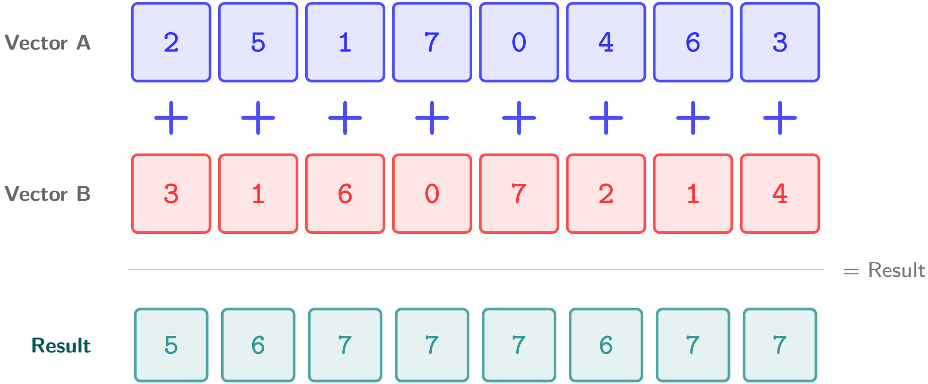
1. Ct-Ct Matrix multiplication as an **atomic** operation
   - Reduced to nonencrypted matrix-matrix multiplication
   - Single key switching, not relying on rotations or slot–coefficient transformation
   - Flexible matrix size
2. Addition and Hadamard products as-is
3. All possible rotations and transpositions

CKKS Binary Operations: Addition

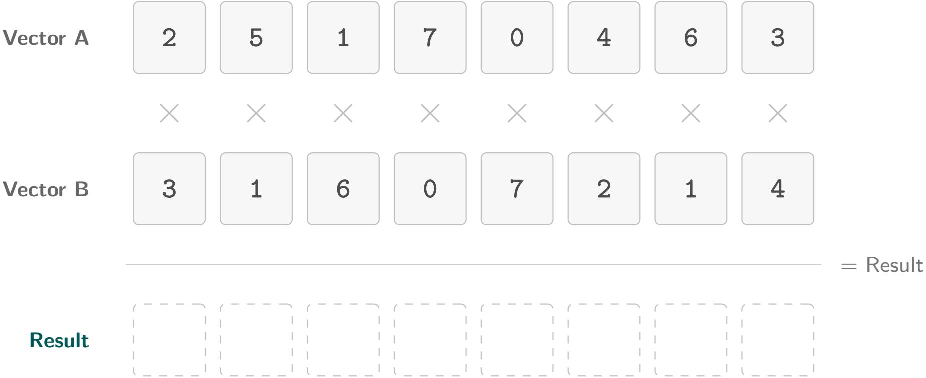
Vector A	2	5	1	7	0	4	6	3
	+	+	+	+	+	+	+	+
Vector B	3	1	6	0	7	2	1	4
Result								

= Result

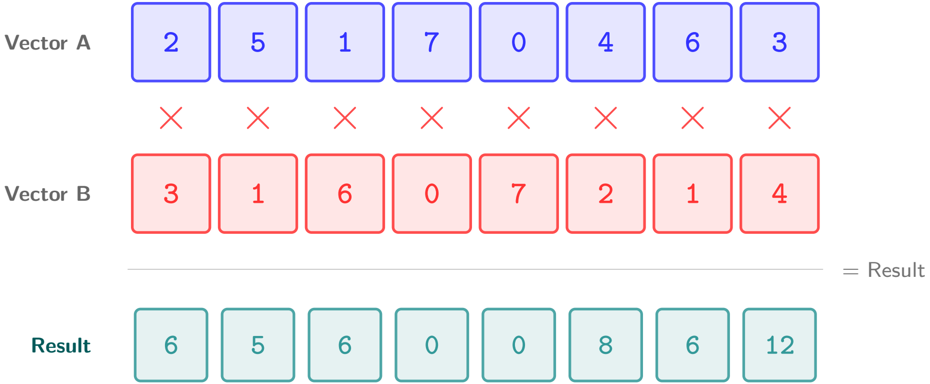
CKKS Binary Operations: Addition



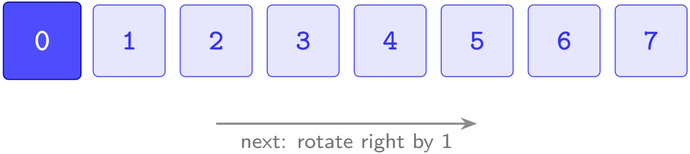
CKKS Binary Operations: Multiplication



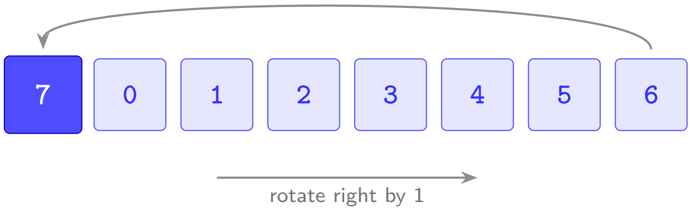
CKKS Binary Operations: Multiplication



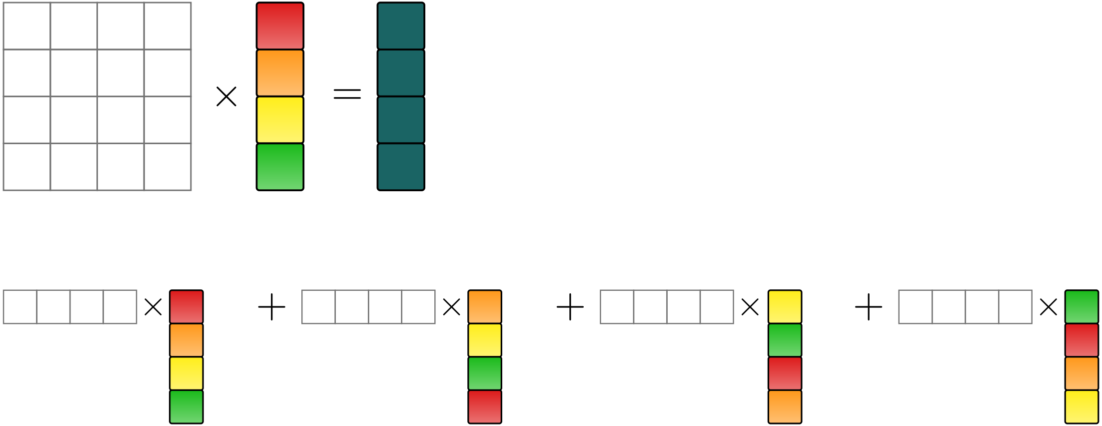
CKKS Rotations



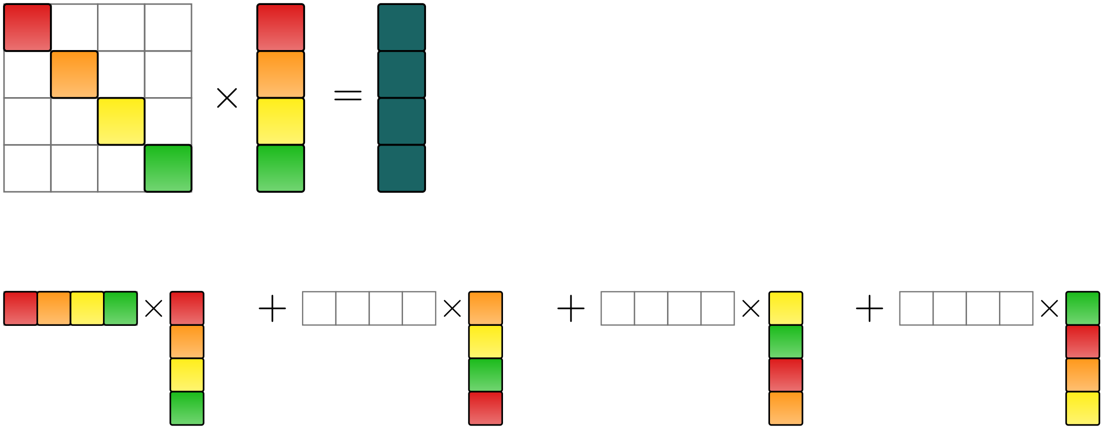
CKKS Rotations



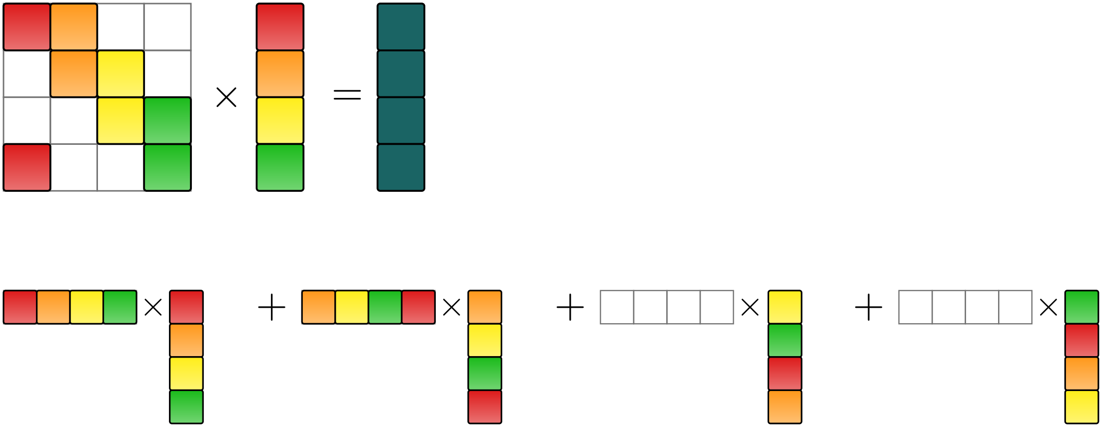
Matrix Multiplication with Rotation (Simplified)



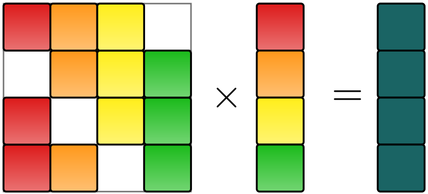
Matrix Multiplication with Rotation (Simplified)

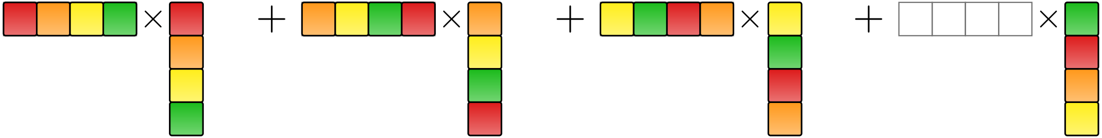


Matrix Multiplication with Rotation (Simplified)

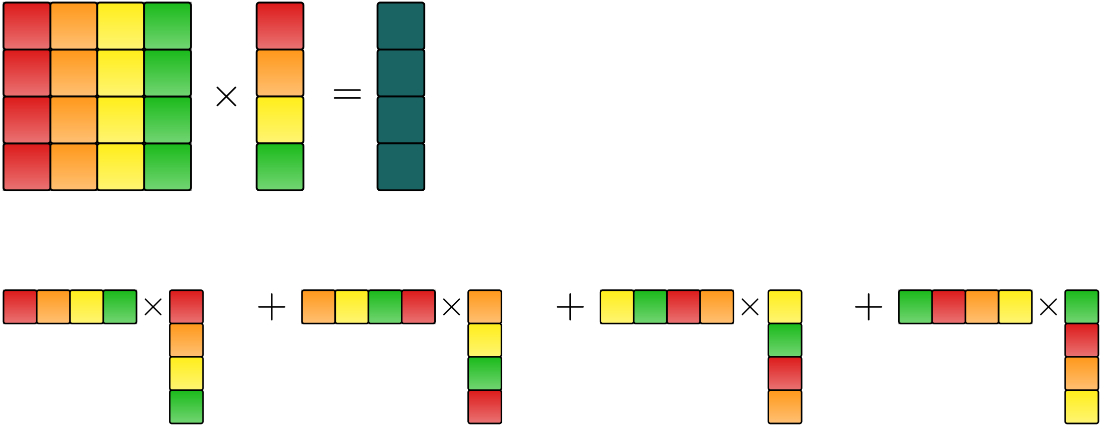


Matrix Multiplication with Rotation (Simplified)





Matrix Multiplication with Rotation (Simplified)



Important Related Works and Limitations

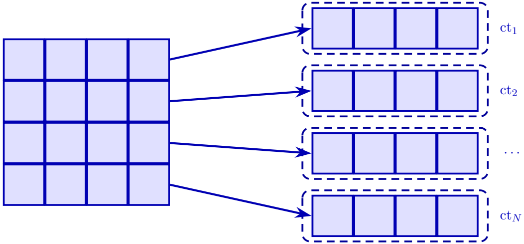
Pt-Ct matrix multiplication to Pt-Pt matrix multiplication:

- Liu, J., Zhang, L.F., *Privacy-preserving and publicly verifiable matrix multiplication,* IEEE Transactions on Services Computing, IEEE Tran. on Serv. Comput. 2023.
- Bae, Y., Cheon, J.H., Hanrot, G., Park, J.H., Stehlé, D., *Plaintext-ciphertext matrix multiplication and FHE bootstrapping,* CRYPTO'24.

Ct-Ct matrix multiplication to Pt-Pt matrix multiplication:

- Park, J.H., *Ciphertext-ciphertext matrix multiplication: Fast for large matrices,* Eurocrypt'25.
- Cheon, J.H., Kang, M., Lee, J., *Fast batch matrix multiplication in ciphertexts,* ia.cr/2025/1957 (parallel).

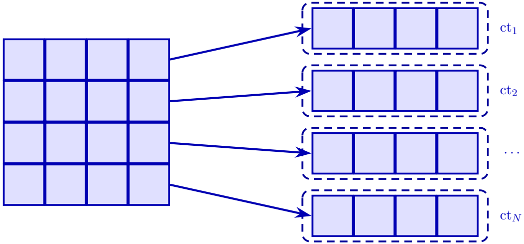
Important Related Works and Limitations (Contd.)



Limitations

[1] ia.cr/2025/1957 resolves this problem

Important Related Works and Limitations (Contd.)

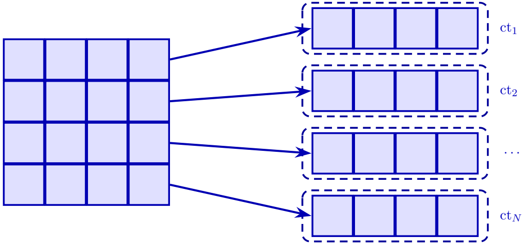


Limitations

- Matrix dimension $\approx$ coefficient count; ring switching required[1]

[1] ia.cr/2025/1957 resolves this problem

Important Related Works and Limitations (Contd.)



Limitations

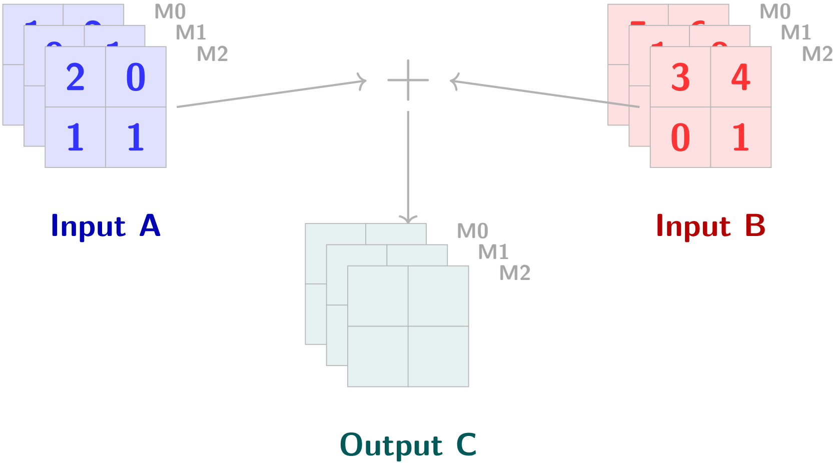
- Matrix dimension $\approx$ coefficient count; ring switching required[1]
- Slot–coefficient transformation required

[1] ia.cr/2025/1957 resolves this problem

Proposed Binary Operations: Addition

Batch-wise: for each $m \in \{0, 1, 2\}$, $C_m = A_m + B_m$

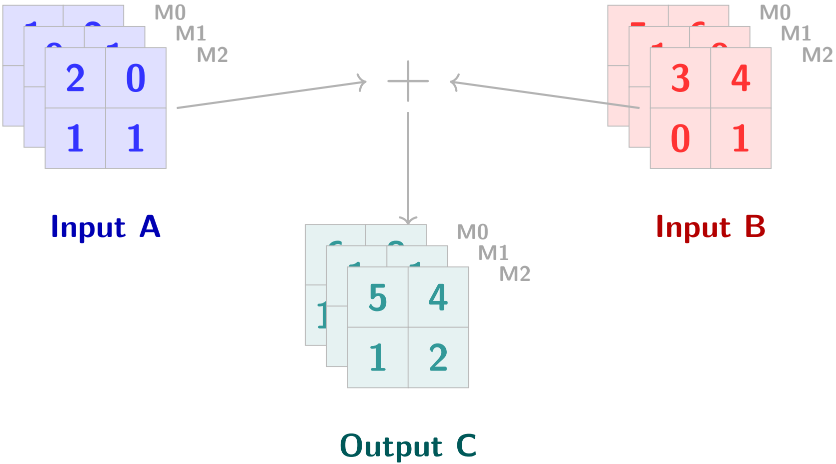
- Two plain additions in $\mathbb{Z}_q$
- No key switching



Proposed Binary Operations: Addition

Batch-wise: for each $m \in \{0, 1, 2\}$, $C_m = A_m + B_m$

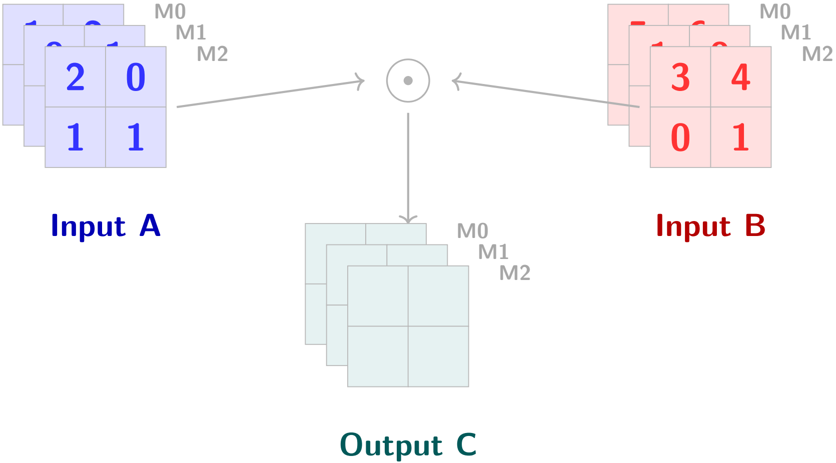
- Two plain additions in $\mathbb{Z}_q$
- No key switching



Proposed Binary Operations: Hadamard Multiplication

Batch-wise Hadamard: for each $m \in \{0, 1, 2\}$, $C_m = A_m \odot B_m$

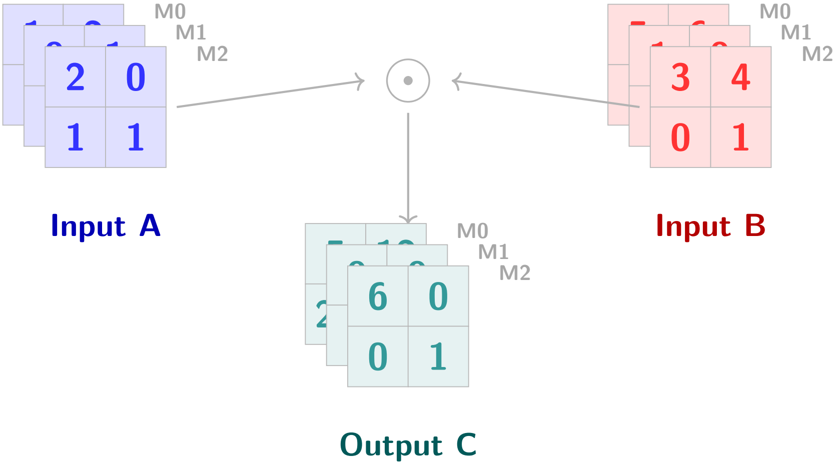
- Four plain ring multiplications in $\mathbb{Z}_q$
- Single key switching.



Proposed Binary Operations: Hadamard Multiplication

Batch-wise Hadamard: for each $m \in \{0, 1, 2\}$, $C_m = A_m \odot B_m$

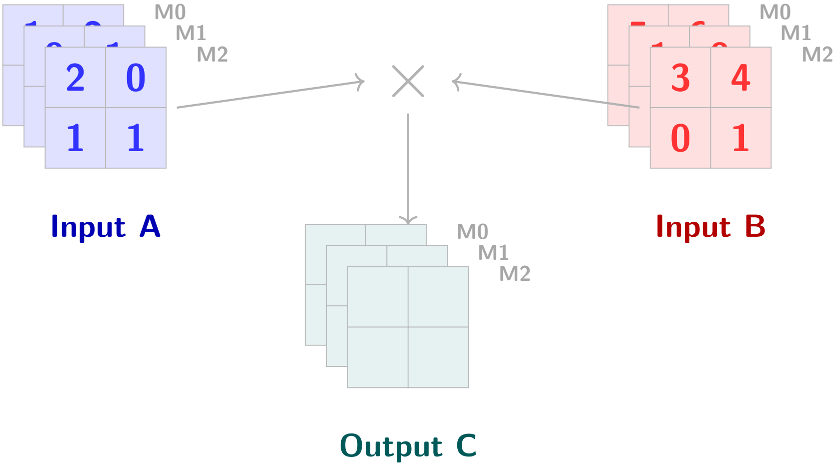
- Four plain ring multiplications in $\mathbb{Z}_q$
- Single key switching.



Proposed Binary Operations: Matrix Multiplication

Batch-wise matrix product: for each $m \in \{0,1,2\}$, $C_m = A_m B_m$

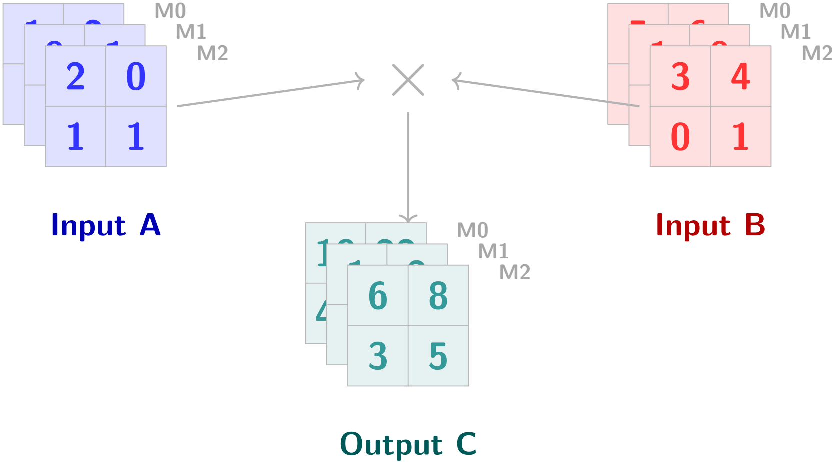
- Four plain MMs in $\mathbb{Z}_q[i]$
- Single (up to x4 heavy) key switching, no rotations or slot–coefficient transformation



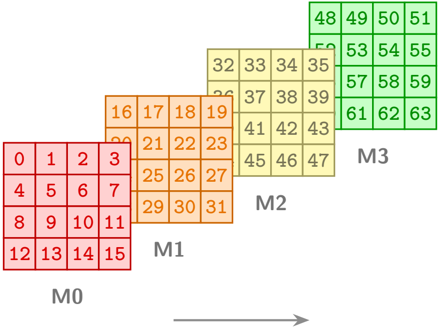
Proposed Binary Operations: Matrix Multiplication

Batch-wise matrix product: for each $m \in \{0, 1, 2\}$, $C_m = A_m B_m$

- Four plain MMs in $\mathbb{Z}_q[i]$
- Single (up to x4 heavy) key switching, no rotations or slot–coefficient transformation

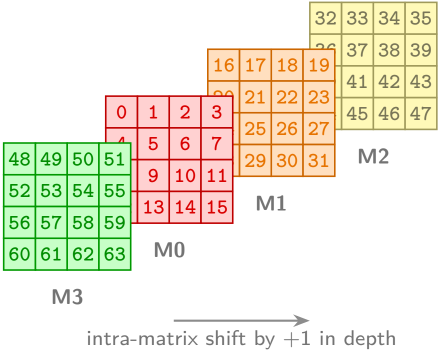


Inter-Matrix Rotations

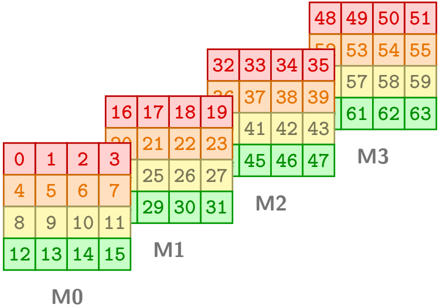


next: intra-matrix shift by +1 in depth

Inter-Matrix Rotations

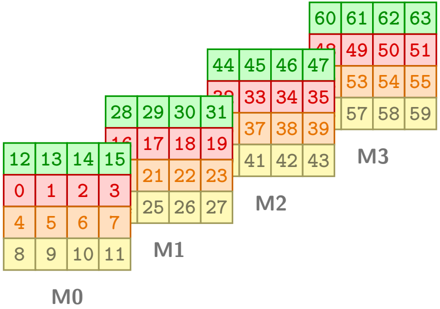


Intra-Matrix Rotations: Row



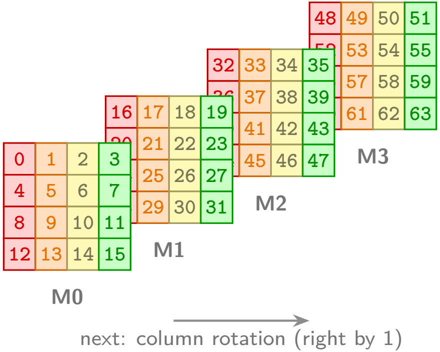
next: row rotation (down by 1)

Intra-Matrix Rotations: Row

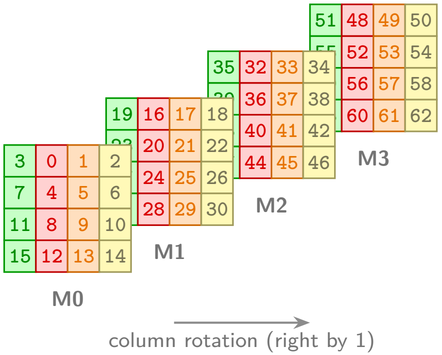


row rotation (down by 1)

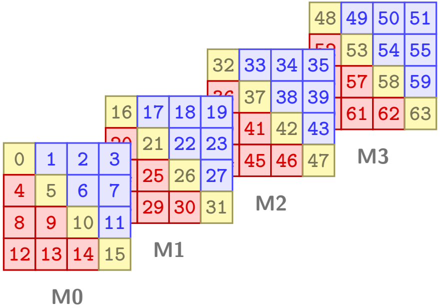
Intra-Matrix Rotations: Column



Intra-Matrix Rotations: Column

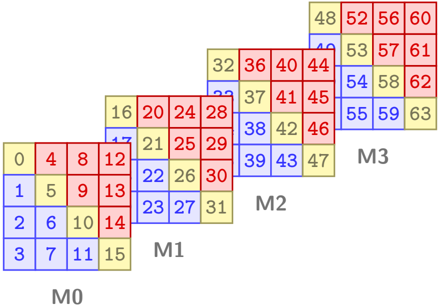


Matrix Transposition



next: transposition $(A \to A^{\mathsf{T}})$

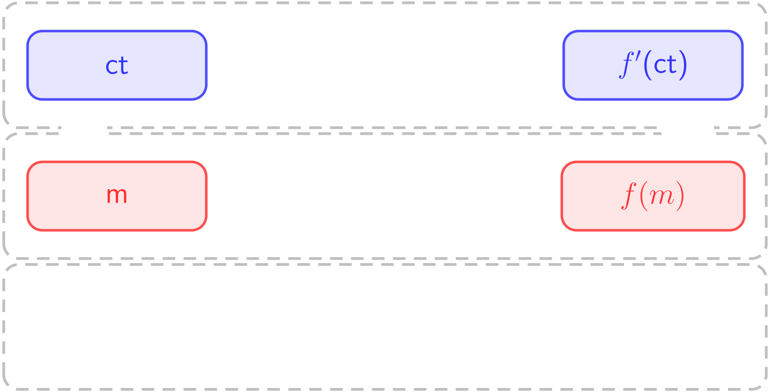
Matrix Transposition



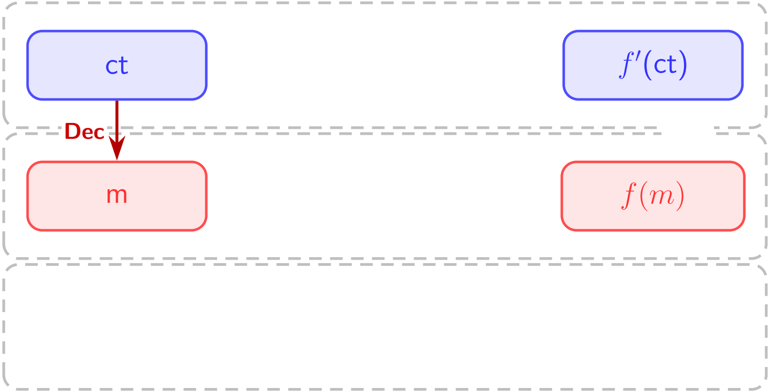
transposition $(A \to A^{\mathsf{T}})$

Technical Details

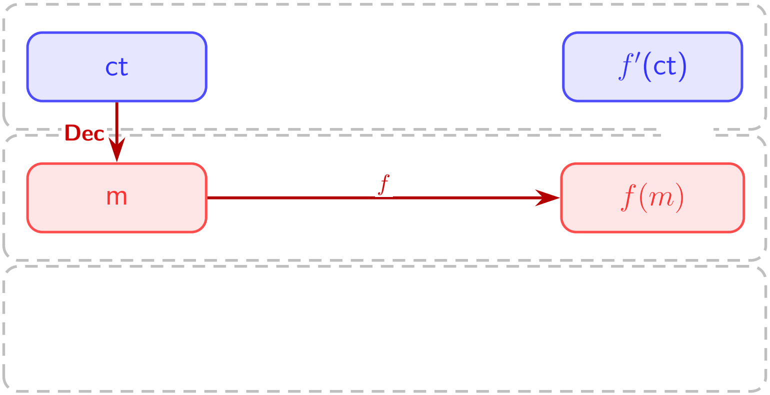
Homomorphic Encryption



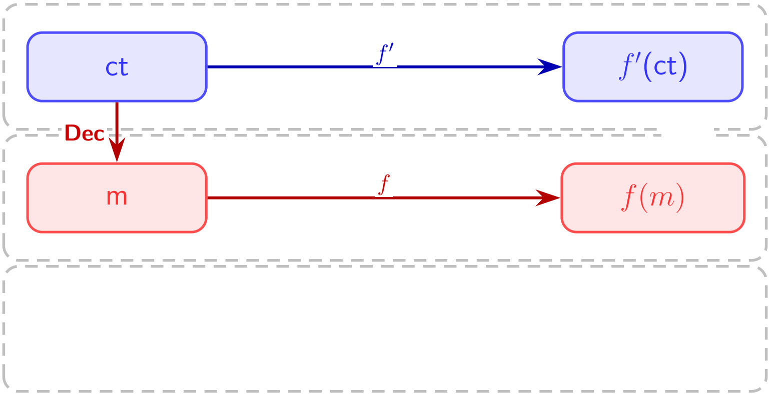
Homomorphic Encryption



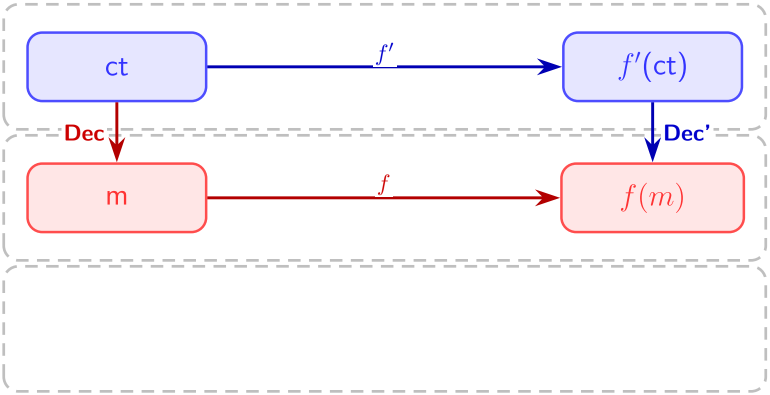
Homomorphic Encryption



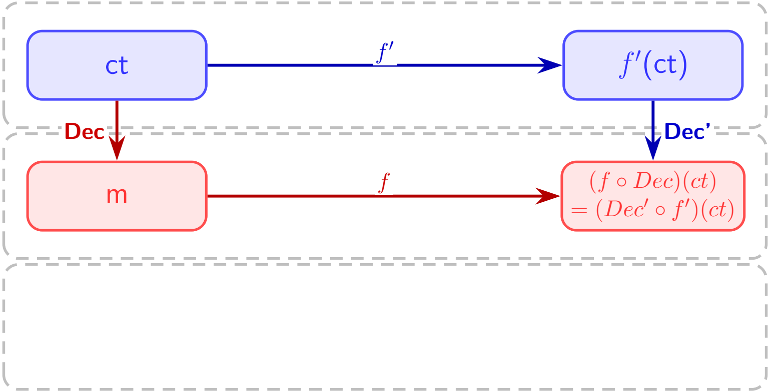
Homomorphic Encryption



Homomorphic Encryption

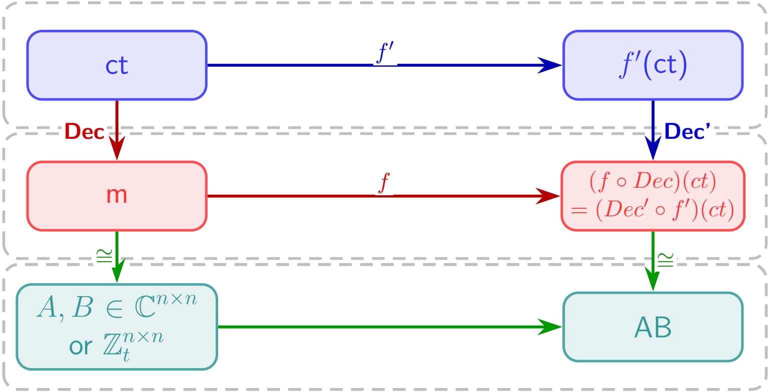


Homomorphic Encryption



Decryption and f commute

Homomorphic Encryption



Decryption and f commute and f should be meaningful

Basic Characteristics of Trace

$$\mathrm{Tr} : S \to R, \quad s \mapsto \sum_{\sigma} \sigma(s)$$

S is some extension of R, and σ are automorphisms fixing R.

Basic Characteristics of Trace

$$\mathrm{Tr} : S \to R, \quad s \mapsto \sum_{\sigma} \sigma(s)$$

S is some extension of R, and σ are automorphisms fixing R. Equivalently, $\mathrm{Tr}(s) = \sum_{\ell} s(Z^{\ell})$.

Linearity

- $\mathrm{Tr}(s_1 + s_2) = \mathrm{Tr}(s_1) + \mathrm{Tr}(s_2)$
- $\mathrm{Tr}(r \cdot s) = r \cdot \mathrm{Tr}(s)$ for $r \in R, s \in S$

Matrix Multiplication and Trace

Matrix Multiplication:

$$C_{jk} = \sum_{\ell} A_{j\ell} B_{\ell k}$$

Trace:

$$c(X, Y) = \sum_{\ell} a(X, Z^{\ell}) b(Y, Z^{\ell})$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$

$$c(\zeta_j, \zeta_k) = C_{jk}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$

$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$c(\zeta_j, \zeta_k) = C_{jk}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$

$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$\begin{aligned} c(\zeta_j, \zeta_k) &= C_{jk} \\ &= \mathrm{Tr}(a(\zeta_j, Z) b(\zeta_k, Z)) \end{aligned}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$

$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$\begin{aligned} c(\zeta_j, \zeta_k) &= C_{jk} \\ &= \mathrm{Tr}(a(\zeta_j, Z) b(\zeta_k, Z)) \\ &= \sum_\ell a(\zeta_j, Z^\ell) b(\zeta_k, Z^\ell) \end{aligned}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$
$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$\begin{aligned} c(\zeta_j, \zeta_k) &= C_{jk} \\ &= \mathrm{Tr}(a(\zeta_j, Z) b(\zeta_k, Z)) \\ &= \sum_\ell a(\zeta_j, Z^\ell) b(\zeta_k, Z^\ell) \\ &= \sum_\ell a(\zeta_j, \zeta_\ell) b(\zeta_k, \zeta_\ell) = \sum_\ell A_{j\ell} B_{k\ell} \end{aligned}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z) b(Y, Z)$$
$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$\begin{aligned} c(\zeta_j, \zeta_k) &= C_{jk} \\ &= \mathrm{Tr}(a(\zeta_j, Z) b(\zeta_k, Z)) \\ &= \sum_\ell a(\zeta_j, Z^\ell) b(\zeta_k, Z^\ell) \\ &= \sum_\ell a(\zeta_j, \zeta_\ell) b(\zeta_k, \zeta_\ell) = \sum_\ell A_{j\ell} B_{k\ell} \\ C_{jk} &= [AB^T]_{jk} \end{aligned}$$

Our Decoding

Decoding:

$$a(\zeta_j, \zeta_k) = A_{jk},$$

$\zeta_j \in \mathbb{C}$ or $\mathbb{Z}_t$ are roots of unity.

Decoding of Traced Value:

$$c(X, Y) = \sum_Z a(X, Z)b(Y, Z)$$
$$c(\zeta_j, \zeta_k) = C_{jk}$$

$$\begin{aligned}
c(\zeta_j, \zeta_k) &= C_{jk} \\
&= \mathrm{Tr}(a(\zeta_j, Z)b(\zeta_k, Z)) \\
&= \sum_\ell a(\zeta_j, Z^\ell)b(\zeta_k, Z^\ell) \\
&= \sum_\ell a(\zeta_j, \zeta_\ell)b(\zeta_k, \zeta_\ell) = \sum_\ell A_{j\ell}B_{k\ell} \\
C_{jk} &= [AB^T]_{jk}
\end{aligned}$$

We have a meaningful f that commutes with ring addition and multiplication.

Encrypted Matrix Multiplication

Two ciphertexts under secret key $s(X)$ s is Y-free.

- $(b, a) : m(X, Y) = b(X, Y) + a(X, Y) \cdot s(X)$
- $(\beta, \alpha) : \mu(X, Y) = \beta(X, Y) + \alpha(X, Y) \cdot s(X)$

With permutations $(X, Y) \mapsto (X, Z)$ and $(X, Y) \mapsto (Y, Z)$:

- $m(X, Z) = b(X, Z) + a(X, Z) \cdot s(X)$
- $\mu(Y, Z) = \beta(Y, Z) + \alpha(Y, Z) \cdot s(Y)$

Encrypted Matrix Multiplication

Two ciphertexts under secret key $s(X)$ s is Y-free.

- $(b, a) : m(X, Y) = b(X, Y) + a(X, Y) \cdot s(X)$
- $(\beta, \alpha) : \mu(X, Y) = \beta(X, Y) + \alpha(X, Y) \cdot s(X)$

With permutations $(X, Y) \mapsto (X, Z)$ and $(X, Y) \mapsto (Y, Z)$: s is Z-free.

- $m(X, Z) = b(X, Z) + a(X, Z) \cdot s(X)$
- $\mu(Y, Z) = \beta(Y, Z) + \alpha(Y, Z) \cdot s(Y)$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$\mathrm{Tr}(m(X, Z) \cdot \mu(Y, Z))$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y)))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z))\\
&=\mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y)))\\
&=\mathrm{Tr}(b(X,Z)\beta(Y,Z))+\mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z) + a(X,Z)s(X))\cdot(\beta(Y,Z) + \alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + \mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z)) \\
&\quad + \mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + \mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z)) \\
&\quad + \mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y)) + \mathrm{Tr}(a(X,Z)s(X)\alpha(Y,Z)s(Y))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z))\\
&=\mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y)))\\
&=\mathrm{Tr}(b(X,Z)\beta(Y,Z))\ +\mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z))\\
&\quad+\mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y))\ +\mathrm{Tr}(a(X,Z)s(X)\alpha(Y,Z)s(Y))\\
&=\mathrm{Tr}(b(X,Z)\beta(Y,Z))+s(X)\mathrm{Tr}(a(X,Z)\beta(Y,Z))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + \mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z)) \\
&\quad + \mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y)) + \mathrm{Tr}(a(X,Z)s(X)\alpha(Y,Z)s(Y)) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + s(X)\mathrm{Tr}(a(X,Z)\beta(Y,Z)) \\
&\quad + s(Y)\mathrm{Tr}(b(X,Z)\alpha(Y,Z)) + s(X)s(Y)\mathrm{Tr}(a(X,Z)\alpha(Y,Z))
\end{aligned}
$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$
\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + \mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z)) \\
&\quad + \mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y)) + \mathrm{Tr}(a(X,Z)s(X)\alpha(Y,Z)s(Y)) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + s(X)\mathrm{Tr}(a(X,Z)\beta(Y,Z)) \\
&\quad + s(Y)\mathrm{Tr}(b(X,Z)\alpha(Y,Z)) + s(X)s(Y)\mathrm{Tr}(a(X,Z)\alpha(Y,Z))
\end{aligned}
$$

The ciphertext for the matrix multiplication is a 4-tuple:

$$\big(\mathrm{Tr}(b(X,Z)\beta(Y,Z))\,,\mathrm{Tr}(b(X,Z)\alpha(Y,Z))\,,\mathrm{Tr}(a(X,Z)\beta(Y,Z))\,,\mathrm{Tr}(a(X,Z)\alpha(Y,Z))\,\big),$$

Encrypted Matrix Multiplication (Contd.)

Trace of their product:

$$\begin{aligned}
&\mathrm{Tr}(m(X,Z)\cdot\mu(Y,Z)) \\
&= \mathrm{Tr}((b(X,Z)+a(X,Z)s(X))\cdot(\beta(Y,Z)+\alpha(Y,Z)s(Y))) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + \mathrm{Tr}(a(X,Z)s(X)\beta(Y,Z)) \\
&\quad + \mathrm{Tr}(b(X,Z)\alpha(Y,Z)s(Y)) + \mathrm{Tr}(a(X,Z)s(X)\alpha(Y,Z)s(Y)) \\
&= \mathrm{Tr}(b(X,Z)\beta(Y,Z)) + s(X)\mathrm{Tr}(a(X,Z)\beta(Y,Z)) \\
&\quad + s(Y)\mathrm{Tr}(b(X,Z)\alpha(Y,Z)) + s(X)s(Y)\mathrm{Tr}(a(X,Z)\alpha(Y,Z))
\end{aligned}$$

The ciphertext for the matrix multiplication is a 4-tuple:

$$\left(\mathrm{Tr}(b(X,Z)\beta(Y,Z)),\mathrm{Tr}(b(X,Z)\alpha(Y,Z)),\mathrm{Tr}(a(X,Z)\beta(Y,Z)),\mathrm{Tr}(a(X,Z)\alpha(Y,Z))\right),$$

whose decryption key is

$$(1, s(X), s(Y), s(X)s(Y)).$$

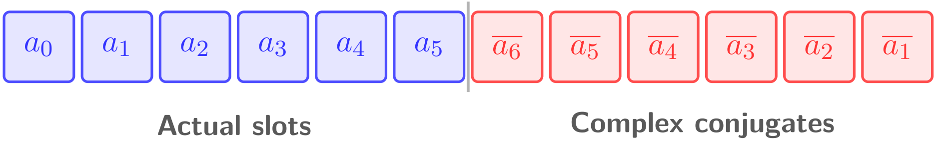
Tweak 1. Gaussian Integer: Removing Conjugate Symmetry

Simply extending to Y does not work:

Tweak 1. Gaussian Integer: Removing Conjugate Symmetry

Simply extending to Y does not work:
Conjugate symmetry is required in CKKS encoding.

$$\mathbb{Z}[X]/\langle X^n + 1 \rangle$$

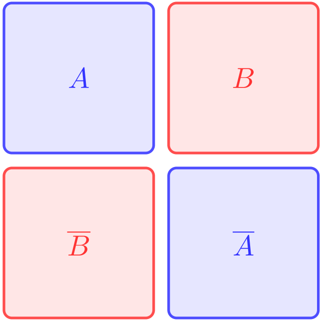


Tweak 1. Gaussian Integer: Removing Conjugate Symmetry

Simply extending to Y does not work:

- Fixed conjugate pairs
- Summing over all roots of unity mixes components

$$\mathbb{Z}[X, Y]/\langle X^n + 1, Y^n + 1\rangle$$



Tweak 1. Gaussian Integer: Removing Conjugate Symmetry

Our idea: remove conjugate symmetry using **Gaussian integers** $\mathbb{Z}[i]$.

- Simple isomorphism $\mathbb{Z}[i][X]/\langle X^{n/2} - i\rangle \cong \mathbb{Z}[X]/\langle X^n + 1\rangle$
- $\zeta_j = \zeta^{5^j}$ are roots of $X^{n/2} - i$, so we can decode $A_{jk} = a(\zeta_j, \zeta_k)$.

$$\mathbb{Z}[i][X,Y]/\langle X^{n/2} - i, Y^{n/2} - i\rangle$$

$$A \in \mathbb{C}^{n/2 \times n/2}$$

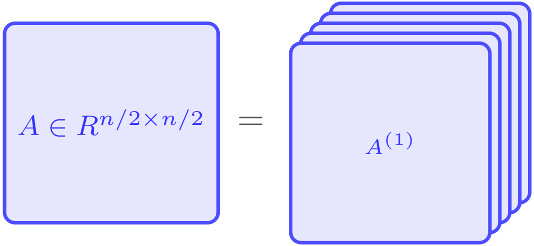
or $\in (\mathbb{Z}_t^{n/2 \times n/2})^2$

Tweak 2. Multi-Variate Ring: Flexible Matrix Dimensions

Remaining challenge: matrix dimensions must be $n/2$.

Our idea:

- Multivariate ring
  - $R[X,Y]/\langle X^{n/2}-i, Y^{n/2}-i\rangle \cong \mathbb{Z}[i][X,Y,W]/\langle X^{n/2}-i, Y^{n/2}-i, \Phi_p(W)\rangle$
- Matrix elements in $R = \mathbb{Z}[i][W]/\langle \Phi_p(W)\rangle$.
- Batching $\varphi(p)$ matrices in the W axis, much smaller n.



Security: Basic RLWE with Φ_{2np}

- Each ciphertext is based on ring $\mathbb{Z}[i][X,W]/\langle X^n - i, \Phi_p(W)\rangle$.
- For odd p, we have

$$\mathbb{Z}[i][X,W]/\langle X^n - i, \Phi_p(W)\rangle \cong \mathbb{Z}[X]/\langle \Phi_{2np}(X)\rangle.$$

- We have the bijection:

$$\mathsf{ct} = \sum_i Y^i \mathsf{ct}_i,$$

ct in X, Y, W-ring and ct_i in X, W-ring.

Implementation Results

```
$ pip install desilofhe
```

```
from desilofhe import GLEngine

engine = GLEngine()

secret_key = engine.create_secret_key()
matrix_multiplication_key =
engine.create_matrix_multiplication_key(secret_key)
ciphertext1 = engine.encrypt(np.ones(engine.shape), secret_key)
ciphertext2 = engine.encrypt(np.ones(engine.shape), secret_key)

multiplied = engine.matrix_multiply(
    ciphertext1, ciphertext2, matrix_multiplication_key
)
```

Implementation Results: Comparison of Operations

Table 1: Runtime for 16 batched 128×128 matrices with three levels ($n = 256, p = 17$) on an Intel i9-11900K (single-threaded)

Operations		Runtime (s)			
Category	Operation	Total	Key switching	Others	Mat. Mult. in $\mathbb{Z}_q$
Matrix Mult.	Ct-Ct matrix mult.	11.582	3.343	8.240	7.407
	(with FLINT)	**(7.237)**	(3.317)	(3.920)	(3.042)
	Pt-Ct Matrix Mult.	4.220	-	4.220	3.705
	(with FLINT)	(2.064)	-	(2.064)	(1.524)
Add/Mult.	Addition	0.018	-	0.018	-
	Hadamard Mult.	**2.779**	1.169	1.610	-
Transposition	Conjugation	1.216	1.178	0.038	-
	Transpose	1.712	1.674	0.038	-
	Conj. Trans.	1.758	1.682	0.039	-
Rotation	Row Rot.	1.059	1.082	0.023	-
	Column Rot.	0.005	-	0.005	-
	Inter-Matrix Rot.	1.115	1.093	0.022	-

Implementation Results: Comparison of Key Switching Cost

Table 2: Comparison with OpenFHE baseline (power-of-two ring)

	runtime (s)	slot count	amortized (μs)
Key switching single ct for R_q	0.004697	256×16	1.147
KS for Hadamard mult.	1.169	$256 \times 256 \times 16$	1.115
KS for ct-ct mult.	3.317	$256 \times 256 \times 16$	3.163
OpenFHE rotate (same coeff count)	0.003367	4096	0.822

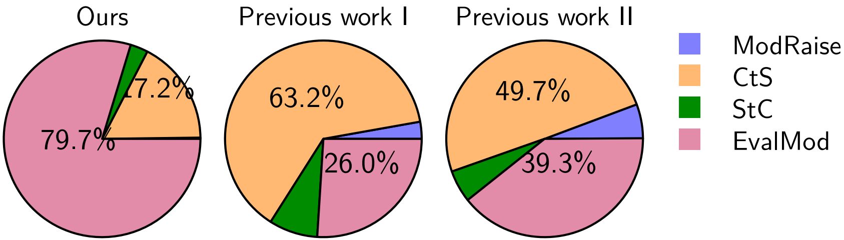
Implementation Results: GPU Acceleration

Table 3: Runtime for various parameter sets (ms): CPU (w/ FLINT) vs NVIDIA RTX 4090

n	p	level	Env.	Ct-Ct Matrix Mult		Pt-Ct Matrix Mult	Hadamard
				Mat Mult.	Total		
2^4	257	3	CPU	30.9 (34.2)	247 (245)	52 (50)	144
2^4	257	3	CUDA	0.27	35.9	-	-
2^8	17	3	CPU	8,625 (3,295)	11,989 (6,661)	4,779 (2,080)	2,406
2^8	17	3	CUDA	24.8	496	-	-
2^{10}	5	3	CPU	160,132 (46,737)	176,286 (63,066)	81,943 (24,961)	11,413
2^{10}	5	3	CUDA	451	2,822	-	-
2^5	257	9	CPU	738 (485)	4,440 (4,191)	796 (654)	2,285
2^5	257	9	CUDA	2.66	263	-	-
2^9	17	9	CPU	182,062 (60,832)	245,644 (124,210)	96,840 (35,778)	38,121
2^6	257	18	CPU	12,642 (6,868)	51,129 (45,341)	10,278 (7,241)	23,296

Application: Bootstrapping

- Our matrix multiplication can be generalized to non-power-of-two matrix dimensions.
- Native matrix multiplication enables CtS/StC with $O(1)$ key switching.
- This reduces their cost from $55–70\%$ to 20% of the total runtime.



Concluding Remarks

Engineer's Perspective:

- A new FHE scheme for matrix multiplication is here.
- Let's build privacy-preserving LLM with it.

Cryptographer's Perspective:

- There is a richer set of homomorphic operations beyond $+, \times$.
- We may be able to move more structure "inside-out".

Limitation

- Due to batching, the unit size increases by $\times n$.
- NTT for Φ_p needs more engineering.

Thank you!

ia.cr/2025/1935

DESILOFHE Library

Quick start with Colab

CORNAMI® Intelligent Computing

DESILO