On the precision loss in approximate encryption

Anamaria Costache, Benjamin R. Curtis, Erin Hales, Sean Murphy, Tabitha Ogilvie and Rachel Player

September 2, 2022

Introduction

Exact vs Approximate schemes

- Until 2017, all schemes we had were exact
  - i.e. for any allowed circuit f, we had $\texttt{Dec}(f(\texttt{Enc}(m))) = f(m)$[1]
- Recall that most FHE schemes rely on (R)LWE, and thus an encryption is equivalent to creating a (R)LWE instance
  - I.e. we add some gaussian noise e
  - The noise growth is managed via either bootstrapping or modulus switching
  - And is completely removed upon decryption
- The novelty with the CKKS scheme is that it is approximate - the noise is never removed
- This has led to significant efficiency improvements, but the results are now approximate, i.e.

$$\texttt{Dec}(f(\texttt{Enc}(m))) = f(m) + e \approx f(m).$$

[1] Single-input for simplicity but generalises

A side-by-side comparison

Scheme	BGV	BFV	CKKS
Message encoding	$m + t \cdot e$	$\Delta \cdot m + e$	$m + e$
Message encoding	Lower bits	Upper bits	Approximate encryption
Decryption	$m' = \left[[c_0 + c_1 s]_q\right]_t$	$m' = \left[\left\lfloor \frac{t}{q} [c_0 + c_1 s]_q \right\rceil\right]_t$	$m' = [c_0 + c_1 s]_q$
Multiplication	$m_0 m_1 + t^2 e_0 e_1 + t(e_0 m_1 + e_1 m_0)$	$\Delta^2 m_0 m_1 + \Delta(e_0 m_1 + e_1 m_0) + e_0 e_1$	$m_0 m_1 + m_1 e_0 + m_0 e_1 + e_0 e_1$

Noise growth is much slower in CKKS.

Encoding noise

The CKKS scheme uses the canonical embedding to define an encoding from the message space $\mathbb{C}^{N/2}$ to the plaintext space $\mathbb{Z}[X]/(X^N+1)$ in the following way: an isomorphism $\tau : \mathbb{R}[X]/(X^N+1) \to \mathbb{C}^{N/2}$ can be defined by considering the canonical embedding restricted to $N/2$ of the $2N^{\text{th}}$ primitive roots of unity and discarding conjugates. Encoding and decoding then use this map τ, as well as a precision parameter Δ, as follows:

$$\text{Encode}(\mathbf{z}, \Delta) = \lceil \Delta \tau^{-1}(\mathbf{z}) \rfloor, \qquad \text{Decode}(m, \Delta) = \frac{1}{\Delta}\tau(m),$$

where $\mathbf{z} \in \mathbb{C}^{N/2}$, $m \in \mathbb{Z}[X]/(X^N+1)$ and $\lceil \cdot \rfloor$ is taken coefficient-wise.

Our work

What is noise and why is it interesting?

Noise in homomorphic encryption

- All ciphertexts have inherent noise
- Noise grows during homomorphic operations

Good understanding of noise growth is essential

- In exact schemes, either need to determine when to bootstrap or need to know the noise in the output ciphertext
- In approximate schemes, cannot know what the precision loss will be if we do not have a good understanding of noise
- This enables us to choose appropriate parameters, ideally small ones
- The noise in CKKS depends on some secret key material, which has enabled the Li-Micciancio attack

Contributions

- So far estimating the noise has mostly been done on an ad-hoc basis; we provide a rigorous noise analysis of CKKS
- We de-tangle the encoding and encryption noise
- We also present an average-case noise analysis for CKKS
- Provide theoretical bounds for the precision loss
- Provide extensive experimental results

Precision loss due to encryption

We propose three ways of looking at the noise in the ring:

- The Canonical Embedding (CE) analysis, which will serve as our "benchmark"
- A Worst-Case in the Ring (WCR) method, where we follow a worst-case analysis, but remain in the ring
- A Central Limit Theorem (CLT) method, where we trace the variance through the homomorphic operations, and derive a bound at the end of the circuit
    - This is in contrast to previous methods, where we derived a worst-case bound for each operation
    - We introduce a failure probability α, which allows us to refine our results further

Experimental Results

Previous work - [CLP20]

Enc		Add		Mult		ModSwitch	
P	$\overline{x}$	P	$\overline{x}$	P	$\overline{x}$	P	$\overline{x}$
35.0	41.1	34.0	40.2	17.0	26.0	-	-
89.0	97.9	88.0	97.0	70.0	82.4	39.0	38.1
197	209	196	209	177	194	147	150
416	433	415	432	395	416	366	373

Table 1: The observed mean $\overline{x}$ of the noise budget in HElib ciphertexts in 10000 trials, with heuristic estimates of the noise growth denoted by P. Each row corresponds to a parameter set with $n \in \{2048, 4096, 8192, 16384\}$.

Noise in the ring

$\log(N)$	$\log(q)$	Experiments	CLT
Addition noise.			
13	109	10.88	11.40
14	219	11.44	11.93
15	443	12.00	12.45
Multiplication noise.			
13	109	17.31	18.69
14	219	18.38	19.72
15	443	19.43	20.75

Table 2: Average bits of noise observed in the ring over 1000 trials in HEAAN, for $\alpha = 0.0001$ and $\Delta = 2^{40}$.

Results in the complex space

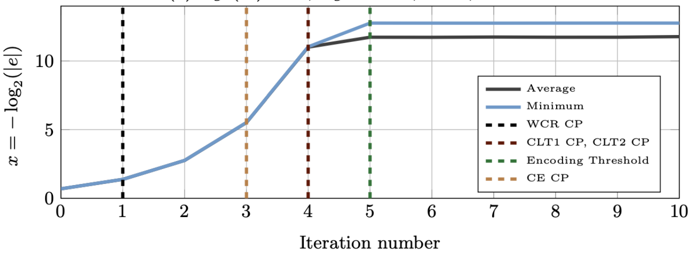
$\log(N)$	$\log(q)$	Experiments	CLT
Addition, complex error.			
13	109	-21.92	-22.55
14	219	-20.72	-21.52
15	443	-19.70	-20.49
Multiplication, complex error.			
13	109	-23.17	-21.51
14	219	-21.68	-19.92
15	443	-20.13	-18.72

Table 3: Average bits of error observed in the message space over 1000 trials in HEAAN, for $\alpha = 0.0001$ and $\Delta = 2^{40}$.

Applications of our results

Iterative algorithms - Newton-Raphson

(b) $log_2(N) = 14$, $log_2\Delta = 50$, $a = 1$, $b = 10$.



(c) $log_2(N) = 15$, $log_2\Delta = 35$, $a = 1$, $b = 15$.

Fig. 1: Accuracy change over successive iterations. Critical Points displayed as vertical lines, using $\alpha = 0.0001$. Note that, in (a) and (b), the values of the CLT1, CLT2 and CE critical points collide, so we plot them as a single line. Similarly, in (c), the value of the CLT1 and CLT2 critical points collide, so we plot them as a single line. All experiments are considered over 100 loops. The number of accurate bits is given by x, as defined in Section 7.1.

The Li-Micciancio attack - an exact scheme

A recent attack by Li and Micciancio ([LM21]) gives a key-recovery attack, by exploiting the fact that the noise contains secret key material.

Definition 2. *(Condition for correctability). Fix parameters, and a circuit* $g : (\mathbb{C}^{N/2})^l \to \mathbb{C}^{N/2}$*. Suppose that the message* $g(\mathbf{z}_1, \ldots, \mathbf{z}_l) + \mathbf{e}$ *is obtained from the decoding and decryption of the output ciphertext of the homomorphic evaluation of the circuit* g *such that* $\|\mathbf{e}\|_\infty < B$ *for some bound* $B > 0$*, with all but negligible probability over the choice of inputs and randomness of encryption. Then* g *is* correctable *for these parameters if* $\frac{1}{\Delta'} g(\mathbf{z}_1, ..., \mathbf{z}_l) \in \mathbb{Z}[i]^{N/2}$*, where* $\Delta' = 2^{\lceil \log B \rceil + 1}$*, for all feasible inputs* $\mathbf{z}_i$*. We will call this* Δ' *a* correcting factor*.*

The Li-Micciancio attack

- **Noise flooding:** Since the ciphertext decrypts to $m + e$, and e may leak secret key information, we can "drown" e with fresh noise e' and output $m + e + e'$. This has already been adopted by PALISADE
    - Our work allows us to determine precisely the distribution of e'
- **Only real decoding:** Only release the real part of the decoding, as adopted in SEAL
    - We provide the theoretical justification for this

Thank you!

https://eprint.iacr.org/2022/162

Code will be published soon