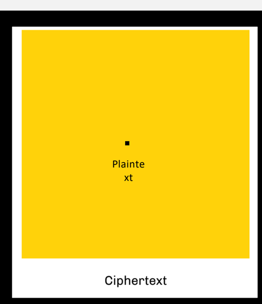
HE is all you need: Compressing FHE Ciphertexts using Additive HE

Rasoul Akhavan Mahdavi, Abdelrahman Diaa, Florian Kerschbaum
University of Waterloo

Problem Statement



FHE Ciphertexts are huge!

Example: Regev ciphertexts are 5KB per byte

Fresh ciphertexts can be compressed by sending the seed.
Processed ciphertexts are much harder to compress!

Intuition

The first step of (R)LWE decryption is linear. So, it can be done using an additive HE scheme

- LWE/RLWE ciphertexts are much larger than the underlying plaintext, i.e., have a large expansion factor.
- Additive ciphertexts have a much smaller expansion factor.
- The client provides an additive encryption of the (R)LWE secret key.
- The server computes the first step of the decryption and return a smaller, additive ciphertext to the client.

LWE Encryption

Algorithm 1 $\text{LWEEncrypt}_{\mathbf{sk}}$

Input: $\mu \in \mathbb{Z}_p$

1: Sample $\mathbf{a} \xleftarrow{\$} \mathbb{Z}_q^n$ and $e \leftarrow \chi$

2: $b = \sum_{i=1}^{n} \mathbf{a}[i] \cdot \mathbf{sk}[i] + \Delta \cdot \mu + e \mod q$

Output: $c = (\mathbf{a}, b)$

Algorithm 2 $\text{LWEDecrypt}_{\mathbf{sk}}$

Input: $c = (\mathbf{a}, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$

1: $\mu^* = (b - \sum_{i=1}^{n} \mathbf{a}[i] \cdot \mathbf{sk}[i]) \mod q$

2: $\mu' = \lfloor \mu^* / \Delta \rceil$

Output: μ'

Linear in sk

RLWE Encryption

Algorithm 3 $\text{RLWEEncrypt}_{S(X)}$

Input: $\mu(X) \in R_p$

1: Sample $A(X) \xleftarrow{\$} R_q$ and $E(X) \leftarrow \chi$

2: $B(X) = A(X) \cdot S(X) + \Delta \cdot \mu(X) + E(X) \mod R_q$

Output: $C = (A(X), B(X))$

Algorithm 4 $\text{RLWEDecrypt}_{S(X)}$

Input: $C = (A(X), B(X)) \in R_q \times R_q$

1: $\mu^*(X) = (B(X) - A(X) \cdot S(X)) \mod R_q$

2: $\mu'(X) = \lfloor \mu^*(X) / \Delta \rceil$

Output: $\mu'(X)$

Linear in S(X)

LWE Compression

Algorithm 5 LWECompress

Input:
$\bar{\mathbf{sk}}[i] = \mathtt{AEnc}_{s_A}(\mathbf{sk}[i])$
$c = (\boldsymbol{a}, b) \in \mathbb{Z}_q^n \times \mathbb{Z}$

1: $x = b \oplus \sum_{i=1}^{n} (q - \mathbf{a}[i]) \otimes \bar{\mathbf{sk}}[i]$

Output: x

Algorithm 6 $\text{ModifiedLWEDecrypt}_s$

Input: Compressed Ciphertext x

1: $y = \mathtt{ADec}_s(x)$

2: $\mu' = \lfloor \frac{y \mod q}{\Delta} \rceil$

Output: $\mu' \in \mathbb{Z}_p$

RLWE Compression

The k^{th} coefficient of the plaintext can be calculated as follows, **which is linear in the secret key!**

$$\mu'[k] = \lfloor \frac{\mu^*[k]}{\Delta} \rceil = \left\lfloor \frac{B[k] - \sum_{i=0}^{k} A[k-i] \cdot S[i] + \sum_{i=k+1}^{N-1} A[N+k-i] \cdot S[i]}{\Delta} \right\rceil$$

Evaluation and Results

	LWE (Prototype)		CGGI (OpenFHE)	RLWE (Prototype)			
	$n = 630$, $\log_2 q = 64$	$n = 750$, $\log_2 q = 64$	$n = 1305$, $\log_2 q = 11$	$N = 1024$, $\log_2 q = 27$	$N = 2048$, $\log_2 q = 54$	$N = 4096$, $\log_2 q = 36$	$N = 8192$, $\log_2 q = 43$
Encrypt Secret Key (Time)	28 s	33 s	0.15 s	45 s	90 s	182 s	369 s
Encrypted Secret Key	483 KB	575 KB	669 KB	786 KB	1572 KB	3145 KB	6290 KB
Ciphertext Compression Time	0.67 s	0.79 s	0.31 s	0.50 s	1.77 s	2.52 s	5.97 s
Compressed Ciphertext Size	768 B	768 B	525 B	767 B	767 B	767 B	767 B
Uncompressed Ciphertext Size	5 KB	6 KB	10.25 KB	2.5 KB	5.6 KB	12.3 KB	26.6 KB
Size Reduction	**86 %**	**87.2 %**	**95.0%**	**70.0%**	**86.36%**	**93.75%**	**97.11%**

- Tested and benchmarked on simple implementation of LWE and RLWE ciphertexts in Python with Paillier as the additive encryption system
- Applied to the CGGI encryption scheme implemented in OpenFHE using Intel Paillier Cryptosystem Library (IPCL) as the additive HE scheme
- Integration with BGV/BFV/CKKS coming soon!

Related Concepts

Scheme Switching	Similar concept but never used for compression Used for reducing circuit depth for bootstrapping [GH11] Switching between scheme to utilize capabilities [BGGJ20]
Modulus Switching	Reduce ciphertext modulus size, limited size reduction
Key Switching	Switches key, but not the scheme
Coefficient Extraction	Converts RLWE to LWE [CGGI16]

Conclusion

- Smaller HE ciphertexts are possible!
- LWE ciphertexts can be compressed up to 95%
- RLWE ciphertext coefficients can be compressed up to 97%

Check out our paper on arxiv!

UNIVERSITY OF WATERLOO

CrySP
Cryptography, Security, and Privacy
— Research Group @ uWaterloo —