Homomorphic Encryption for Arithmetic of Approximate Numbers

Jung Hee Cheon*, Andrey Kim*, Miran Kim†, Yongsoo Song*

*Seoul National University

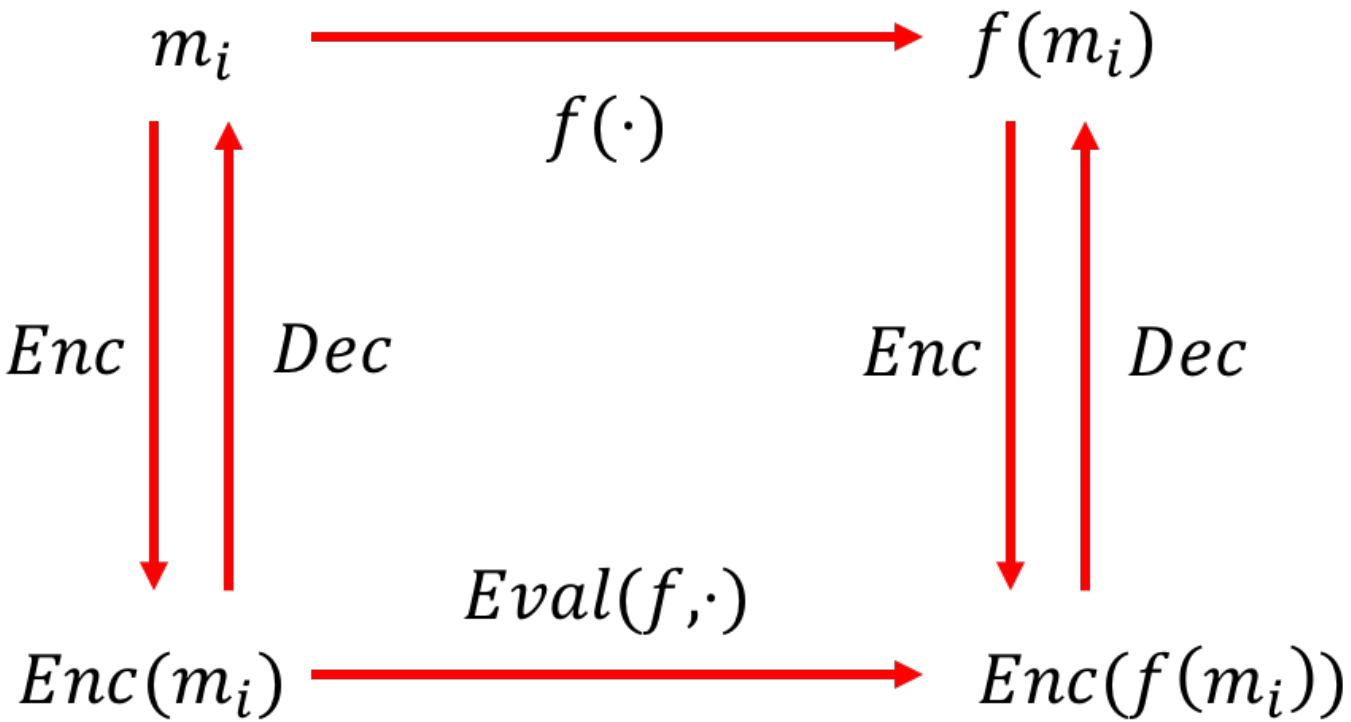
†University of California - SD

2017. 12. 04.

ASIACRYPT 2017

Homomorphic Encryption

- $ct_1 \leftarrow Enc(m_1), \ldots, ct_k \leftarrow Enc(m_k)$.
- $ct^* \leftarrow Eval(f, ct_1, \ldots, ct_k) \implies Dec(ct^*) = f(m_1, \ldots, m_k)$.



Can we compute the significant digits of four 32-bit integers in a second?

How long does it take to multiply 1024 floating-point numbers?

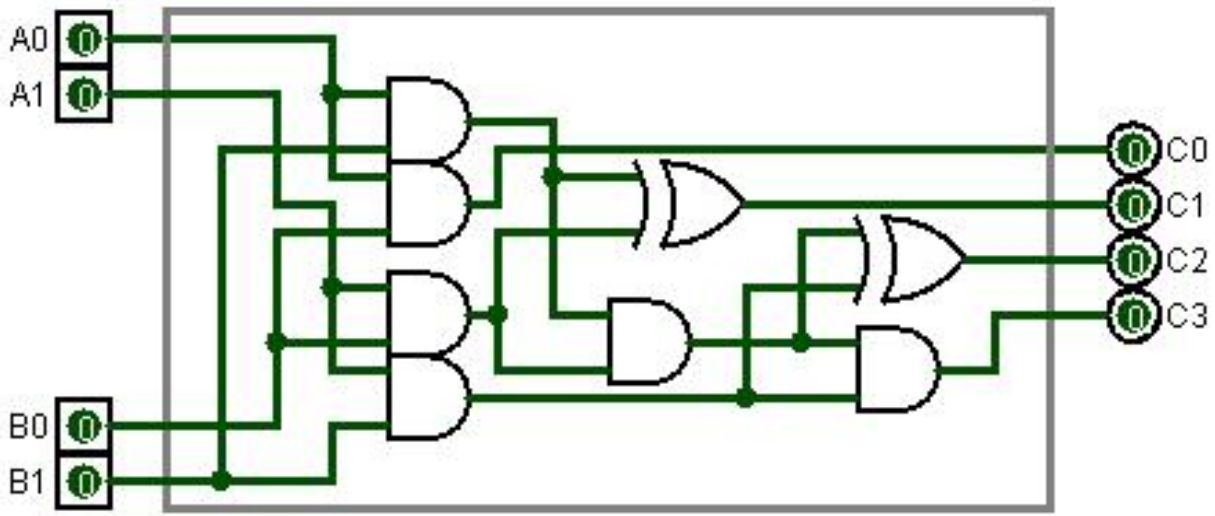
Homomorphic Rounding on Encrypted Plaintexts?

Can we compute the significant digits of four 32-bit integers in a second?

How long does it take to multiply 1024 floating-point numbers?

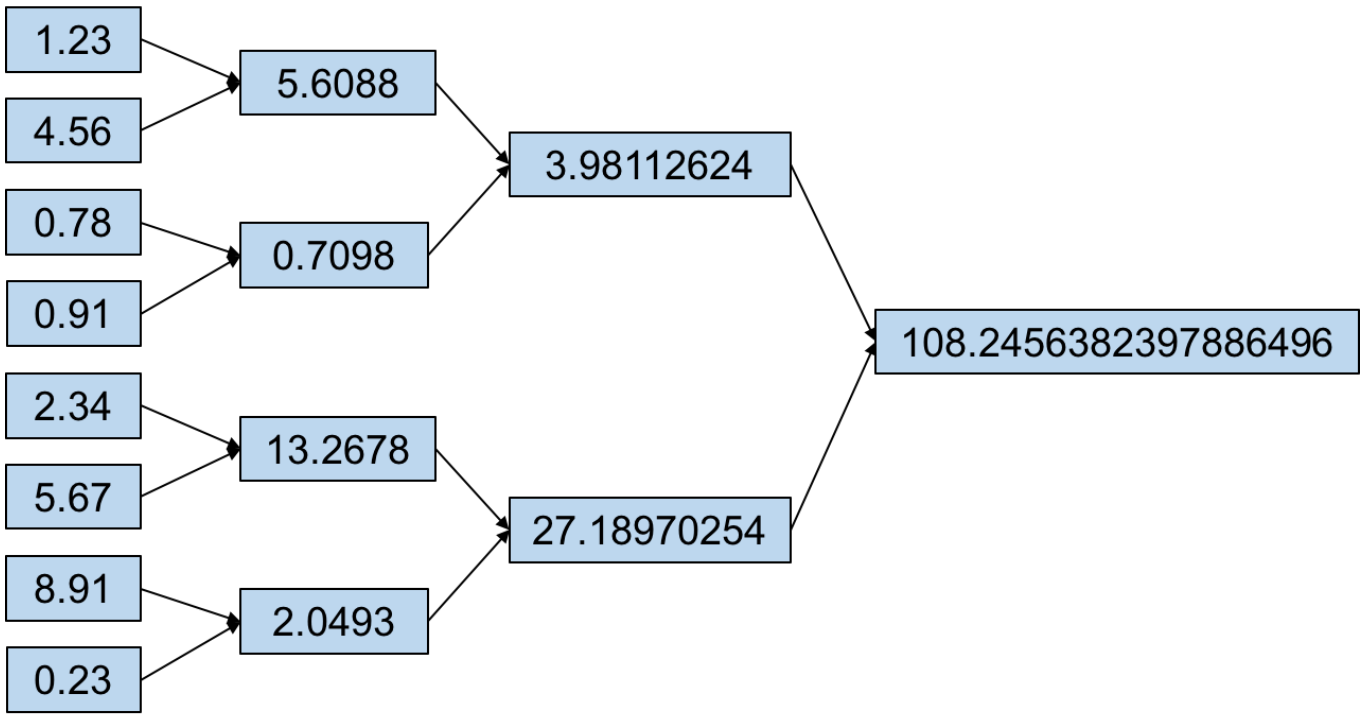
Homomorphic Rounding on Encrypted Plaintexts?

Bit-wise Encryption



- Input values have η-bit of precision.
- The depth of a circuit grows linearly on input precision: $L = \Omega(\eta)$.
  e.g. A single mult with $\eta = 64$ requires a bootstrapping.
- Fast Bootstrapping [DM15]: (Boot-time)$\times$(# gate) $>$ 1min.

Word Encryption



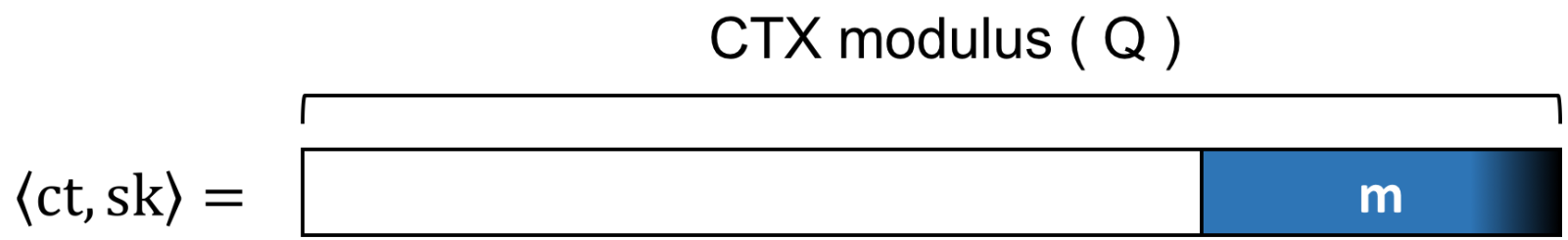
- Bitsize of plaintext grows exponentially on the depth.
- Base Encoding [DGL+15,CSVW16]
- High-Precision [CLPX17]

Our Contributions

- Support approximate addition, multiplication & rounding-off.
- Enable batching techinque with complex plaintexts.
- Evaluate analytic functions (e.g. sigmoid function)

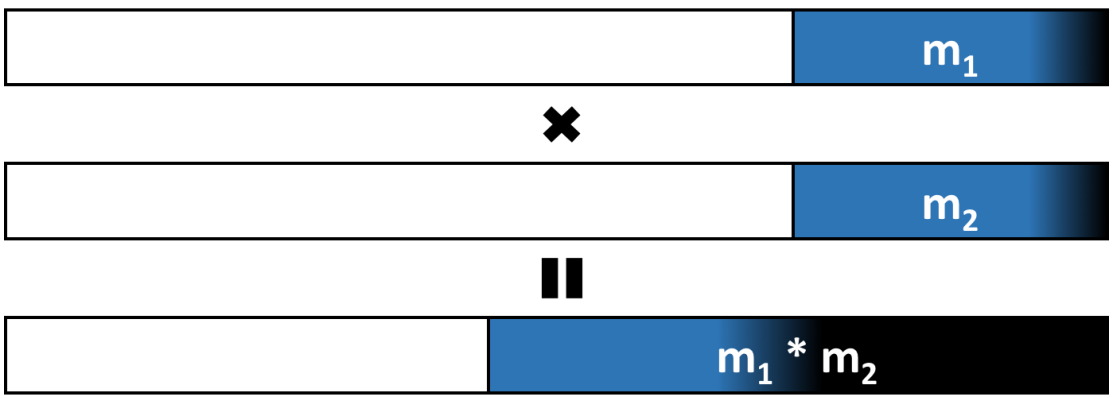
Embracing Noise

- $ct = Enc_{sk}(m)$ satisfies $[\langle ct, sk \rangle]_Q = m + e$ for some small error e.
- Inserted noise is considered to be a part of the computational error from approximate arithmetic.
- The decrypted value $m + e$ is an approximate to the original message.
- The precision of a plaintext is almost preserved.
  e.g. $m = 1.23 * 10^4, e = -17$. $m + e = 12283 \approx m$.



Homomorphic Operations and Precision

- $[\langle ct_i, sk \rangle]_Q \approx m_i \quad \& \quad |m_i| \ll Q$
- $m_1 \cdot (1 \pm r_1) + m_2 \cdot (1 \pm r_2) = (m_1 + m_2) \cdot (1 \pm \max_i r_i)$.
- $m_1 \cdot (1 \pm r_1) * m_2 \cdot (1 \pm r_2) + e_{mult} \approx m_1 m_2 \cdot (1 \pm (r_1 + r_2))$.
- Optimal in the sense of precision loss.

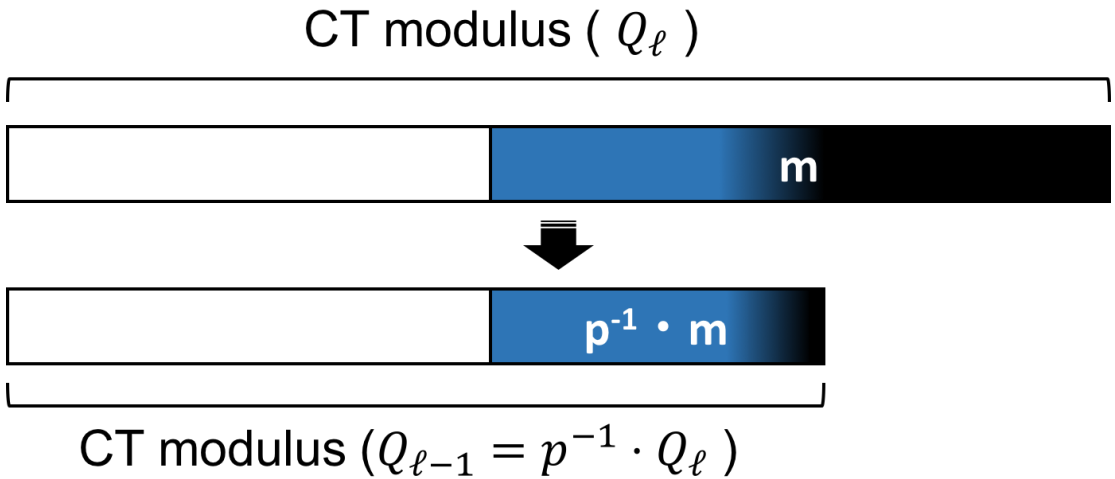


Rescaling Process for Plaintext Rounding

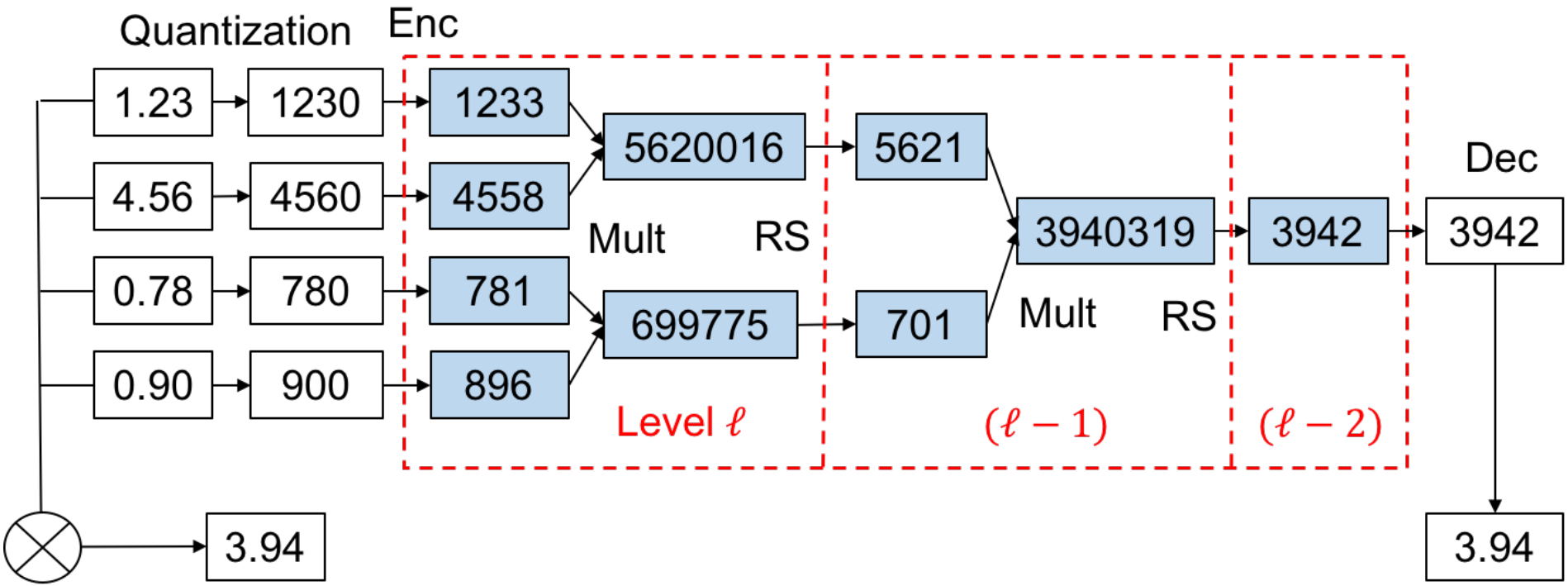
- Divide the ciphertext modulus & encrypted plaintext by the base p.
- $ct \pmod{Q_\ell = p^\ell} \mapsto RS(ct) = \lfloor p^{-1} \cdot ct \rceil \pmod{Q_{\ell-1} = p^{\ell-1}}$.

$$[\langle RS(ct), sk \rangle]_{Q_{\ell-1}} \approx p^{-1} \cdot [\langle ct, sk \rangle]_{Q_\ell}$$

- The relative error is almost preserved.



Leveled Structure



- Evaluation of degree d circuit requires $L = \log d$ levels.
- Precision loss $< (L+1)$ bits.
- $\log Q = \mathcal{O}(L \cdot \log p)$: Linear on depth & precision bits

Batching Technique

- Cyclotomic ring structure $\mathcal{R} = \mathbb{Z}[X]/(\Phi_M(X))$ with $N = \phi(M)$.
- Previous Method:
  - $\Phi_M(X) = \prod_i F_i(X) \pmod t$ & $CRT : \mathcal{R}_t \to \prod_i \mathbb{Z}[X]/(F_i(X))$.
- A plaintext is a small polynomial in $\mathcal{R}$.
  - Evaluate the roots of $\Phi_M(X)$ in algebraic extension $\mathbb{C}$
  - $\Phi_M(X) = \prod_{j \in \mathbb{Z}_M^*}(X - \zeta^j)$ over $\mathbb{C}$ for $\zeta = \exp(-2\pi i/M)$.
- Decoding Map:
  - $\{\zeta_0, \ldots, \zeta_{N/2}\}$: Non-conjugate primitive roots of unity.
  - $m(X) \mapsto (m(\zeta_j))_{0 \le j < N/2}$.

Example

- Decoding map: $m(X) \mapsto (m(\zeta_j))_{0 \le j < N/2} \in \mathbb{C}^{N/2}$.
- $\mathbb{Z}_M^* = \langle 5, -1 \rangle$ when M is a power-of-two.
  Set $\zeta_j = \zeta^{5^j}$ for $\zeta = \exp(-2\pi i/M)$ and $0 \le j < N/2$.
- Example: $M = 8$ ($\Phi_M(X) = X^4 + 1$) and $\Delta = 128$.

$$\vec{z} = (1.2 - 3.4i, 5.6 + 7.8i) \xmapsto{\text{invDFT}} \frac{1}{10}(34 - 39\sqrt{2}X + 22X^2 - 17\sqrt{2}X^3)$$
$$\xmapsto{\lfloor (\cdot) \times \Delta \rceil} m(X) = 435 - 706X + 282X^2 - 308X^3.$$

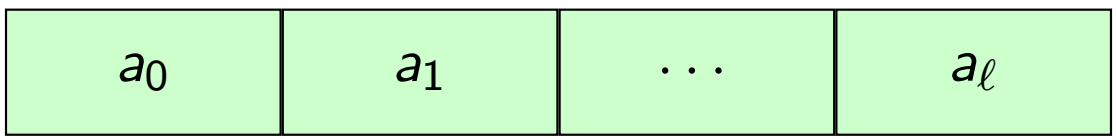
$$m(\zeta) = 128(1.1998.. + i * 3.3984..), m(\zeta^5) = 128(5.5970.. + i * 7.8047..).$$

Rotation

- Let $m(X) = \langle ct, sk \rangle = b(X) + a(X) \cdot s(X) \pmod{Q}$ for $ct = (b(X), a(X))$ and $sk = (1, s(X))$.
- Decoding map: $m(X) \mapsto (m(\zeta_j))_{0 \le j < N/2} \in \mathbb{C}^{N/2}$ for $\zeta_j = \zeta^{5^j}$.
- Slot Rotation
  - $ct' = (b(X^5), a(X^5))$ encrypts $m(X^5)$ w.r.t. $sk' = (1, s(X^5))$.
  - $m(X^5) \mapsto (m(\zeta_{j+1}))_{0 \le j < N/2}$.
- Slotwise Conjugtation
  - $ct'' = (b(X^{-1}), a(X^{-1}))$ encrypts $m(X^{-1})$ w.r.t. $sk'' = (1, s(X^{-1}))$.
  - $m(X^{-1}) \mapsto (\overline{m(\zeta_j)})_{0 \le j < N/2}$.

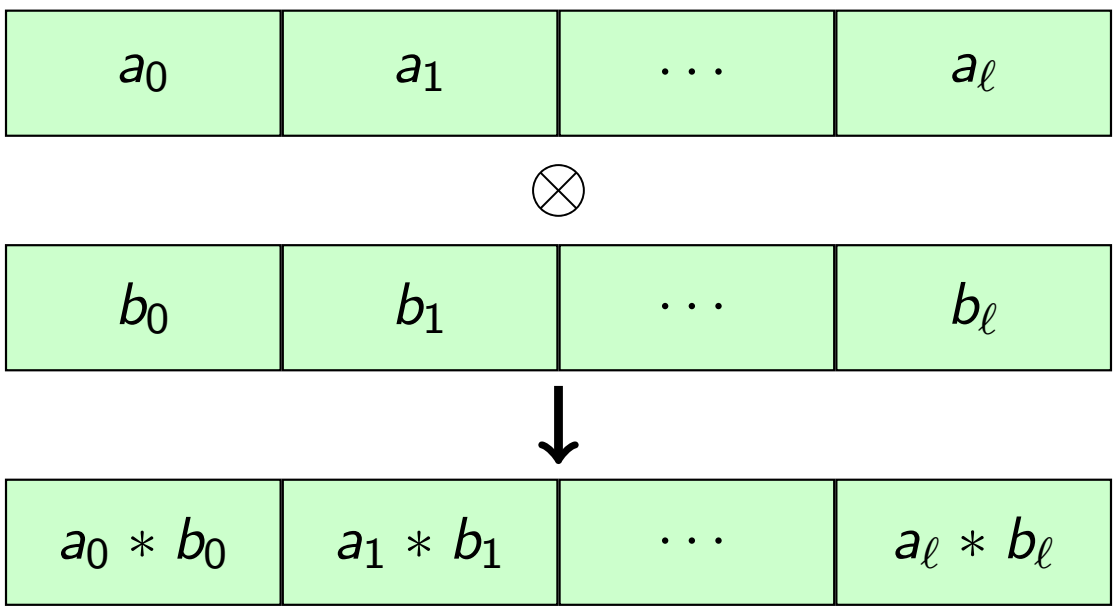
Functionalities

- Packing multiple complex numbers (max. $N/2$) in a single ciphertext.



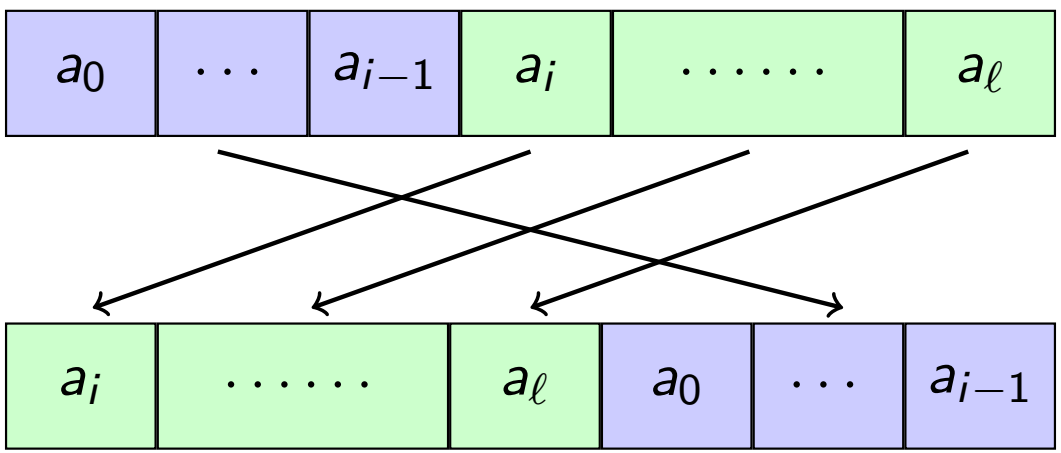
Functionalities

- Packing multiple complex numbers (max. $N/2$) in a single ciphertext.
- Addition, multiplication, and rounding in a SIMD manner.

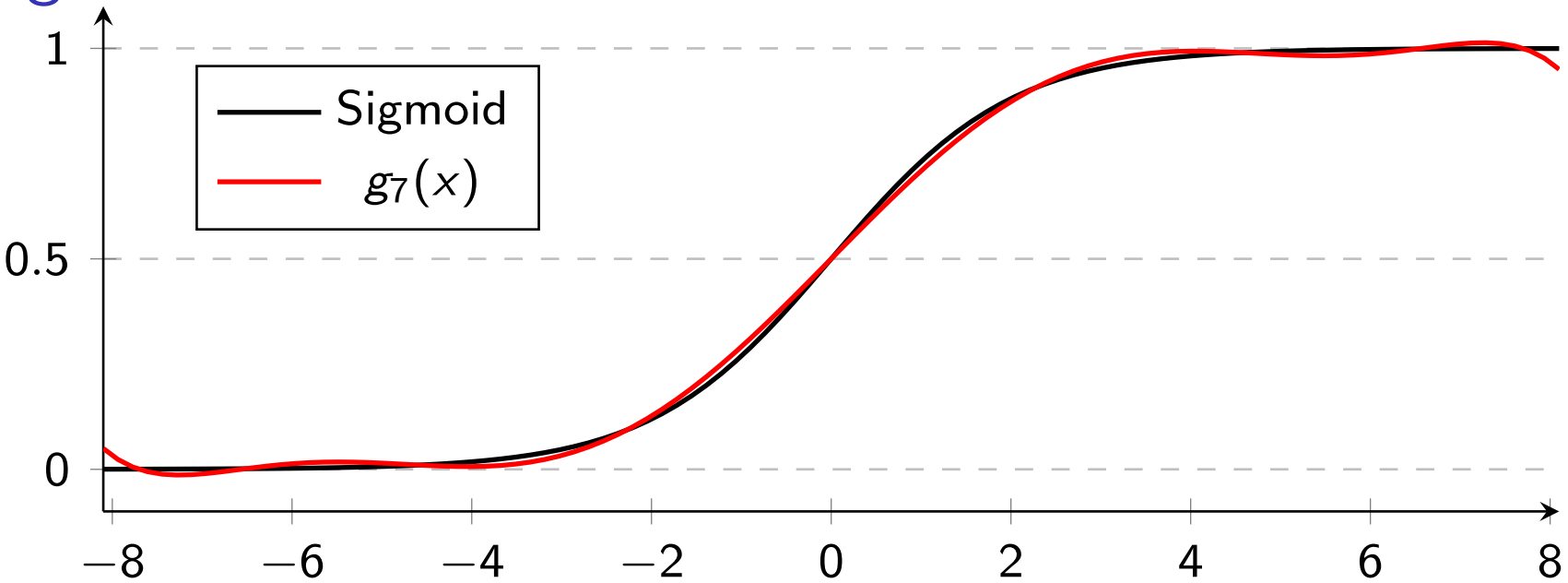


Functionalities

- Packing multiple complex numbers (max. $N/2$) in a single ciphertext.
- Addition, multiplication, and rounding in a SIMD manner.
- Rotation & Conjugation



Sigmoid Function



- Globol Approximation on [-8,8].
- Compute $y = g_7(x)$ homomorphically. $|y - \frac{1}{1+\exp(-x)}| \leq 0.03$.

Input Precision	Depth	Total time	Amortized time
16 bits	3	0.43s	0.10ms

Multiplicative Inverse

- Exponential function: $\exp x$.
- Trigonometric functions: $\cos x, \sin x, \ldots$
- Multiplicative inverse.
  - Let $y = 1 - x$ with $|y| \leq 1/2$.
  - $x^{-1} \approx (1+y)(1+y^2)\cdots(1+y^{2^{L-1}}) = x^{-1} \cdot (1 \pm 2^{-2^L})$.

Input Precision	Depth	Total time	Amortized time
16 bits	4	0.45s	0.11ms

iDASH Genomic S&P Protection Competition

Task3: HE based Logistic Regression Model Learning

- Binary Classification based on 18 features of 1579 records.
- Dataset: 1422 for training & 157 for prediction.
- Learning: 10 min, with 2% of AUC loss
  (On a machine with Xeon CPUs).

Teams	AUC 0.7136	Encryption		Secure learning		Decryption		Overall time (mins)
		Size (MB)	Time (mins)	Time (mins)	Memory (MB)	Size (MB)	Time (mins)	
SNU	0.6934	537.667	**0.060**	**10.250**	2775.333	64.875	**0.050**	10.360
CEA LIST	0.6930	**53.000**	1.303	2206.057	**238.255**	**0.350**	**0.003**	2207.363
KU Leuven	0.6722	4904.000	4.304	155.695	7266.727	10.790	0.913	160.912
EPFL	0.6584	1011.750	1.633	15.089	1498.513	7.125	0.017	16.739
MSR	0.6574	1945.600	11.335	385.021	26299.344	76.000	0.033	396.390
Waseda*	0.7154	20.390	1.178	2.077	7635.600	20.390	2.077	5.332
Saarland	N/A	65536.000	1.633	48.356	29752.527	65536	7.355	57.344

* Interactive mechanism, no complete guarantee on 80-bit security at “analyst” side

HE Standardization Activity

- `http://homomorphicencryption.org`
- White papers about APIs, Security, and Applications.
- 2nd: MIT in Mar. 2018 (1st workshop in Jul. 2017).

- cuHE: This library explores the use of GPGPUs to accel
- HeaAn: This library implements a scheme with native
- HELib: This is an early and widely used library from IB
- $\Lambda \circ \lambda$ (pronounced "L O L"): This is a Haskell library fo FHE.
- NFLlib: This library is an outgrowth of the European H homomorphic encryption using low-level processor p
- PALISADE: This is a general lattice encryption library t including multiple homomorphic encryption schemes.
- SEAL: This is a widely used library from Microsoft that

- Homomorphic Encryption for Arithmetic of Approximate Numbers
- `HEAAN` (慧眼). Available at `http://github.com/kimandrik/HEAAN`.

Protecting REAL Data with HE

Comming Soon!