On the precision loss in approximate homomorphic encryption

Anamaria Costache[1], Benjamin R. Curtis[2], Erin Hales[3], Sean Murphy[3], Tabitha Ogilvie[3], and Rachel Player[3]

[1] Norwegian University of Science and Technology (NTNU), Norway
anamaria.costache@ntnu.no
[2] Zama, Paris, France
ben.curtis@zama.ai
[3] Royal Holloway, University of London, UK
{erin.hales.2018}, {tabitha.ogilvie.2019} @live.rhul.ac.uk
{s.murphy}, {rachel.player} @rhul.ac.uk

Abstract. Since its introduction at Asiacrypt 2017, the CKKS approximate homomorphic encryption scheme has become one of the most widely used and implemented homomorphic encryption schemes. Due to the approximate nature of the scheme, application developers using CKKS must ensure that the evaluation output is within a tolerable error of the corresponding plaintext computation. Choosing appropriate parameters requires a good understanding of how the noise will grow through the computation. A strong understanding of the noise growth is also necessary to limit the performance impact of mitigations to the attacks on CKKS presented by Li and Micciancio (Eurocrypt 2021).

In this work we present a comprehensive noise analysis of CKKS, that considers noise coming both from the encoding and homomorphic operations. Our main contribution is the first average-case analysis for CKKS noise, and we also introduce refinements to prior worst-case noise analyses. We develop noise heuristics both for the original CKKS scheme and the RNS variant presented at SAC 2018. We then evaluate these heuristics by comparing the predicted noise growth with experiments in the HEAAN and FullRNS-HEAAN libraries, and by comparing with a worst-case noise analysis as done in prior work. Our findings show mixed results: while our new analyses lead to heuristic estimates that more closely model the observed noise growth than prior approaches, the new heuristics sometimes slightly underestimate the observed noise growth. This evidences the need for implementation-specific noise analyses for CKKS, which recent work has shown to be effective for implementations of similar schemes.

1 Introduction

Homomorphic Encryption (HE) enables computation on ciphertexts without revealing any information about the underlying plaintexts. The first scheme was proposed by Gentry [19] and, since then, many homomorphic encryption schemes have been proposed [4, 18, 11, 10] based on the security of the Learning with Errors (LWE) problem [39] and its variants.

One of the most popular schemes is the approximate homomorphic encryption scheme CKKS [10], which we describe in Section 2. Ciphertexts in all homomorphic encryption schemes based on LWE variants contain noise, which grows with each evaluation operation, and must be carefully controlled to ensure correct decryption. The main insight of [10] is that it may be tolerable for decryption to be approximate, for example in applications where we expect small errors to occur. This enables the CKKS scheme to natively support real-valued plaintexts, making it attractive for application settings such as privacy-preserving machine learning [27, 3, 37]. In contrast, other similar schemes such as BGV [4] or BFV [18], are exact, and thus have a finite plaintext space that data must be encoded into. CKKS has been extensively optimised [7, 8, 28] and is implemented in many prominent open-source homomorphic encryption libraries [23, 25, 30, 2, 38, 40].

Homomorphic encryption schemes involve many different parameters, and it can be a challenge to choose appropriate parameters that balance efficiency, security, and noise growth. This is particularly true for the CKKS scheme, for two main reasons. Firstly, unlike for exact schemes, encoding and encryption noises must be considered together. Secondly, in CKKS, we have to track not only the level of ciphertexts (as in BFV and BGV), but we must also track the scaling factor Δ. Unfortunately, there is no clear guidance for choosing Δ and a trial-and-error approach is usually advised[4].

Prior noise analyses for CKKS [10, 7, 8, 21, 28] employ a worst-case analysis in the canonical embedding, in analogue to the line of work [13, 14, 20] for analysing noise growth in BGV and BFV. In particular, a worst-case bound on the noise of each ciphertext in the canonical embedding is tracked through each homomorphic operation. This leads to a bound on the noise in the output ciphertext, which can be used to set parameters for correctness. These worst-case bounds are developed assuming that the random variable falls within a certain multiple of standard deviations (e.g. six [20] or ten [21]) from its mean. We can thus expect the bounds to be loose even from the beginning of the computation (as a freshly sampled noise is likely to be closer to the mean than several standard deviations away), and that the looseness will compound as we move further through the computation. This intuition was confirmed in experiments of [14] for the BFV and BGV schemes, whose operations are similar to those of CKKS.

An alternative, average-case approach to noise analysis was proposed in [12] for the CGGI scheme [11], in which the noise is modelled as a Gaussian, and its variance is tracked through each homomorphic operation. The noise in the output ciphertext is finally bounded from the output variance, in order to pick parameters for correctness. Adopting a similar approach for CKKS appears challenging, as the noise after a homomorphic multiplication is a product of the noises in the two input ciphertexts, whereas as the output distribution of the product

[4] See e.g. `https://ibm.github.io/fhe-toolkit-linux/html/helib/md__opt__i_b_m__f_h_e-distro__h_elib__c_k_k_s-security.html`

of two subgaussians is not necessarily subgaussian, and can have a much heavier tail [36]. In this work, we will demonstrate how a Central Limit Theorem approach can be applied to give a heuristic average-case noise growth analysis for CKKS.

Contributions: Our first contribution is a new result relating the CKKS message and plaintext spaces. Recall that CKKS encoding maps an element from the (complex) message space into an element in the (polynomial ring) plaintext space via a scaled restriction of the inverse canonical embedding. In Theorem 1, we provide a new, tighter bound relating the size of an error in the plaintext space to the size of the induced error in the message space. Moreover, we prove that this bound is the best possible. In addition, we show that the worst case expansion factor in either the real or complex part of our message equals the worst case expansion factor of the entire embedding. This means that, perhaps surprisingly, bounding a decrypted and decoded message over only the real part, rather than the whole embedding, provides no benefit for worst-case analyses.

Our next contribution is to present the first average-case noise analysis for CKKS. In Theorem 3 we give a result showing that the product of two Normally distributed polynomials has Normally distributed coefficients under a Central Limit assumption. Using this result we are able to heuristically model all CKKS noise operations as operations on Normal random variables, thus recovering an analysis similar to [12]. We present our noise analyses for ‘Textbook’ CKKS as originally presented in [10] and the RNS variant presented in [8].

In order to evaluate the efficacy of our average-case noise analysis for CKKS, we compare the noise heuristics developed under this analysis with the worst-case bounds of prior work. We also present refinements to these prior worst-case noise analyses using the techniques of [26]. We parameterise all our noise bounds in terms of a failure probability, α, rather than a-priori fixing a one dimensional failure probability as in prior work [20, 13, 14]. We evaluate the bounds arising from all these noise analyses with experiments in HEAAN v1.0 [24] and FullRNS-HEAAN [22]. We note that neither the Textbook CKKS nor the RNS variant noise analysis is implementation-specific, and we chose the HEAAN library as it is the implementation that most closely resembles the theoretical description of both variants of the scheme.

Our experimental results are given in Table 4 for Textbook CKKS heuristics as compared with HEAAN v1.0 [24] and in Table 5 for heuristics for the RNS variant [8] as compared with FullRNS-HEAAN [22]. Our results show that our new heuristics improve upon prior noise analyses in terms of modelling more closely the observed noise. However, we also observe that our heuristics may underestimate the noise growth observed in practice. Prior work [15] for BGV has noted another example of a noise analysis that was not implementation specific that also led to underestimates of the observed noise. Our work can therefore be seen as an improved starting point for a tight noise analysis for CKKS, but an implementation-specific analysis may be more suitable for applications that cannot afford this underestimate.

As an additional contribution, we consider the recent key recovery attack for CKKS presented by Li and Micciancio [33]. The attack exploits the fact that the noise ϵ of an output $m + \epsilon$ from CKKS decryption depends on the secret key. We discuss how an improved noise analysis for CKKS can support one possible mitigation for this attack. In particular, we show how to develop an IND-CPA^D secure exact variant of CKKS for correctable circuits.

Related work: Average-case noise analyses were presented for the BGV scheme in [35], by applying results from the present work. An implementation-specific average-case noise analysis for the BGV scheme was presented in [15]. An average-case noise analysis for BFV was presented in [**?**].

Lee *et al.* [31] use the signal-to-noise ratio to analyse CKKS noise, and proposes to track the variance of the errors, rather than an upper bound. The variances of the noise in multi-key BFV and CKKS operations were tracked in [6], but a proof that the noises are distributed as Gaussians was not presented. Our work thus provides a theoretical justification for these approaches. Our study of encoding also provides theoretical support for the heuristics in [21].

Mitigations for the Li-Micciancio attack were also discussed in [9, 33, 34]. In [17], an approach is presented that can increase the number of correctable circuits for a fixed scale.

Structure: In Section 2 we introduce relevant background material and notation, including the Textbook CKKS scheme [10] and its RNS variant [8]. In Section 3 we study the precision loss coming from encoding and decoding in CKKS. In Section 4 we describe the three methods for noise analysis that we will apply to Textbook CKKS and its RNS variant. We then apply this to Textbook CKKS. In Section 5 we describe the modifications required to the noise analysis methods for the RNS setting, and provide heuristics for this setting. In Section 6 we report on experimental results to evaluate the noise analysis approaches that we introduced. In Section 7 we discuss an application of improved CKKS noise analyses to mitigate the Li-Micciancio attack on CKKS.

2 Preliminaries

Notation: Vectors are denoted in small bold font $\mathbf{z}$, and z_j refers to the j^{th} element of a vector, indexing from zero. The notation $\lfloor \cdot \rceil$ is used for rounding to the nearest integer and $[\cdot]_q$ represents reduction modulo q. For $z = x + iy \in \mathbb{C}$, we denote by $\lceil z \rfloor := \lceil x \rfloor + i \lceil y \rfloor$ the rounding of both its real and imaginary components, and extend this componentwise to define the rounding $\lceil \mathbf{z} \rfloor$ of a complex vector $\mathbf{z} \in \mathbb{C}^{N/2}$. Unless otherwise stated, log will always mean $\log_2$.

In this work, we will consider several different norms. We denote the p-norm by $\|\cdot\|_p$ and the infinity norm by $\|\cdot\|_\infty$. We consider norms on a polynomial m both as a vector of its coefficients and under the canonical embedding, and denote these norms by $\|m\|$ and $\|m\|^{\mathsf{can}}$ respectively. We use $s \leftarrow D$ to denote sampling s according to the distribution D.

$\texttt{SecretKeyGen}(\lambda)$: Sample $s \leftarrow S$ and output $\texttt{sk} = (1, s)$.

$\texttt{PublicKeyGen}(\texttt{sk})$: For $\texttt{sk} = (1, s)$, sample $a \leftarrow R_Q$ uniformly at random and $e \leftarrow \chi$. Output $\texttt{pk} = ([-as + e]_Q, a)$.

$\texttt{EvaluationKeyGen}(\texttt{sk}, w)$: Sample $a' \leftarrow R_{P \cdot Q}$ uniformly at random and $e' \leftarrow \chi$. Output $\texttt{evk} = \big([-a's + e' + Ps^2]_{P \cdot Q}, a'\big)$.

$\texttt{Encrypt}(\texttt{pk}, m)$: For the message $m \in R$. Let $\texttt{pk} = (p_0, p_1)$, sample $v \leftarrow V$ and $e_1, e_2 \leftarrow \chi$. Output $\texttt{ct} = ([m + p_0 v + e_1]_Q, [p_1 v + e_2]_Q)$.

$\texttt{Decrypt}(\texttt{sk}, \texttt{ct})$: Let $\texttt{ct} = (c_0, c_1)$. Output $m' = [c_0 + c_1 s]_Q$.

$\texttt{Add}(\texttt{ct}_0, \texttt{ct}_1)$: Output $\texttt{ct} = ([\texttt{ct}_0[0] + \texttt{ct}_1[0]]_Q, [\texttt{ct}_0[1] + \texttt{ct}_1[1]]_Q)$.

$\texttt{Pre-Multiply}(\texttt{ct}_0, \texttt{ct}_1)$: Set $d_0 = [\texttt{ct}_0[0]\texttt{ct}_1[0]]_Q$, $d_1 = [\texttt{ct}_0[0]\texttt{ct}_1[1] + \texttt{ct}_0[1]\texttt{ct}_1[0]]_Q$, and $d_2 = [\texttt{ct}_0[1]\texttt{ct}_1[1]]_Q$. Output $\texttt{ct} = (d_0, d_1, d_2)$.

$\texttt{KeySwitch}(\texttt{ct}, \texttt{evk})$: Here, $\texttt{ct}$ is an output of $\texttt{Pre-Multiply}$. Let $\texttt{ct}[0] = d_0$, $\texttt{ct}[1] = d_1$ and $\texttt{ct}[2] = d_2$. Recall $\texttt{evk}[0] = -a's + e' + Ps^2$ and $\texttt{evk}[1] = a'$. Set $c_0' = [d_0 + \lfloor P^{-1} \cdot d_2 \cdot (-a's + e' + Ps^2) \rceil]_Q$, and $c_1' = [d_1 + \lfloor P^{-1} \cdot d_2 \cdot a' \rceil]_Q$. Output $\texttt{ct}' = (c_0', c_1')$.

$\texttt{Rescale}(\texttt{ct}, \Delta)$: For $\texttt{ct} = (c_0, c_1)$ a ciphertext at level ℓ. Set $c_0' = \big[\lfloor \frac{1}{\Delta} c_0 \rceil\big]_{Q_{\ell-1}}$ and $c_1' = \big[\lfloor \frac{1}{\Delta} c_1 \rceil\big]_{Q_{\ell-1}}$. Output $\texttt{ct} = (c_0', c_1')$.

$\texttt{Multiply}(\texttt{ct})$: Here, $\texttt{ct}$ is an output of $\texttt{Pre-Multiply}$. We apply $\texttt{KeySwitch}$ followed by $\texttt{Rescale}$. Output $\texttt{ct} = \frac{1}{\Delta} \big[(d_0, d_1) + \lfloor P^{-1} d_2 \texttt{evk} \rceil\big]_Q$.

Fig. 1: The Textbook CKKS Scheme

We use the notation $\mathrm{N}(\mu, \sigma^2)$ to refer to a univariate Normal distribution with mean μ and variance σ^2, and $\mathrm{N}(\boldsymbol{\mu}; \Sigma)$ to refer to an N-dimensional multivariate Normal distribution with N-dimensional mean vector $\boldsymbol{\mu}$ and $N \times N$ covariance matrix Σ. For a polynomial $Z(X) \in \mathbb{R}[X]/(X^N + 1)$, we will write $Z \sim \mathrm{N}(\boldsymbol{\mu}, \rho^2 I_N)$ to indicate that each coefficient of Z is independently and identically normally distributed, i.e., $Z_i \sim \mathrm{N}(\mu_i, \rho^2)$. We denote by $\texttt{erf}$ the (Gauss) error function, by $\texttt{erf}^{-1}$ its inverse, and by $\texttt{erfc}$ the complementary function.

The Textbook CKKS scheme: The CKKS scheme as originally presented in [10] is a levelled HE scheme that we refer to as *Textbook CKKS*. The scheme is specified in Figure 1.

The Textbook CKKS scheme is parameterised by L, p, q_0, N, λ, χ, S, V, and Δ. The base $p > 0$ and modulus q_0 are used to form the scale parameter and the chain of moduli (one for each level) as follows: $\Delta = 2^p$ and $Q_\ell = \Delta^\ell q_0$ for $1 \leqslant \ell \leqslant L$. The dimension N is typically chosen as a power of two, and we will only use such N in this work. The dimension N and the chain of moduli parameterise the underlying plaintext and ciphertext rings. The plaintext space is $\mathcal{R} = \mathbb{Z}[X]/(X^N + 1)$. We denote by Q some fixed level in the description below, so that the ciphertext space at any given moment is $\mathcal{R}_Q = \mathbb{Z}_Q[X]/(X^N + 1)$.

The security parameter is λ. The Ring-LWE error distribution is denoted by χ and is such that each coefficient is sampled as a discrete Gaussian with standard deviation $\sigma = 3.2$ [1]. The parameter S denotes the secret key distribution, which is specified in [10] to be $HWT(h)$, i.e. the secret is ternary with Hamming weight exactly h. The parameter V denotes the ephemeral secret distribution, which is specified in [10] to be $ZO(\rho)$ with $\rho = 0.5$, i.e. the secret is ternary with coefficients having probability $\rho/2$ for each of -1 and 1, and probability $1 - \rho$ of being 0.

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 via considering the canonical embedding restricted to $N/2$ of the $2N^{\text{th}}$ primitive roots and discarding conjugates. Encoding and decoding then use this map τ, as well as a precision parameter Δ, as follows: $\text{Encode}(\mathbf{z}, \Delta) = \lceil \Delta\tau^{-1}(\mathbf{z})\rfloor$ and $\text{Decode}(m, \Delta) = \frac{1}{\Delta}\tau(m)$, where $\mathbf{z} \in \mathbb{C}^{N/2}$, $m \in \mathbb{Z}[X]/(X^N + 1)$, and $\lceil\cdot\rfloor$ is taken coefficient-wise.

RNS variants of CKKS: Variants of CKKS using RNS have been proposed [8, 28]. In this work, we focus on the RNS-CKKS scheme as described in [8]. This scheme is specified in Figure 2.

In RNS variants of CKKS [8, 28], the chain of ciphertext moduli changes compared to the original scheme. The ℓ^{th} ciphertext modulus is given by $Q_\ell = \prod_{j=0}^{\ell} q_j$ where the j^{th} ciphertext slot is with respect to the modulus q_j. In the RNS variant [8], the key switching procedure requires the large modulus P to be formed similarly from a set of k pairwise coprime p_i as $P = \prod_{i=0}^{k} p_i$. The other parameters as specified in [8], and the encoding and decoding, are the same as for Textbook CKKS.

Precision Loss: in this work we are concerned with bounding the precision loss in CKKS, which we can define informally as the difference between evaluating a circuit in the clear and evaluating the same circuit homomorphically. A more formal description is given below.

Definition 1. *Consider a normed space $(\mathcal{M}, ||\cdot||)$, messages $m_1, ..., m_n \in \mathcal{M}$, and a circuit $C : \mathcal{M}^n \to \mathcal{M}$. Then we define the* ***precision loss*** *associated with calculating the circuit C homomorphically as the distance $||\tilde{m} - m||$, where $\tilde{m}$ is the output of the homomorphic evaluation of $C(m_1, ..., m_n)$, and m is the true value of the circuit.*

This definition is similar to Definition 10 of [**?**]. We will consider precision loss in three spaces: firstly, the plaintext space R with infinity norm on the vector of coefficients; secondly, the message space $\mathbb{C}^{N/2}$ with infinity norm, which is equivalent to R with infinity canonical norm; and lastly the projection to the real part $\mathbb{R}^{N/2}$ with infinity norm.

3 Encoding Analysis

In this section, we give theoretical bounds on the precision loss from encoding and decoding. To understand precision loss due to encoding, as well as translate

$\mathtt{SecretKeyGen}(\lambda)$**:** Sample $s \leftarrow S$ and output $\mathtt{sk} = (1, s)$.

$\mathtt{PublicKeyGen}(\mathtt{sk})$**:** For $\mathtt{sk} = (1, s)$, for all $0 \leq j \leq L$, a representative $a^{(j)}$ is sampled uniformly from R_{q_j}, and $b^{(j)} \leftarrow -a^{(j)}s + e \mod q_j$ is set. Output $\mathtt{pk} = (\mathtt{pk}^{(j)})_{0 \leq j \leq L} = (b^{(j)}, a^{(j)})_{0 \leq j \leq L}$.

$\mathtt{EvaluationKeyGen}(\mathtt{sk}, w)$**:** Sample $e' \leftarrow \chi$. Output $(\mathtt{evk}^{(0)}, \ldots, \mathtt{evk}^{(k+L)}) = ((b'^{(0)}, a'^{(0)}), \ldots, (b'^{(k+L)}, a'^{(k+L)}))$, where, for $0 \leq i < k$, $a'^{(i)} \leftarrow R_{p_i}$ uniformly and $b'^{(i)} = -a'^{(i)}s + e' \mod p_i$; and for $0 \leq j \leq L$, $a'^{(k+j)} \leftarrow R_{q_j}$ uniformly and $b'^{(k+j)} = -a'^{(k+j)}s + [P]_{q_j} s^2 + e' \mod q_j$.

$\mathtt{Encrypt}(\mathtt{pk}, m)$**:** For $m \in R$. Sample $v \leftarrow V$ and $e_0, e_1 \leftarrow \chi$. For all $0 \leq j \leq L$, and for the public key $\mathtt{pk} = (\mathtt{pk}^{(j)})_{0 \leq j \leq L} = (b^{(j)}, a^{(j)})_{0 \leq j \leq L}$, output $\mathtt{ct} = (\mathtt{ct}^{(j)})_{0 \leq j \leq L}$ where $\mathtt{ct}^{(j)} = (b^{(j)}v + e_0 + m, a^{(j)}v + e_1) \in R_{q_j}^2$.

$\mathtt{Decrypt}(\mathtt{sk}, \mathtt{ct})$**:** For $\mathtt{ct} = (\mathtt{ct}^{(j)})_{0 \leq j \leq \ell}$, output $m' = \left\langle \mathtt{ct}^{(0)}, \mathtt{sk} \right\rangle \mod q_0$.

$\mathtt{Add}(\mathtt{ct}_0, \mathtt{ct}_1)$**:** For $0 \leq j \leq \ell$, for input ciphertexts $\mathtt{ct}_1 = \{\mathtt{ct}_1^{(j)}\}$ and $\mathtt{ct}_2 = \{\mathtt{ct}_2^{(j)}\}$, output $\mathtt{ct}_{\text{add}} = (\mathtt{ct}_{\text{add}}^{(j)})_{0 \leq j \leq \ell}$ where $\mathtt{ct}_{\text{add}}^{(j)} = \mathtt{ct}_1^{(j)} + \mathtt{ct}_2^{(j)} \mod q_j$.

$\mathtt{Pre\text{-}Multiply}(\mathtt{ct}_0, \mathtt{ct}_1)$**:** For $0 \leq j \leq \ell$, for input ciphertexts $\mathtt{ct}_1 = \{\mathtt{ct}_1^{(j)}\} = \left\{\left(c_0^{(j)}, c_1^{(j)}\right)\right\}$ and $\mathtt{ct}_2 = \{\mathtt{ct}_2^{(j)}\} = \left\{\left(C_0^{(j)}, C_1^{(j)}\right)\right\}$, output $\mathtt{ct}_{\text{pre-mult}} = \{\mathtt{ct}_{\text{pre-mult}}^{(j)}\}_{0 \leq j \leq \ell} = \{(d_0^{(j)}, d_1^{(j)}, d_2^{(j)})\}$ where $d_0^{(j)} = c_0^{(j)} C_0^{(j)} \mod q_j$, $d_1^{(j)} = c_0^{(j)} C_1^{(j)} + c_1^{(j)} C_0^{(j)} \mod q_j$, and $d_2^{(j)} = c_1^{(j)} C_1^{(j)} \mod q_j$.

$\mathtt{KeySwitch}(\mathtt{ct}, \mathtt{evk})$**:** For $0 \leq j \leq \ell$, for input ciphertext $\mathtt{ct}_{\text{pre-mult}} = \{\mathtt{ct}_{\text{pre-mult}}^{(j)}\}_{0 \leq j \leq \ell} = \{(d_0^{(j)}, d_1^{(j)}, d_2^{(j)})\}$, output $\mathtt{ct}_{\text{ks}} = \{\mathtt{ct}_{\text{ks}}^{(j)}\}_{0 \leq j \leq \ell} = \left\{\left(c_0^{(0)}, c_1^{(0)}\right), \ldots, \left(c_0^{(\ell)}, c_1^{(\ell)}\right)\right\}$, where $\left(c_0^{(j)}, c_1^{(j)}\right) = \left([d_0^{(j)} + \hat{c}_0^{(j)}]_{q_j}, [d_1^{(j)} + \hat{c}_1^{(j)}]_{q_j}\right)$, for $\hat{c}_0^{(j)}$ and $\hat{c}_1^{(j)}$ as defined in Supplementary Material Section E.

$\mathtt{Rescale}(\mathtt{ct})$ **:** For $0 \leq j \leq \ell$, for input ciphertext $\mathtt{ct} = \{\mathtt{ct}^{(j)}\}_{0 \leq j \leq \ell} = \left((c_0^{(j)}, c_1^{(j)})\right)_{0 \leq j \leq \ell}$ output$\mathtt{ct}_{\text{rs}} = \{\mathtt{ct}_{\text{rs}}^{(j)}\}_{0 \leq j \leq \ell - 1} = \{(c_0'^{(j)}, c_1'^{(j)})\}_{0 \leq j \leq \ell - 1}$, where $c_0'^{(j)} = q_\ell^{-1}(c_0^{(j)} - c_0^{(\ell)}) \mod q_j$ and $c_1'^{(j)} = q_\ell^{-1}(c_1^{(j)} - c_1^{(\ell)}) \mod q_j$.

$\mathtt{Multiply}(\mathtt{ct}_0, \mathtt{ct}_1)$**:** Output $\mathtt{Rescale}(\mathtt{KeySwitch}(\mathtt{Pre\text{-}Multiply}(\mathtt{ct}_0, \mathtt{ct}_1), \mathtt{evk}))$.

Fig. 2: The RNS-CKKS Scheme

noise bounds derived in the plaintext space to noise bounds in the message space, we investigate how distance measured in $\mathbb{R}[X]/(X^N + 1)$ corresponds to distance measured in $\mathbb{C}^{N/2}$, when we move between the two via τ for N a power of 2.

If we measure using the 2-norm in both spaces, these two distances correspond exactly as here τ gives a scaled isometry with $\|\tau(m)\|_2 = \sqrt{\frac{N}{2}} \|m\|_2$. However, we will use the infinity norm in both spaces to support our Worst Case in the Ring analysis (see Section 4). We find that, in the worst case, there is an $O(N)$ expansion in the infinity norm under the map τ and unlike the 2-norm, there is no contraction under the map τ^{-1}.

The section is organised as follows. In Section 3.1, we develop new theoretical results on the relationships between distances in the two spaces. In Section 3.2, we then apply these results in the context of CKKS encoding and decoding.

3.1 Mapping Theory

Lemma 1 **([16]).** *Let $m \in \mathbb{R}^N$. Then $\|m\|_\infty \leqslant \|m\|_\infty^{\mathsf{can}}$.*

This inequality is best possible in the sense that it is achieved: let $m = \tau^{-1}(\mathbf{z})$ and let $z_k = B\zeta_k^j$ for $0 \leqslant k \leqslant \frac{N}{2} - 1$, so that $\|\mathbf{z}\|_\infty = B$. Then we find $\|m\|_\infty^{\mathsf{can}} = \|\mathbf{z}\|_\infty = B = |m_j| = \|m\|_\infty$. In particular, there is no contraction as we move from $\mathbb{C}^{N/2}, \|\cdot\|_\infty$ to $\mathbb{R}^N, \|\cdot\|_\infty$ but there is an expansion as we move the other way. The prior result on this bound is as follows.

Lemma 2 **([16, 20]).** *Let $m \in \mathbb{R}^N$. Then $\|m\|_\infty^{\mathsf{can}} \leqslant N \|m\|_\infty$.*

Using generic proof methods and properties of the norm, we can reduce this factor to $N/\sqrt{2}$. Before improving further, we require some definitions and a Lemma. The proof technique of the following Lemmas 3 and 4 is adapted from [5]. We introduce the notation $I(N, j)$ and $I(N)$ as follows:

$$I(N, j) := \sum_{k=0}^{N-1} \left| \sin\left(\frac{jk\pi}{N}\right) \right|, \qquad I(N) := \max_{0 \leqslant j \leqslant N-1} I(N, 2j+1).$$

Lemma 3. *For $j \in \mathbb{Z}$, we have that $I(N, 2j+1) = I(N, 1)$, so that $I(N) = I(N, 1)$.*

Proof. $2j + 1 \in \mathbb{Z}_N^\times$, so $\{(2j+1)k \mod N : k = 0, ..., N-1\} = \{k \mod N : k = 0, ..., N-1\}$. Therefore

$$I(N, 2j+1) = \sum_{\substack{x = \frac{(2j+1)k}{N}, \\ 0 \leqslant k \leqslant N-1}} |\sin(x\pi)| = \sum_{\substack{x = \frac{k}{N}, \\ 0 \leqslant k \leqslant N-1}} |\sin(x\pi)| = I(N, 1).$$

Here, the central equality follows from the π-periodicity of $|\sin(\cdot)|$. □

Lemma 4. $\lim_{N\to\infty} \frac{1}{N} I(N) = \frac{2}{\pi}$*, and this limit is approached from below.*

Proof. We have that $\frac{1}{N} I(N) = \frac{1}{N} I(N, 1) = \frac{1}{N} \sum_{k=0}^{N-1} \left|\sin\left(\frac{k\pi}{N}\right)\right|$, while using Riemann sums we get:

$$\int_0^\pi |\sin(x)| \, dx = \lim_{N\to\infty} \frac{\pi}{N} \sum_{k=0}^{N-1} \left| \sin\left(\frac{k\pi}{N}\right) \right| = \pi \lim_{N\to\infty} \frac{1}{N} I(N).$$

As the LHS is equal to 2, we get the claimed limit. Moreover, as $|\sin(\cdot)|$ is concave, we get this sequence of Riemann sums is increasing. □

Theorem 1. *Let $m \in \mathbb{R}^N$. Then $\|m\|_\infty^{\mathsf{can}} \leqslant \sqrt{I(N)^2 + 1}\, \|m\|_\infty$, and this bound is the best possible for fixed N.*

Proof. Fix $\|m\|_\infty \leqslant 1$, and consider optimising $\|m\|_\infty^{\mathsf{can}}$ subject to this constraint. Since we can rotate the entries of $\sigma(m) \in \mathbb{C}^{N/2}$ via performing automorphisms on m, we may consider without loss of generality maximising $|m(\zeta)|$, where $\zeta = \exp(\frac{\pi i}{N})$.

Let $M(X) = \sum_{k=0}^{N-1} X^j$. We will show that, if $m(X)$ achieves a maximum of $|m(\zeta)|$ with $\|m\|_\infty \leqslant 1$, then $m(X) = X^k M(X)$ for some $k \in \mathbb{Z}_{2N}$. This is sufficient to prove our result as $\left\|X^k M(X)\right\|_\infty^{\mathsf{can}} = \sqrt{I(N)^2 + 1}$.

Let us fix some maximising polynomial $m(X)$ with $\|m\|_\infty \leqslant 1$, and let $k \in \mathbb{Z}_{2N}$ be such that $k = \arg\max\limits_j \operatorname{Im}(\zeta^{-j} m(\zeta))$. If $m(X) = X^k M(X)$, we are done, as otherwise we will derive a contradiction to either the maximality of k or the maximality of m. By negating m as necessary, we say $\operatorname{Im}(\zeta^{-k} m(\zeta)) \geqslant 0$. In fact, we can say $\operatorname{Im}(\zeta^{-k} m(\zeta)) \geqslant 1$, since otherwise the maximality of m would give $\operatorname{Re}(\zeta^{-k} m(\zeta))^2 > \operatorname{Im}(\zeta^{-k} m(\zeta))^2$, contradicting the maximality of k via either $k + N/2$ or $k - N/2$.

The polynomial $M(X)$ (up to sign, uniquely) maximises the imaginary component $|\operatorname{Im}(M(\zeta))|$. We therefore have $\left|\operatorname{Im}(\zeta^{-k} m(\zeta))\right| \leqslant |\operatorname{Im}(M(\zeta))|$. Comparing $|m(\zeta)|$ and $|M(\zeta)|$ we find

$$\begin{aligned} |m(\zeta)|^2 - |M(\zeta)|^2 &= \left|\zeta^{-k} m(\zeta)\right|^2 - \operatorname{Im}(M(\zeta))^2 - \operatorname{Re}(M(\zeta))^2 \\ &= \underbrace{\operatorname{Im}(\zeta^{-k} m(\zeta))^2 - \operatorname{Im}(M(\zeta))^2}_{(1)} + \underbrace{\operatorname{Re}(\zeta^{-k} m(\zeta))^2 - 1}_{(2)}. \end{aligned}$$

As discussed, we certainly have (1) $\leqslant 0$. We will show that we also have (2) $\leqslant 0$, since otherwise k is not the maximal choice.

Suppose $\operatorname{Re}(\zeta^{-k} m(\zeta)) > 1$, so that $\sum_{j=0}^{N-1} m_j \cos\left(\frac{(j-k)\pi}{N}\right) > 1$. Then:

$$\begin{aligned} &\operatorname{Im}(\zeta^{-(k-1)} m(\zeta)) - \operatorname{Im}(\zeta^{-k} m(\zeta)) \\ &\qquad = \sum_{j=0}^{N-1} m_j \left(\sin\left(\frac{(j-k+1)\pi}{N}\right) - \sin\left(\frac{(j-k)\pi}{N}\right)\right) \\ &\qquad = \left(\cos\left(\frac{\pi}{N}\right) - 1\right) \operatorname{Im}(\zeta^{-k} m(\zeta)) + \sin\left(\frac{\pi}{N}\right) \operatorname{Re}(\zeta^{-k} m(\zeta)) \\ &\qquad > \cos\left(\frac{\pi}{N}\right) + \sin\left(\frac{\pi}{N}\right) - 1 \geqslant 0 \end{aligned}$$

since $N \geqslant 2$, contradicting the maximality of k. We can derive a similar contradiction using $\zeta^{-(k+1)} m(\zeta)$ in the case $\operatorname{Re}(\zeta^{-k} m(\zeta)) < -1$.

We have therefore shown $|m(\zeta)|^2 \leqslant |M(\zeta)|^2$, with equality if and only if there exists a $k \in \mathbb{Z}_{2N}$ with $m(X) = X^k M(X)$. The maximum for $|m(\zeta)|$ is therefore given by $|M(\zeta)| = \sqrt{I(N)^2 + 1}$ as claimed. □

Corollary 1. *Suppose for all $m \in \mathbb{R}^N$ we have $\|m\|_\infty^{\mathsf{can}} \leqslant N \cdot M(N) \|m\|_\infty$ with $M(N)$ a least upper bound. Then $M(N) \to \frac{2}{\pi}$ as $N \to \infty$.*

Proof. Immediate from Lemma 4 and Theorem 1. □

We now bound just the real component of the canonical embedding of m, although the following results apply equally to the imaginary component. We use the notation $\|m\|_\infty^{\mathsf{can},\mathrm{Re}} = \max_j |\mathrm{Re}(m(\zeta_j))|$. We find that, in the limit, the upper bound on expansion of just the real component equals the upper bound on the entire expansion.

Lemma 5. *Let $m \in \mathbb{R}^N$. Then $\|m\|_\infty^{\mathsf{can},\mathrm{Re}} \leqslant I(N)\|m\|_\infty$, and this result is best possible.*

Proof. This proof technique is again adapted from [5]. Consider the j^{th} component of the canonical embedding z_j, given by evaluating $m(X)$ at the j^{th} primitive $2N^{\mathrm{th}}$ root of unity. Then the real part of z_j is given by:

$$\mathrm{Re}(z_j) = \sum_{k=0}^{N-1} m_k \mathrm{Re}(\zeta_j^k).$$

We maximise the magnitude of this quantity, subject to $\|m\|_\infty = B$, by setting each $m_k = B \cdot \mathrm{Sign}(\mathrm{Re}(\zeta_j^k))$. We therefore have:

$$\begin{aligned}\|m\|_\infty^{\mathsf{can},\mathrm{Re}} &\leqslant B \max_{j=0,\dots,N-1} \sum_{k=0}^{N-1} \left|\cos\left(\frac{k(2j+1)\pi}{N}\right)\right| \\ &= I(N)\|m\|_\infty .\end{aligned}$$

□

Corollary 2. *Let $m \in \mathbb{R}^N$. If for all N we have that $\|m\|_\infty^{\mathsf{can},\mathrm{Re}} \leqslant kN\|m\|_\infty$ then $k \leqslant \frac{2}{\pi}$ and $k \to \frac{2}{\pi}$ as $N \to \infty$.*

Proof. Immediate from Lemma 4 and Lemma 5. □

3.2 Application to Encoding

In this section, we apply the results from Section 3.1 to produce bounds on the growth of polynomials under encoding and decoding. Our first result enables us to produce bounds in the plaintext space given bounds in the message space.

Lemma 6. *Suppose $m \in \mathbb{R}^N$ and $\mathbf{z} \in \mathbb{C}^{N/2}$ are such that $m = \mathrm{Encode}(\mathbf{z}, \Delta)$. Then $\|m\|_\infty \leqslant \Delta\|\mathbf{z}\|_\infty + \frac{1}{2}$.*

Proof. Immediate from Lemma 1 and that encoding rounds coefficient-wise.

□

The result in Theorem 1 enables us to give bounds in the message space given bounds in the plaintext space.

Lemma 7. *Suppose $m \in \mathbb{R}^N$ has $\|m\|_\infty \leqslant B$. If $\mathbf{z} = \mathrm{Decode}(m, \Delta)$ we have that $\|\mathbf{z}\|_\infty \leqslant \frac{\sqrt{I(N)^2+1}}{\Delta}B$, and this bound is the best possible.*

Due to the fast convergence of $I(N)$ to $\frac{2N}{\pi}$, we can replace this result by its limiting value. We can therefore precisely bound the error introduced during encoding.

Corollary 3. *Suppose* $\mathbf{z} \in \mathbb{C}^{N/2}$ *is encoded under scale factor* Δ. *Then the precision lost in each slot as a result of encoding is bounded by* $\frac{\sqrt{I(N)^2+1}}{2\Delta}$, *and this bound tends to* $\frac{N}{\pi\Delta}$ *as* $N \to \infty$.

Proof. Immediate from Lemma 7 and the fact that the encoding error polynomial has coefficients in $\left[-\frac{1}{2}, \frac{1}{2}\right]$. □

We can also give analogous results for the real and imaginary components alone.

Lemma 8. *Suppose* $m \in \mathbb{R}^N$ *has* $\|m\|_\infty \leqslant B$. *Then if* $\mathbf{z} = \mathrm{Decode}(m, \Delta)$ *we have that* $\|Re(\mathbf{z})\|_\infty, \|Im(\mathbf{z})\|_\infty \leqslant \frac{2N}{\pi\Delta}B$, *and this bound is the best possible.*

Corollary 4. *Suppose* $\mathbf{z} \in \mathbb{C}^{N/2}$ *is encoded under scale factor* Δ. *Then the precision lost on both the real and imaginary components of each slot is bounded by* $\frac{N}{\pi\Delta}$.

This shows that, perhaps surprisingly, if using a worst case analysis, it is not possible to achieve a tighter analysis of precision loss by considering only the error on the real part of the message. To benefit from restricting our attention to only the real part, we must be able to specify statistical, rather than worst case, behaviour.

4 Noise analysis methods

In this section, we present the three noise analysis methods considered in this work, and apply them to give noise heuristics for the Textbook CKKS scheme [10]. We first introduce some notation and definitions.

Noise definitions and notation: For a Textbook CKKS ciphertext $(\mathtt{ct}_0, \mathtt{ct}_1)$ at level ℓ encrypting a message m, we define its noise as the polynomial ϵ such that $\langle \mathtt{ct}, \mathtt{sk} \rangle = m + \epsilon \mod Q_\ell$ where this noise ϵ is small. We denote by ρ^2 the (component) variance of a noise polynomial ϵ. Some operations, such as key switching, introduce an additive noise term, whose variance we denote by η^2. We treat noise polynomials as continuous random variables for simplicity, but the distributional results are applicable to the corresponding discrete random variables for practical purposes.

Variance: We will use the following variance results. A polynomial f with coefficients distributed uniformly in $[-k/2, k/2]$, has coefficient variance $\rho_f^2 = k^2/12$. A polynomial sampled from $ZO(\rho)$ has coefficient variance ρ. A polynomial sampled from the Ring-LWE error distribution has coefficient variance σ^2.

4.1 Bounding noise random variables

In this subsection, we introduce our refinement for bounding a random variable of a given variance. Given a (multivariate) random variable, we wish to identify a reasonable upper bound on the size of the components of the random variable(s). It has been common practice [13, 14, 20] to give an upper bound using the fact that $\texttt{erfc}(6) \approx 2^{-55}$. Instead of deferring to such a bound in all contexts, we express our bounds on distributions in terms of a new failure probability parameter α, defined as follows.

Definition 2. *Suppose a random variable Z has real support. We will say B* is a probability $1-\alpha$ bound *on Z if* $\Pr(Z > B) = \alpha$. *Equivalently, we will say B has* failure probability α*, or that B has* error tolerance α.

This refinement enables us to determine bounds on a random variable that hold with probability $(1-\alpha)$. In Theorem 2, we give bounds both for the canonical embedding as in prior CKKS analyses (e.g. [10]), and for the plaintext ring. When applying Theorem 2 in real and complex settings respectively, we use the following functions for notational convenience:

$$H_{\mathbb{R}}(\alpha, N) := \texttt{erf}^{-1}((1-\alpha)^{\frac{1}{N}}) \text{ and } H_{\mathbb{C}}(\alpha, N) := (-\ln(1-(1-\alpha)^{\frac{2}{N}}))^{\frac{1}{2}}\,.$$

For example, a ciphertext with noise variance ρ^2 can be bounded in the canonical embedding as $\sqrt{N} \cdot \rho \cdot H_{\mathbb{C}}(\alpha, N)$ using Theorem 2 part(3).

Theorem 2. *Suppose $Z \sim \mathrm{N}(\mathbf{0}, \rho^2 I_N)$. Then:*

1. *A probability $(1-\alpha)$ bound on the random variable $\|Z\|_\infty$ is given by*
$$B = \sqrt{2}\ \rho\ \texttt{erf}^{-1}((1-\alpha)^{\frac{1}{N}}).$$
2. *Let τ denote the map used in encoding and decoding and consider $\tau(Z)$. Then we have that $Re(\tau(Z)), Im(\tau(Z)) \sim \mathrm{N}(\mathbf{0}, \frac{N}{2}\rho^2 I_{N/2})$, and a probability $(1-\alpha)$ bound on both $\|Re(\tau(Z))\|_\infty$ and $\|Im(\tau(Z))\|_\infty$ is given by*
$$B = \sqrt{N}\ \rho\ \ \texttt{erf}^{-1}((1-\alpha)^{\frac{2}{N}}).$$
3. *A probability $(1-\alpha)$ bound on the random variable $\|Z\|_\infty^{\mathsf{can}}$ is given by*
$$B = \sqrt{N}\rho\ (-\ln(1-(1-\alpha)^{\frac{2}{N}}))^{\frac{1}{2}}.$$

Proof. A proof is given in Supplementary Material Section A. □

4.2 Worst-case noise analysis methods

In this subsection, we introduce the two worst-case noise analysis methods considered in this work.

Worst-case canonical embedding analysis: The first noise analysis method we consider is a refinement of the standard approach for analysis of CKKS noise, as e.g. in [10]. This method tracks bounds on the noise polynomials under the canonical embedding, $\|\epsilon\|_\infty^{\mathsf{can}}$, i.e. the bounds are presented in the message space. We improve on the canonical embedding bounds in [10] by following the Iliashenko approach [26]. For a noise polynomial that consists of several summands, this approach calculates the coefficient variance of the whole sum and then maps under the canonical embedding to obtain a bound on the noise. In contrast, the prior approach relies on repeated applications of the triangle inequality to bound individual summands that are then combined into a final bound. The Iliashenko approach is expected to provide tighter bounds than the prior approach [14]. We use Theorem 2 or triangle inequalities to derive bounds in the canonical embedding. We use the fact that $\|p(X)q(X)\|_\infty^{\mathsf{can}} \leqslant \|p(X)\|_\infty^{\mathsf{can}} \|q(X)\|_\infty^{\mathsf{can}}$ is the worst case bound on a product of polynomials.

Worst-case analysis in the ring: In this method, like the other worst-case analyses, we track how a bound $\|\epsilon\|_\infty$ on the size of the largest coefficient of the noise polynomial grows with each homomorphic operation. The difference is that we give the bound 'in the ring', i.e. the bounds are presented in the plaintext space, into which decryption takes place. We again use triangle inequalities and Theorem 2 to derive bounds in the ring. We use the fact that $\|p(X)q(X)\|_\infty \leqslant N\|p(X)\|_\infty \|q(X)\|_\infty$ is the worst case bound on a product. We note that this noise analysis method has has been considered for other homomorphic encryption schemes, as e.g. in [32], and this is the first work that considers it for CKKS.

4.3 Average-case noise analysis method

We present the main result of this section, Theorem 3, that considers the product of two Normally distributed polynomials. We then show how Theorem 3 enables us to develop the first average-case noise analysis for CKKS using Heuristic **??**.

Theorem 3. *Suppose that $Z \sim \mathrm{N}(\boldsymbol{\mu}; \rho^2 I_N)$ and $Z' \sim \mathrm{N}(\boldsymbol{\mu}'; \rho'^2 I_N)$, then the polynomial product ZZ' (modulo X^N+1) has mean vector $\mathbf{E}(ZZ')$ and covariance matrix $\mathrm{Cov}(ZZ')$ given by*

$$\mathbf{E}(ZZ') = \boldsymbol{\mu}^* \quad \textit{and} \quad \mathrm{Cov}(ZZ') = \rho_*^2 I_N + S,$$

where $\boldsymbol{\mu}^*$ is the polynomial product of $\boldsymbol{\mu}$ and $\boldsymbol{\mu}'$, $\rho_*^2 = N\rho^2\rho'^2 + \rho'^2\|\boldsymbol{\mu}\|_2^2 + \rho^2\|\boldsymbol{\mu}'\|_2^2$ and S is an off-diagonal matrix with entries

$$S_{i,i'} = \rho'^2 \sum_{j=0}^{N-1} \xi(i-j)\xi(i'-j)\mu_{i-j}\mu_{i'-j} + \rho^2 \sum_{j=0}^{N-1} \xi(i-j)\xi(i'-j)\mu'_{i-j}\mu'_{i'-j},$$

for a modified sign function ξ given by $\xi(z) = \mathit{Sign}(z)$ for $z \neq 0$ and $\xi(0) = 1$. Furthermore, the components $(ZZ')_i$ of this polynomial product can be approximated as a Normal $\mathrm{N}(\mu_i^*, \rho_*^2)$ distribution.

Proof. Let $Y = ZZ'$, so Y has components $Y_i = \sum_{j=0}^{N-1} \xi(i-j) Z_{i-j} Z'_j$. The mean of such a component Y_i is given by

$$\mathbf{E}(Y_i) = \textstyle\sum_{j=0}^{N-1} \mathbf{E}(\xi(i-j) Z_{i-j} Z'_j) = \sum_{j=0}^{N-1} \xi(i-j) \mu_{i-j} \mu'_j = \mu^*_i,$$

so ZZ' has mean vector $\mathbf{E}(ZZ') = \mathbf{E}(Y) = \boldsymbol{\mu}^*$. The summands of a component Y_i are independent, so the variance of Y_i is given by

$$\begin{aligned} \mathrm{Var}(Y_i) &= \textstyle\sum_{j=0}^{N-1} \mathrm{Var}\left(Z_{i-j} Z'_j\right) = \sum_{j=0}^{N-1} \mathbf{E}\left(Z_{i-j}^2\right) \mathbf{E}\left(Z'^2_j\right) - \mathbf{E}\left(Z_{i-j}\right)^2 \mathbf{E}\left(Z'_j\right)^2 \\ &= \textstyle\sum_{j=0}^{N-1} \left(\rho^2 + \mu_{i-j}^2\right)\left(\rho'^2 + \mu'^2_j\right) - \mu_{i-j}^2 \mu'^2_j \\ &= \textstyle\sum_{j=0}^{N-1} \rho^2 \rho'^2 + \rho^2 \mu'^2_j + \rho'^2 \mu_{i-j}^2 = N\rho^2\rho'^2 + \rho'^2 \left\|\boldsymbol{\mu}\right\|_2^2 + \rho^2 \left\|\boldsymbol{\mu}'\right\|_2^2 . \end{aligned}$$

A similar argument shows that the covariance of distinct components Y_i and $Y_{i'}$ (so $i \neq i'$) is given by

$$\mathrm{Cov}(Y_i, Y_{i'}) = \rho'^2 \sum_{j=0}^{N-1} \xi(i-j)\xi(i'-j)\mu_{i-j}\mu_{i'-j} + \rho^2 \sum_{j=0}^{N-1} \xi(i-j)\xi(i'-j)\mu'_{i-j}\mu'_{i'-j}.$$

Thus $Y = ZZ'$ has covariance matrix $\rho_*^2 I_N + S$. The distribution of $Y_i = (ZZ')_i$ can be addressed by considering the related sum

$$\begin{aligned} Y'_i &= \textstyle\sum_{j=0}^{N-1} \xi(i-j)(Z_{i-j} - \mu_{i-j})(Z'_j - \mu'_j) \\ &= Y_i - \textstyle\sum_{j=0}^{N-1} \xi(i-j)\left(\mu_{i-j} Z'_j + \mu'_j Z_{i-j} - \mu_{i-j}\mu'_j\right) . \end{aligned}$$

These summands are a product of $Z_{i-j} - \mu_{i-j} \sim \mathrm{N}(0, \rho^2)$ and $Z'_j - \mu'_j \sim \mathrm{N}(0, \rho'^2)$, so have mean 0 and variance $\rho^2\rho'^2$. Furthermore, the summands of Y'_i are independent and identically distributed, so a Central Limit argument shows that the distribution of Y'_i is well-approximated by a Normal distribution for large N. However, Y_i differs from Y'_i by a Normal random variable so $Y_i = (ZZ')_i$ is well-approximated by a Normal $\mathrm{N}(\mu^*_i, \rho^2_*)$ distribution. □

Theorem 3 gives the mean and covariance of the product $Y = ZZ'$, and shows the components Y_i of Y can be well-approximated as Normal. Our average-case analysis will model ZZ' as a multivariate Normal distribution of the established mean and covariance. This is expressed in Heuristic **??** and will be justified below.

Heuristic 1 *Suppose that $Z \sim \mathrm{N}(\boldsymbol{\mu}; \rho^2 I_N)$ and $Z' \sim \mathrm{N}(\boldsymbol{\mu}'; \rho'^2 I_N)$. Then, for $\boldsymbol{\mu}^*$, ρ^2_* and S as specified in Theorem 3, the polynomial product ZZ' (modulo $X^N + 1$) can be approximated as a multivariate Normal distribution as*

$$ZZ' \sim \mathrm{N}\left(\boldsymbol{\mu}^*;\ \rho^2_* I_N + S\right).$$

Small-S assumption: To simplify our analysis, we make the assumption that the off-diagonal matrix S encountered in Theorem 3 is negligible. While we believe this assumption is reasonable in many circumstances of interest, we note that it would not hold e.g. if the mean vectors have large constant components.

Definition 3. *A covariance matrix of the form $\rho^2_* I_N + S$ with constant component covariance ρ^2_* and off-diagonal matrix S satisfies the* Small-S assumption *if this off-diagonal matrix S is negligible compared to $\rho^2_* I_N$.*

Average-case noise analysis: In an average-case noise analysis, we track the how the variance of the noise polynomial ϵ develops with each homomorphic operation, rather than tracking how a bound on the coefficients of ϵ develops. In our average-case noise analysis of CKKS, we consider how the variance of ϵ develops 'in the ring', i.e., in the plaintext space. Heuristic **??** shows that, in the ring, the polynomial product (modulo $X^N + 1$) of multivariate Normal vectors can be well-approximated as a multivariate Normal distribution. Moreover, under the Small-S assumption, we can model the input and output polynomials in an application of Heuristic **??** as Normal random variables of a specified component variance. This enables us to track the variance of the noise polynomial through each homomorphic operation using the results presented below in Corollary 5. Given the variance in an output ciphertext, we can then use Theorem 2 to derive a bound on the noise in the output ciphertext.

Corollary 5. *Suppose that $Z \sim \mathrm{N}(\boldsymbol{\mu}; \rho^2 I_N)$ and $Z' \sim \mathrm{N}(\boldsymbol{\mu}'; \rho'^2 I_N)$ are independent, λ is a constant vector. Approximations to the distribution of $Z + Z'$, λZ and the rounding $\lfloor Z \rceil$ are then given by:*

$$Z + Z' \sim \mathrm{N}(\mathbf{0}, (\rho^2 + \rho'^2) I_N), \lambda Z \sim \mathrm{N}\left(\lambda \boldsymbol{\mu}\ ;\ \rho^2 \left\|\lambda\right\|_2^2 I_N\right), \lfloor Z \rceil \sim \mathrm{N}\left(\boldsymbol{\mu}\ ,\ \rho^2 + \tfrac{1}{12}\ \right).$$

Furthermore, an approximation to the distribution of ZZ' when the Small-S assumption is valid for ZZ' and an approximation to the distribution of Z^2 when the Small-S assumption is valid for Z^2 are given by:

$$ZZ' \sim \mathrm{N}\left(\boldsymbol{\mu}\boldsymbol{\mu}'\ ;\ (N\rho^2\rho'^2 + \rho'^2 \left\|\boldsymbol{\mu}\right\|_2^2 + \rho^2 \left\|\boldsymbol{\mu}'\right\|_2^2) I_N\right)$$
$$\text{and } Z^2 \sim \mathrm{N}\left(\boldsymbol{\mu}^2\ ;\ 2\rho^2(N\rho^2 + 2\left\|\boldsymbol{\mu}\right\|_2^2) I_N\right).$$

4.4 Summary of textbook noise heuristics

In this subsection, we summarise the noise heuristics obtained when analysing the Textbook CKKS scheme [10] according to the three different noise analysis methods presented in this work. Table 1 gives the worst-case analyses in the ring (WCR) and in the canonical embedding (CE) for Textbook CKKS [10]. Table 2 gives the average-case analysis in terms of the variance of the noise after each homomorphic operation, and illustrates how this variance could be converted to a bound on the noise in the output ciphertext using Theorem 2.

The full justification for the distributional results leading to the noise heuristics in Tables 1 and 2 is given in Supplementary Material Section B. This gives a variance for the noise polynomial after each operation, directly giving the average-case analysis. The variances can then be converted to a bound in either the canonical embedding or the ring after each operation to give the respective worst-case analyses, using Theorem 2. This is illustrated in Supplementary Material Section C.

The worst-case bounds developed in this work can be contrasted with the worst-case canonical embedding bounds given in prior work (as e.g. in [10]). These are restated in Supplementary Material Section D.

Operation	WCR	CE
Fresh	$\sigma\sqrt{N+2h+2}\cdot H_{\mathbb{R}}(\alpha, N)$	$\sigma\sqrt{\frac{N^2}{2}+hN+N}\cdot H_{\mathbb{C}}(\alpha, N)$
Add	B_1+B_2	B_1+B_2
PreMult	$N\cdot\left(\lVert m_1\rVert_\infty B_2+\lVert m_2\rVert_\infty B_1+B_1B_2\right)$	$\lVert m_1\rVert_\infty^{\mathsf{can}} B_2+\lVert m_2\rVert_\infty^{\mathsf{can}} B_1+B_1B_2$
Key-Switch	$B+\sqrt{2}\cdot\eta_{\mathsf{ks}}\cdot H_{\mathbb{R}}(\alpha, N)$	$B+\sqrt{N}\cdot\eta_{\mathsf{ks}}\cdot H_{\mathbb{C}}(\alpha, N)$
Rescale	$\Delta^{-1}B+\sqrt{\frac{1}{6}(h+1)}\ H_{\mathbb{R}}(\alpha, N)$	$\Delta^{-1}B+\sqrt{\frac{N}{12}(h+1)}H_{\mathbb{C}}(\alpha, N)$

Table 1: Worst-case bounds for Textbook CKKS [10]. Here, B_1, B_2, and B denote input noise bounds in the ring or canonical embedding, as appropriate, and $\eta_{\mathsf{ks}}=\sqrt{\frac{1}{12}\left(P^{-2}NQ_\ell^2\sigma^2+\mathbb{1}_{P\nmid Q_\ell}(h+1)\right)}$.

Operation	Output Variance	Final output bound (CE)
Fresh	$\rho^2_{\mathtt{fresh}}=(\frac{N}{2}+h+1)\sigma^2$	$\sqrt{N}\cdot\rho_{\mathtt{fresh}}\cdot H_{\mathbb{C}}(\alpha, N)$
Add	$\rho^2_{\mathsf{add}}=\rho_1^2+\rho_2^2$	$\sqrt{N}\cdot\rho_{\mathsf{add}}\cdot H_{\mathbb{C}}(\alpha, N)$
PreMult	$\rho^2_{\mathsf{pre\text{-}mult}}=N\rho_1^2\rho_2^2+\rho_2^2\lVert m_1\rVert_2^2+\rho_1^2\lVert m_2\rVert_2^2$	$\sqrt{N}\cdot\rho_{\mathsf{pre\text{-}mult}}\cdot H_{\mathbb{C}}(\alpha, N)$
Key-Switch	$\rho^2_{\mathtt{ks}}=\rho^2+\frac{1}{12}\left(P^{-2}NQ_\ell^2\sigma^2+\mathbb{1}_{P\nmid Q_\ell}(h+1)\right)$	$\sqrt{N}\cdot\rho_{\mathtt{ks}}\cdot H_{\mathbb{C}}(\alpha, N)$
Rescale	$\rho^2_{\mathtt{rs}}=\frac{\rho^2}{\Delta^2}+\frac{1}{12}(h+1)$	$\sqrt{N}\cdot\rho_{\mathtt{rs}}\cdot H_{\mathbb{C}}(\alpha, N)$

Table 2: Average-case noise analysis for Textbook CKKS [10]. Here, ρ_1, ρ_2, and ρ denote input noise variances. The final output variance can be converted to (e.g.) a canonical embedding bound using Theorem 2.

5 Application of methods to RNS-CKKS

In this section we discuss the application of the noise analysis methods described in Section 4 to RNS variants of CKKS [8, 28]. We focus mainly on [8]. We also present, in Section 5.4, the application of the analyses to some of the optimisations presented in [28].

5.1 Differences from Textbook CKKS

The operations in RNS variants of CKKS are performed 'slotwise' with respect to the constituent moduli q_j making up the ℓ^{th} ciphertext modulus $Q_\ell=\prod_{j=0}^{\ell}q_j$. In [8], for all $0\le j\le L$, a distinct $q_j=1 \mod 2N$ is chosen to support NTT operations in each slot. It is also required that $\Delta/q_j\approx 1$ for all $1\le j\le L$ and q_0 is sufficiently large for correctness. The need for distinct q_j that are not exactly equal to Δ incurs an approximation error not present in Textbook CKKS.

The changes in parameters in the RNS variants require modifications to the rescale and key switch operations. The other operations carry over to the RNS case in a more straightforward way. When rescaling from Q_ℓ to $Q_{\ell-1}$ in RNS variants, instead of dividing by Δ, we divide by q_ℓ. The key switching procedure presented in [8] translates the key switching approach of [10] to the RNS setting and so requires the large modulus P to be formed from a set of k pairwise coprime p_i as $P = \prod_{i=0}^{k} p_i$. We also note that a hybrid key switching is possible in the RNS setting, for example as done in [28].

The definition of noise in the RNS variant [8] also differs from the Textbook CKKS definition. A RNS ciphertext $\mathtt{ct}$ at level ℓ can be expressed as a vector of its RNS representatives $(\mathtt{ct}^{(j)})_{0 \leq j \leq \ell}$. The noise in a ciphertext is defined in [8] as ϵ such that $\left\langle \mathtt{ct}^{(0)}, \mathtt{sk} \right\rangle = m + \epsilon \mod q_0$.

5.2 Distribution of noise polynomials for the RNS variant [8]

In this subsection we derive the distributions of the noise polynomials for the RNS variant [8] that differ from Textbook CKKS, namely, for the rescale and key switch operations. The analysis for the other operations is analogous to the Textbook CKKS case as presented in Section 4.4. Proofs for the results in this subsection are presented in Supplementary Material Section E.

Lemma 9. **[Key Switch – RNS]** *The* RNS-CKKS `Key Switch` *operation applied to a ciphertext at level ℓ introduces an additive error such that the output noise is given by $\epsilon_{ks} := \epsilon + \varepsilon_{\mathtt{ks}}$ if the input noise is given by ϵ. The additive error $\varepsilon_{\mathtt{ks}}$ has a Normal distribution given by*

$$\varepsilon_{ks} \sim \mathrm{N}(\mathbf{0}, \eta_{ks}^2 I_N), \quad \textit{where } \eta_{ks}^2 = \tfrac{1}{12} P^{-2} N Q^2 (\ell^2 + 1)\sigma^2 + \tfrac{1}{12}(k^2+1)(\|s\|_2^2 + 1)\,.$$

For example, if the secret is sparse with fixed Hamming weight h, we have $\eta_{\mathsf{ks}}^2 = \frac{1}{12} P^{-2} N Q^2 (\ell^2 + 1)\sigma^2 + \frac{1}{12}(k^2+1)(h+1)$.

Lemma 10. **[Rescale – RNS]** *Let* $\mathtt{ct}$ *be a ciphertext encrypting m with noise ϵ. Let* $\mathtt{ct}_{\mathtt{rs}}$ *encrypting m be the ciphertext with noise $\epsilon_{\mathtt{rs}}$ resulting from the* `Rescale` *operation. The* `Rescale` *noise $\epsilon_{\mathtt{rs}} \sim N(\mathbf{0}; \rho_{\mathtt{rs}}^2 I_N)$, where the component variance $\rho_{\mathtt{rs}}^2$ is given by*

$$\rho_{\mathtt{rs}}^2 = \frac{\rho^2}{q_\ell^2} + \left(\tfrac{1}{12}(\|s\|_2^2 + 1)\right).$$

For example, if the secret is sparse with fixed Hamming weight h, we have $\rho_{\mathtt{rs}}^2 = \frac{\rho^2}{q_\ell^2} + \frac{1}{12}(h+1)$.

5.3 Summary tables of noise bounds

In this subsection, we present noise heuristics for the RNS variant [8] that were developed by applying the noise analyses of Section 4 to this variant. Table 3 summarises the worst-case canonical embedding and average-case noise heuristics for the RNS variant [8]. These heuristics can be justified in the same manner as explained in Section 4.4 for the Textbook CKKS case, using the distributional results presented in Section 5.2.

Operation	CE	Output variance (CLT)
Fresh	$\sigma\sqrt{\frac{N^2}{2}+hN+N}\cdot H_{\mathbb{C}}(\alpha, N)$	$\rho^2_{\mathtt{fresh}} = (\frac{N}{2}+h+1)\sigma^2$
Add	$B_1 + B_2$	$\rho^2_{\mathsf{add}} = \rho_1^2 + \rho_2^2$
PreMult	$\lVert m_1 \rVert_\infty^{\mathsf{can}} B_2 + \lVert m_2 \rVert_\infty^{\mathsf{can}} B_1 + B_1 B_2$	$\rho^2_{\mathsf{pre\text{-}mult}} = N\rho_1^2\rho_2^2 + \rho_2^2 \lVert m_1 \rVert_2^2 + \rho_1^2 \lVert m_2 \rVert_2^2$
Key-Switch	$B + \sqrt{N}\ \eta_{\mathsf{ks}}\ H_{\mathbb{C}}(\alpha, N)$	$\rho^2_{\mathsf{ks}} = \rho^2 + \eta^2_{\mathsf{ks}}$
Rescale	$q_\ell^{-1} B + \sqrt{\frac{N}{12}(h+1)}\ H_{\mathbb{C}}(\alpha, N)$	$\rho^2_{\mathtt{rs}} = \frac{\rho^2}{q_\ell^2} + \frac{1}{12}(h+1)$

Table 3: Worst-case canonical embedding bounds (CE) and average-case noise analysis (CLT) for RNS CKKS [8]. Here, B_1 and B_2 denote input noise bounds in the canonical embedding; ρ_1, ρ_2, and ρ denote input noise variances; and $\eta^2_{\mathsf{ks}} = \frac{1}{12}P^{-2}NQ^2(\ell^2+1)\sigma^2 + \frac{1}{12}(k^2+1)(h+1)$.

5.4 Other RNS optimisations

In this work, we mainly focus on Textbook CKKS [10] and the RNS CKKS variant [8]. Several optimisations for RNS-CKKS have been proposed, e.g. in [28]. Our noise analysis techniques can also be applied to other RNS variants. In this section, we illustrate this with some examples. Proofs for the results in this subsection are provided in Supplementary Material Section F.

Level-specific scaling factor: As a first example, in [28] it is proposed to use a level-specific scaling factor $\varDelta_\ell$ to remove the approximation error in rescaling arising from the fact that each of the q_i is not exactly equal to $\varDelta$. Having level-specific scaling factors $\varDelta_\ell$ means that we may wish to add or multiply two ciphertexts $\mathtt{ct}_1$ and $\mathtt{ct}_2$ at different scales $\varDelta_{\ell_1}$ and $\varDelta_{\ell_2}$. If WLOG $\ell_1 > \ell_2$ then the `Adjust` operation is used to bring the ciphertext $\mathtt{ct}_1$ to level ℓ_2 before the addition or multiplication is performed.

For simplicity, we give the analysis when adjusting by one level. This operation takes as input a ciphertext $\mathtt{ct} := (c_0, c_1)$ with respect to level $\ell+1$ and the output ciphertext is with respect to level ℓ. By definition of the chain of moduli, $Q_{\ell+1} = Q_\ell \cdot q_{\ell+1}$. Let

$$\mathtt{ct}_1 := \left(\left[\left\lfloor \frac{\varDelta_\ell \cdot q_{\ell+1}}{\varDelta_{\ell+1}} c_0 \right\rceil \right]_{Q_{\ell+1}}, \left[\left\lfloor \frac{\varDelta_\ell \cdot q_{\ell+1}}{\varDelta_{\ell+1}} c_1 \right\rceil \right]_{Q_{\ell+1}} \right)$$

The output ciphertext is then $\mathtt{ct}_{\mathrm{adj}} = \mathtt{Rescale}(\mathtt{ct}_1, q_{\ell+1})$. That is,

$$\mathtt{ct}_{\mathrm{adj}} = \left(\left[\left\lfloor \frac{1}{q_{\ell+1}} \left[\left\lfloor \frac{\varDelta_\ell \cdot q_{\ell+1}}{\varDelta_{\ell+1}} c_0 \right\rceil \right]_{Q_{\ell+1}} \right\rceil \right]_{Q_\ell}, \left[\left\lfloor \frac{1}{q_{\ell+1}} \left[\left\lfloor \frac{\varDelta_\ell \cdot q_{\ell+1}}{\varDelta_{\ell+1}} c_1 \right\rceil \right]_{Q_{\ell+1}} \right\rceil \right]_{Q_\ell} \right).$$

Lemma 11. [Adjust] *Let $\mathtt{ct} = (c_0, c_1)$ be a ciphertext encrypting m with noise $\epsilon \sim N(\mathbf{0}; \rho^2 I_N)$. Let $\mathtt{ct}_{\mathtt{adj}}$ encrypting m be the ciphertext with noise $\epsilon_{\mathtt{adj}}$ resulting* *from the* `Adjust` *by one level operation. The* `Adjust` *noise* $\epsilon_{\mathtt{adj}} \sim N(\mathbf{0}; \rho^2_{\mathtt{adj}} I_N)$*, where the component variance* $\rho^2_{\mathtt{adj}}$ *of the noise after the* `Adjust` *operation is*

$$\rho^2_{\mathtt{adj}} = \frac{\Delta_\ell^2}{\Delta_{\ell+1}^2}\rho^2 + \tfrac{1}{12}(\|s\|_2^2 + 1).$$

Rescale before multiplication: As a second example, in [28] it is proposed to reorder multiplication and rescaling. This enables the reduction of the noise in a ciphertext after multiplication. We give the analysis for the case of level-specific Δ_ℓ. In this case, we have:

$$\mathtt{ct}_{\text{Pre-Mult}'} = \mathtt{Pre\text{-}Mult}(\mathtt{Rescale}(\mathtt{ct}_1, q_\ell), \mathtt{Rescale}(\mathtt{ct}_2, q_\ell))\,.$$

Lemma 12. [Pre-Mult′] *Let* $\mathtt{ct}_1$ *encrypt* $\Delta_\ell^2 \cdot m_1$ *with noise* $\epsilon_1 \sim N(\mathbf{0}, \rho_1^2)$ *and let* $\mathtt{ct}_2$ *encrypt* $\Delta_\ell^2 \cdot m_2$ *with noise* $\epsilon_2 \sim N(\mathbf{0}, \rho_2^2)$*. Then* $\mathtt{ct}_{\textit{Pre-Mult}'}$ *encrypts* $\Delta_{\ell-1}^2 \cdot m_1 m_2$ *and noise* $\epsilon_{\textit{Pre-Mult}'}$*. If the conditions of Lemma 17 hold then* $\epsilon_{\textit{Pre-Mult}'} \sim N(\mathbf{0}, \rho^2_{\textit{Pre-Mult}'})$ *where*

$$\begin{aligned}\rho^2_{\textit{Pre-Mult}'} := \frac{N\rho_1^2\rho_2^2}{q_\ell^4} + \left(\frac{N^2}{18q_\ell^2} + \frac{N}{12q_\ell}\right)(\rho_1^2 + \rho_2^2) + \frac{\Delta_{\ell-1}}{q_\ell^2}\left(\rho_2^2\|m_1\|_2^2 + \rho_1^2\|m_2\|_2^2\right) \\ + \left(\frac{\Delta_{\ell-1}N}{18} + \frac{\Delta_{\ell-1}}{12}\right)\left(\|m_1\|_2^2 + \|m_2\|_2^2\right) + \frac{N^3}{18} + \frac{N^2}{108} + \frac{N}{144}\,.\end{aligned}$$

6 Experimental results

In this section we evaluate the efficacy of the noise analyses developed in this work for Textbook CKKS and the RNS variant of [8] as compared with their implementations HEAAN [24] and FullRNS-HEAAN [22] respectively. We also compare the new heuristics with those obtained from a worst-case canonical embedding approach as in prior work (denoted as P-CE). The code used to generate our results is available at `https://github.com/bencrts/CKKS_noise`.

Experimental framework: We run experiments in the HEAAN v1.0 [24] and FullRNS-HEAAN [22] libraries. We note that neither the Textbook CKKS nor the RNS variant noise analysis is implementation-specific, and we chose the HEAAN library as it is the implementation most closely resembles the theoretical description of both variants of the scheme.

The LWE parameters (ring dimension N, ciphertext modulus q, error standard deviation σ, secret distribution S) were set as follows. Following [1], we used $(\log_2(N), \log_2(q)) \in \{(13, 109), (14, 219), (15, 443)\}$ in HEAAN v1.0. We used $(\log_2(N), \log_2(q)) \in \{(12, 100), (13, 100), (14, 220), (15, 420)\}$ in FullRNS-HEAAN. We used $\sigma = 3.2$ and the default secret distribution in both libraries. We set the error tolerance as $\alpha = 0.0001$ and the scale parameter as $\Delta = 2^{40}$.

For FullRNS-HEAAN the moduli chains are parameterised by L and k. The bit-size of the top-level modulus is generated by FullRNS-HEAAN as $60+(L-1)\cdot$

$\log_2(\varDelta)$. For $\log(N) \in \{13, 14, 15\}$, we choose L to allow for a top-level modulus which is close to the choices in HEAAN v1.0. For $\log(N) = 12$ we choose a modulus large enough to support one multiplication. We always set the default library selection of $k = L + 1$.

For both libraries we evaluate the following circuit, similarly to [14]. We generate fresh ciphertexts $\mathtt{ct}_0$, $\mathtt{ct}_1$ and $\mathtt{ct}_2$ and evaluate the circuit $\mathtt{ct}_2 * (\mathtt{ct}_1 + \mathtt{ct}_0)$, i.e. a homomorphic addition, followed by a (full) homomorphic multiplication. In each experiment, we iterate 1000 times and record the average, and maximum, observed noise. In Tables 4 and 5 we report the observed noises together with the noise predicted from the heuristics developed in this work: the average-case approach (CLT), and the worst-case heuristics (WCR and CE). We also compare with the noise predicted from the prior heuristics (P-CE). For the multiplication operation estimates we use worst-case message bounds, specifically $\varDelta$ for WCR, $N\varDelta^2$ for CLT, and $\frac{2N\varDelta}{\pi}$ for CE.

In Table 4 we report the experimental results for HEAAN v1.0 [24] in two settings. We first (in the rows marked as 'Ring') report the observed noise in the plaintext space. In these experiments, in each trial, we generate a random plaintext with coefficients in $[-\varDelta, \varDelta]$, evaluate the specified circuit, and measure noise in the ring after each operation. We also (in the rows marked as 'Real' and 'Complex') report the observed noise in the message space. In Table 5 we report the experimental results for FullRNS-HEAAN [22] in the message space.

For the HEAAN v1.0 [24] and FullRNS-HEAAN [22] experiments in the message space, in each trial, we generate a vector of random numbers, encode them, encrypt them, and homomorphically evaluate the circuit as described above. Then, we decrypt and decode and measure the precision loss. The rows marked as 'Real' correspond to generating numbers with real part and imaginary part both uniform in $[0, 1]$, encoding and decoding with scale factor $\varDelta$, and reporting only the real error on the computation. The rows marks as 'Complex' correspond to generating numbers with real part and imaginary part both uniform in $[0, 1]$ and reporting the magnitude of the largest error.

While in exact schemes, it is trivial to observe the noise, this is not so straightforward for CKKS. Our methodology was to generate three plaintexts m_1, m_2 and m_3, and to run the circuit both in the plaintext space and in the ciphertext space. In other words, the noise reported in Tables 4 and Table 5 is the result of

$$((m_1 + m_2) \cdot m_3) - \mathtt{Dec}((\mathtt{Enc}(m_1) + \mathtt{Enc}(m_2)) \cdot \mathtt{Enc}(m_3))\,.$$

Results: The plaintext space experiments of Table 4 illustrate that for Textbook CKKS [10], the average case noise approach (CLT) introduced in this work, and our refinements to the prior worst case canonical embedding approach (CE), both improve on the heuristics given in prior work (P-CE), in the sense of predicting a value closer to the observed noise. For CLT compared to P-CE, the heuristic-to-practical gap reduces from around 8 bits to less than 1 bit. However, the CLT approach slightly underestimates the maximal noise, and sometimes slightly underestimates the maximum noise (as illustrated in the column $\mathtt{gap}$). The WCR approach leads to a large heuristic-to-practical gap after multiplication, which we also observed in the Complex experiments (in the message space), so we omit it in the FullRNS-HEAAN experiments.

For the message space results in Table 4, the addition and multiplication results are similar for both the Real and Complex case. The CLT and CE approaches both underestimate the average and maximum noise by 3 to 7 bits. The WCR approach correctly bounds the noise: tightly for addition, but very loosely after multiplication.

The results in Table 5 illustrate that for the RNS variant of [8], the CLT approach and the CE approach both improve on the prior approach (P-CE), in the sense of predicting a value closer to the observed noise. For CLT compared to P-CE, the heuristic-to-practical gap typically reduces from around 6 bits to less than 1 bit. However, we again very frequently observe the CLT giving an underestimate. At $\log N = 14$, a jump is seen in the observed noise values, and in this case the prior approach P-CE gives a tight bound on the noise.

Discussion: Our results illustrate that, for the plaintext space for Textbook CKKS, and for the message space in the RNS variant of [8], both the average-case noise analysis introduced in this work and the refinement of the prior worst-case canonical embedding approach improve upon prior noise analyses in terms of modelling more closely the observed noise. Our work can thus represent an improved starting point for manual parameter selection compared to prior approaches.

However, some discrepancies can be seen between the observed results and the predictions from the heuristic analyses. For example, in the multiplication results in the ring in Table 4, the WCR respectively CE bounds seem to increase by 1 respectively 0.5 bits as $\log N$ increases by 1 bit, while the average observed noise shows a much slower growth. As another example, in the message space results in Table 4, the prior canonical embedding approach (P-CE) also leads to underestimates of the predicted noise. Moreover, in the multiplication results of Table 5, there seems to be a jump in the observed noise after $\log N = 14$, whereas the noise heuristics all grow more smoothly as $\log N$ grows. This means that, for larger $\log N$, the P-CE approach gives a correct and tight noise growth prediction, while for smaller $\log N$, the CLT and CE approaches give a closer prediction of the observed noise. This discussion suggests a fundamental issue with the modelling in all existing noise analysis approaches, including those prior to this work, suggesting that a refined theory of CKKS noise is needed.

One of the most crucial observations is that our heuristics underestimate the noise growth in many places (denoted by negative `gap` values in Tables 4 and 5). Similar underestimates have been observed in the literature before [15, **?**]. In more detail, the authors of [**?**] show that an average-case analysis of the BFV scheme that assumes independence of the coefficients of the noise leads to underestimates of the multiplication noise, and they develop a correcting function to account for this discrepancy. The authors of [15] compare experimental results of the BGV scheme as implemented in HElib to the theoretical bounds from the work of [29], and observe that the latter also underestimates the noise growth in practice. In contrast, the implementation-specific analysis of [15] is shown to

log(N)	log(q)	Average	Maximum	CLT	WCR	CE	P-CE	gap
Ring Addition noise.								
13	109	4.58	5.52	4.32	4.82	10.87	12.77	-1.20
14	219	4.63	5.39	4.35	4.85	11.40	13.27	-1.04
15	443	4.68	5.49	4.37	4.87	11.92	13.77	-1.12
Ring Multiplication noise.								
13	109	5.18	6.19	5.67	19.32	12.61	14.32	-0.52
14	219	5.21	6.04	5.70	20.35	13.13	14.82	-0.34
15	443	5.27	6.09	5.72	21.37	13.66	15.32	-0.37
Real Addition error.								
13	109	-25.37	-23.42	-29.70	-22.83	-29.13	-27.22	-6.28
14	219	-24.41	-22.55	-29.18	-21.80	-28.60	-26.72	-6.63
15	443	-23.35	-21.32	-28.65	-20.78	-28.08	-26.22	-7.33
Real Multiplication error.								
13	109	-25.07	-23.00	-28.35	-8.33	-27.39	-25.68	-5.35
14	219	-24.03	-21.77	-27.83	-6.30	-26.87	-25.18	-6.06
15	443	-23.03	-20.98	-27.30	-4.28	-26.34	-24.68	-6.32
Complex Addition error.								
13	109	-24.81	-23.12	-29.63	-22.83	-29.13	-27.22	-6.51
14	219	-23.81	-22.22	-29.10	-21.80	-28.60	-26.72	-6.88
15	443	-22.76	-21.17	-28.58	-20.78	-28.08	-26.22	-7.41
Complex Multiplication error.								
13	109	-24.45	-22.52	-28.28	-8.33	-27.39	-25.68	-5.76
14	219	-23.47	-21.53	-27.75	-6.30	-26.87	-25.18	-6.22
15	443	-22.41	-20.61	-27.23	-4.28	-26.34	-24.68	-6.62

Table 4: Average and maximum bits of noise observed in the ring and message space over 1000 trials in HEAAN compared with noise predicted by the CLT, WCR and CE noise analyses. The column `gap` denotes the difference between the predicted CLT noise value and the maximum experimental observation, with a negative value representing a heuristic underestimate.

very closely match the observed noise growth. Our heuristics are not specific to the implementations in HEAAN v1.0 [24] or Full-RNS HEAAN [22], and assumptions on which the heuristics rely may not hold for each implementation. For example, our experiments indicate that in HEAAN v1.0 [24] (though not in Full-RNS HEAAN [22]), the independence heuristic between coefficients of the noise polynomial fails at encryption. We believe that developing implementation-specific noise analyses for CKKS is an important direction for future work.

$\log(N)$	$\log(q)$	L	k	Average	Maximum	CLT	CE	P-CE	`gap`
Real Addition error.									
12	100	2	3	-24.38	-24.21	-24.25	-23.63	-18.89	-0.04
13	100	2	3	-23.16	-22.93	-23.23	-22.61	-17.89	-0.30
14	220	5	6	-22.07	-21.75	-22.21	-21.59	-16.89	-0.46
15	420	10	11	-21.00	-20.74	-21.19	-20.57	-15.89	-0.45
Real Multiplication error.									
12	100	2	3	-21.86	-21.80	-22.96	-21.62	-17.39	-1.16
13	100	2	3	-21.70	-21.41	-21.94	-20.61	-16.39	-0.53
14	220	5	6	-17.79	-17.67	-20.92	-19.59	-15.39	-3.25
15	420	10	11	-16.77	-16.73	-19.90	-18.57	-14.39	-3.17
Complex Addition error.									
12	100	2	3	-24.03	-23.78	-24.17	-23.63	-18.89	-0.39
13	100	2	3	-22.83	-22.42	-23.16	-22.61	-17.89	-0.74
14	220	5	6	-21.84	-21.52	-22.14	-21.59	-16.89	-0.62
15	420	10	11	-20.76	-20.57	-21.12	-20.57	-15.89	-0.55
Complex Multiplication error.									
12	100	2	3	-21.17	-21.08	-22.88	-21.62	-17.39	-1.80
13	100	2	3	-21.03	-20.95	-21.86	-20.61	-16.39	-0.91
14	220	5	6	-16.94	-16.82	-20.84	-19.59	-15.39	-4.02
15	420	10	11	-15.93	-15.90	-19.82	-18.57	-14.39	-3.92

Table 5: Average and maximum bits of noise observed in the message space over 1000 trials in FullRNS-HEAAN compared with noise predicted by the CLT, CE and P-CE noise analyses. The column `gap` denotes the difference between the predicted CLT noise value and the maximum experimental observation, with a negative value representing a heuristic underestimate.

7 Application to Li-Micciancio mitigations

A natural application of improved noise analyses for CKKS is in the implementation of more performant countermeasures to the CKKS key recovery attack of Li and Micciancio [33]. One possible countermeasure suggested in [33] is to make the CKKS scheme exact. In this section, we describe how to do so under certain conditions, and prove this modified version is sufficient to satisfy the security notion IND-CPA^D introduced in [33]. In particular, we rely on a correctability condition for the circuit being homomorphically evaluated, defined as follows.

Definition 4. (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.

In this definition, the bound B and the factor Δ' are properties of the function g itself, and not specific to any particular noise analysis method. Informally, correctability ensures that for set parameters, the evaluation noise of g and the desired result $g(\mathbf{z}_1, \ldots, \mathbf{z}_l)$ never interact. Observe that it should always be possible to select parameters for which g is correctable.

We note that, for real circuits, analogous definitions could be made using bounds on the real noise. We could instead define correctability in the ring, but we opt for a definition in the message space due to its easy interpretability in terms of the precision of input messages. Correcting in the ring rather than in the message space would lead to a scheme similar to BFV with the addition of a rescale procedure and an implicit plaintext modulus.

We now proceed to show how to define an exact version of CKKS in the case of correctable circuits, show it is exact in the sense of [33], and outline why our adapted version achieves the IND-CPA^D security.

Lemma 13. *Suppose $\mathbf{z} \in \mathbb{C}^{N/2}$ has $\|\mathbf{z}\|_\infty < B$, and let $\Delta' = 2^{\lceil \log B \rceil + 1}$. Then $\lceil \frac{1}{\Delta'}\mathbf{z} \rfloor = 0$.*

Proof. We have that for all i, $|z_i| < B \leqslant \frac{\Delta'}{2}$. We must therefore have that $\left|\frac{z_i}{\Delta'}\right| < \frac{1}{2}$, so that $\lceil \frac{z_i}{\Delta'} \rfloor = 0$, and as claimed $\lceil \frac{1}{\Delta'}\mathbf{z} \rfloor = 0$. □

We therefore introduce the following procedure, performed after decoding, which converts an approximate homomorphic evaluation of a correctable function to an exact one:

Definition 5. *Suppose we have a message $\mathbf{z} \in \mathbb{C}^{N/2}$. Then we define the algorithm* Correct : $\mathbb{C}^{N/2} \times \mathbb{R}^+ \to \mathbb{C}^{N/2}$ *via*

$$\text{Correct}(\mathbf{z}, \Delta') = \Delta' \left\lceil \frac{1}{\Delta'}\mathbf{z} \right\rfloor .$$

Definitions 3 and 4 allow us to derive a correct scheme in the following sense.

Lemma 14. *Fix parameters, and suppose g is a correctable circuit with correcting factor Δ'. Suppose $\mathbf{z} = g(\mathbf{z}_1, ..., \mathbf{z}_l) + \mathbf{e}$ is the result of the homomorphic evaluation of the circuit g on inputs $\mathbf{z}_1, ..., \mathbf{z}_l$. Then with all but negligible probability, we have* Correct$(\mathbf{z}, \Delta') = g(\mathbf{z}_1, ..., \mathbf{z}_l)$.

Proof.

$$\begin{aligned}\text{Correct}(\mathbf{z}, \Delta') = \Delta' \left\lceil \frac{1}{\Delta'}\mathbf{z} \right\rfloor &= \Delta' \left\lceil \frac{1}{\Delta'}(g(\mathbf{z}_1, ..., \mathbf{z}_l) + \mathbf{e}) \right\rfloor = g(\mathbf{z}_1, ..., \mathbf{z}_l) + \left\lceil \frac{1}{\Delta'}\mathbf{e} \right\rfloor \\ &= g(\mathbf{z}_1, ..., \mathbf{z}_l),\end{aligned}$$

as required, with the third equality following due to the correctability of g, and the final equality following with all but negligible probability from definition of Δ' and Lemma 13. □

If we therefore augment the decryption procedure by performing `Correct` after decoding, we have that the resulting scheme is exact, or correct, for correctable circuits g in the sense of [33]:

$$\Pr\begin{pmatrix}\mathtt{ct}_i \leftarrow \mathtt{Enc}_{\mathtt{pk}}(\mathbf{z}_i) \text{ for } 1 \leqslant i \leqslant l, \\ \quad \mathtt{Dec}_{\mathtt{sk}}(\mathtt{Eval}(g, (\mathtt{ct}_i)_{i=1}^l)) = g((m_i)_{i=1}^l)\end{pmatrix} = 1 - \operatorname{negl}(\kappa)\,,$$

where κ is the security parameter. Therefore, by [33, Lemma 1], we have that our corrected scheme is IND-CPA^D secure. In more detail, where the IND-CPA adversary is unable to provide decryptions to the IND-CPA^D adversary in the pure CKKS case due to their inability to simulate the noise, for our corrected scheme the noise is eliminated and so the decryptions are simply a function of messages the adversary possesses.

It can be seen that a tight noise analysis for CKKS enables an accurate choice for B in applying this countermeasure. On the other hand, a significant overestimate of the noise would have the undesirable effect of eliminating correct bits during correction. This further motivates the development of tighter analyses of CKKS noise growth.

Another natural direction for future work is to explore whether such a corrected version of CKKS can be developed for arbitrary circuits, whose correctability may not be guaranteed. The difficulty here is the rounding during correction may corrupt higher bits, thus the IND-CPA adversary cannot necessarily simulate decryptions. This may yield an effective attack on the corrected CKKS scheme in the IND-CPA^D model.

References

[1] M. Albrecht, M. Chase, H. Chen, J. Ding, S. Goldwasser, S. Gorbunov, S. Halevi, J. Hoffstein, K. Laine, K. Lauter, S. Lokam, D. Micciancio, D. Moody, T. Morrison, A. Sahai, and V. Vaikuntanathan. Homomorphic encryption security standard. Technical report, HomomorphicEncryption.org, 2018.

[2] Ahmad Al Badawi, Jack Bates, Flavio Bergamaschi, David Bruce Cousins, Saroja Erabelli, Nicholas Genise, Shai Halevi, Hamish Hunt, Andrey Kim, Yongwoo Lee, Zeyu Liu, Daniele Micciancio, Ian Quah, Yuriy Polyakov, Saraswathy R.V., Kurt Rohloff, Jonathan Saylor, Dmitriy Suponitsky, Matthew Triplett, Vinod Vaikuntanathan, and Vincent Zucca. Openfhe: Open-source fully homomorphic encryption library. Cryptology ePrint Archive, Paper 2022/915, 2022. https://eprint.iacr.org/2022/915.

[3] Fabian Boemer, Anamaria Costache, Rosario Cammarota, and Casimir Wierzynski. ngraph-he2: A high-throughput framework for neural network inference on encrypted data. In Michael Brenner, Tancrède Lepoint, and Kurt Rohloff, editors, *Proceedings of the 7th ACM Workshop on Encrypted Computing & Applied Homomorphic Cryptography, WAHC@CCS 2019, London, UK, November 11-15, 2019*, pages 45–56. ACM, 2019.

[4] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *ITCS 2012*, pages 309–325. ACM, January 2012.

[5] Nicolas Brisebarre, Mioara Joldeş, Jean-Michel Muller, Ana-Maria Naneş, and Joris Picot. Error analysis of some operations involved in the cooley-tukey fast fourier transform. *ACM Transactions on Mathematical Software (TOMS)*, 46(2):1–27, 2020.

[6] Hao Chen, Wei Dai, Miran Kim, and Yongsoo Song. Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 395–412. ACM Press, November 2019.

[7] Jung Hee Cheon, Kyoohyung Han, Andrey Kim, Miran Kim, and Yongsoo Song. Bootstrapping for approximate homomorphic encryption. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 360–384. Springer, Heidelberg, April / May 2018.

[8] Jung Hee Cheon, Kyoohyung Han, Andrey Kim, Miran Kim, and Yongsoo Song. A full RNS variant of approximate homomorphic encryption. In Carlos Cid and Michael J. Jacobson Jr:, editors, *SAC 2018*, volume 11349 of *LNCS*, pages 347–368. Springer, Heidelberg, August 2019.

[9] Jung Hee Cheon, Seungwan Hong, and Duhyeong Kim. Remark on the security of CKKS scheme in practice. Cryptology ePrint Archive, Report 2020/1581, 2020. `https://eprint.iacr.org/2020/1581`.

[10] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yong Soo Song. Homomorphic encryption for arithmetic of approximate numbers. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 409–437. Springer, Heidelberg, December 2017.

[11] 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.

[12] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. TFHE: fast fully homomorphic encryption over the torus. *J. Cryptology*, 33(1):34–91, 2020.

[13] Ana Costache and Nigel P. Smart. Which ring based somewhat homomorphic encryption scheme is best? In Kazue Sako, editor, *CT-RSA 2016*, volume 9610 of *LNCS*, pages 325–340. Springer, Heidelberg, February / March 2016.

[14] Anamaria Costache, Kim Laine, and Rachel Player. Evaluating the effectiveness of heuristic worst-case noise analysis in FHE. In Liqun Chen, Ninghui Li, Kaitai Liang, and Steve A. Schneider, editors, *ESORICS 2020, Part II*, volume 12309 of *LNCS*, pages 546–565. Springer, Heidelberg, September 2020.

[15] Anamaria Costache, Lea Nürnberger, and Rachel Player. Optimisations and tradeoffs for helib. In *Topics in Cryptology–CT-RSA 2023: Cryptographers' Track at the RSA Conference 2023, San Francisco, CA, USA, April 24–27, 2023, Proceedings*, pages 29–53. Springer, 2023.

[16] Ivan Damgård, Valerio Pastro, Nigel P. Smart, and Sarah Zakarias. Multiparty computation from somewhat homomorphic encryption. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 643–662. Springer, Heidelberg, August 2012.

[17] Nir Drucker, Guy Moshkowich, Tomer Pelleg, and Hayim Shaul. BLEACH: Cleaning errors in discrete computations over CKKS. Cryptology ePrint Archive, Report 2022/1298, 2022. `https://eprint.iacr.org/2022/1298`.

[18] Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Report 2012/144, 2012. `https://eprint.iacr.org/2012/144`.

[19] Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 169–178. ACM Press, May / June 2009.

[20] Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic evaluation of the AES circuit. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 850–867. Springer, Heidelberg, August 2012.

[21] Shai Halevi and Victor Shoup. Design and implementation of HElib: a homomorphic encryption library. Cryptology ePrint Archive, Report 2020/1481, 2020. `https://eprint.iacr.org/2020/1481`.

[22] Fullrns-heaan. Online: `https://github.com/KyoohyungHan/FullRNS-HEAAN`, October 2018.

[23] Heaan v2.1. Online: `https://github.com/snucrypto/HEAAN`, September 2021.

[24] Heaan v1.0. Online: `https://github.com/snucrypto/HEAAN/releases/tag/1.0`, September 2018.

[25] HElib. `https://github.com/shaih/HElib`, January 2019.

[26] I. Iliashenko. *Optimisations of fully homomorphic encryption.* PhD thesis, KU Leuven, 2019.

[27] A. Kim, Y. Song, M. Kim, and J. H. Lee, K.and Cheon. Logistic regression model training based on the approximate homomorphic encryption. *BMC medical genomics*, 11(4):83, 2018.

[28] Andrey Kim, Antonis Papadimitriou, and Yuriy Polyakov. Approximate homomorphic encryption with reduced approximation error. In Steven D. Galbraith, editor, *Topics in Cryptology - CT-RSA 2022 - Cryptographers' Track at the RSA Conference 2022, Virtual Event, March 1-2, 2022, Proceedings*, volume 13161 of *Lecture Notes in Computer Science*, pages 120–144. Springer, 2022.

[29] Andrey Kim, Yuriy Polyakov, and Vincent Zucca. Revisiting homomorphic encryption schemes for finite fields. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part III*, volume 13092 of *LNCS*, pages 608–639. Springer, Heidelberg, December 2021.

[30] Lattigo v2.2.0. Online: `http://github.com/ldsec/lattigo`, July 2021. EPFL-LDS.

[31] Yongwoo Lee, Joon-Woo Lee, Young-Sik Kim, Yongjune Kim, Jong-Seon No, and HyungChul Kang. High-precision bootstrapping for approximate homomorphic encryption by error variance minimization. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part I*, volume 13275 of *LNCS*, pages 551–580. Springer, Heidelberg, May / June 2022.

[32] Tancrède Lepoint and Michael Naehrig. A comparison of the homomorphic encryption schemes FV and YASHE. In David Pointcheval and Damien Vergnaud, editors, *AFRICACRYPT 14*, volume 8469 of *LNCS*, pages 318–335. Springer, Heidelberg, May 2014.

[33] Baiyu Li and Daniele Micciancio. On the security of homomorphic encryption on approximate numbers. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part I*, volume 12696 of *LNCS*, pages 648–677. Springer, Heidelberg, October 2021.

[34] Baiyu Li, Daniele Micciancio, Mark Schultz, and Jessica Sorrell. Securing approximate homomorphic encryption using differential privacy. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA,*

USA, August 15-18, 2022, Proceedings, Part I, volume 13507 of *Lecture Notes in Computer Science*, pages 560–589. Springer, 2022.

[35] Sean Murphy and Rachel Player. A central limit framework for ring-lwe decryption. Cryptology ePrint Archive, Report 2019/452, 2019. `https://eprint.iacr.org/2019/452`.

[36] Sean Murphy and Rachel Player. Discretisation and product distributions in Ring-LWE. *J. Math. Cryptol.*, 15(1):45–59, 2021.

[37] Tabitha Ogilvie, Rachel Player, and Joe Rowell. Improved privacy-preserving training using fixed-hessian minimisation. In Michael Brenner, Tancrède Lepoint (Eds.), proceedings of the 8th Workshop on Encrypted Computing and Applied Homomorphic Cryptography (WAHC '20), 2020. `https://doi.org/10.25835/0072999`.

[38] PALISADE Lattice Cryptography Library (release 1.11.5). `https://palisade-crypto.org/`, September 2021.

[39] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.

[40] Microsoft SEAL (release 3.6). `https://github.com/Microsoft/SEAL`, November 2020. Microsoft Research, Redmond, WA.

Supplementary Material

A Proof of Theorem 2

Proof of Theorem 2 Part 1: If $Z_j \sim \mathrm{N}(0, \rho^2)$, then $|Z_j| \sim \rho\chi_1$ has a scaled χ-distribution with 1 degree of freedom and cumulative distribution function $F_{|Z_j|}(t) = \mathcal{P}(|Z_j| < t) = \texttt{erf}\left(\frac{t}{\sqrt{2}\rho}\right)$ where $t > 0$. Thus $\|Z\|_\infty = \max\{|Z_1|, \ldots, |Z_N|\}$ has distribution function $F_{\|Z\|_\infty}(t) = \mathcal{P}(\|Z\|_\infty \leq t) = \mathcal{P}(|Z_j| < t)^N = \texttt{erf}\left(\frac{t}{\sqrt{2}\rho}\right)^N$. The probability of $\|Z\|_\infty^{\mathsf{can}}$ exceeding a bound B is therefore given by $\mathcal{P}(\|Z\|_\infty > B) = 1 - \mathcal{P}(\|Z\|_\infty \leq t) = 1 - \texttt{erf}\left(\frac{B}{\sqrt{2}\rho}\right)^N$. Setting this equal to α gives the claimed probability $(1-\alpha)$ bound. □

Proof of Theorem 2 Part 2: We define a "real" version $\tau' \colon \mathbb{R}^N \to \mathbb{R}^N$ of the function $\tau \colon \mathbb{R}^N \to \mathbb{C}^{N/2}$ by the $N \times N$ matrix

$$T' = \begin{pmatrix} \left(\mathrm{Re}(\zeta_j^k)\right) \\ \hline \left(\mathrm{Im}(\zeta_j^k)\right) \end{pmatrix} \qquad \begin{bmatrix} j = 0, \ldots, \frac{1}{2}N - 1 \\ k = 1, \ldots, N \end{bmatrix}.$$

The matrix $\frac{1}{2}N \times N$ matrix $T = \left(I_{N/2} \middle| \, i\, I_{N/2}\right) T'$ then gives the canonical embedding τ. The matrix T' satisfies $T'T'^T = \frac{1}{2}NI_N$ as τ is a scaled isometry with $|\tau(x)|^2 = \frac{1}{2}N|x|^2$ (see Section 3 preamble), so that for $Z \sim \mathrm{N}(\mathbf{0}; \rho^2 I_N)$, we have $X = T'Z \sim \mathrm{N}\left(\mathbf{0}; \frac{N}{2}\rho^2 I_N\right)$. The vector $Y := \tau(Z)$ therefore has real part $\mathrm{Re}(Y) = (X_0, X_1, ..., X_{\frac{N}{2}-1})$ and imaginary part $\mathrm{Im}(Y) = (X_{\frac{N}{2}}, X_{\frac{N}{2}+1}, ..., X_{N-1})$, so that the distributions of $\mathrm{Re}(Y)$ and $\mathrm{Im}(Y)$ can be read off. The bounds follows by part (1) applied to $\mathrm{Im}(Y)$ and $\mathrm{Re}(Y)$. □

Proof of Theorem 2 Part 3: If we write $Y_j = X_j + iX_{j+\frac{1}{2}N}$, then $|Y_j|^2 = X_j^2 + X^2_{j+\frac{1}{2}N}$, which has a scaled χ^2_2-distribution as the sum of two independent squared standard Normal random variables. Thus $|Y_0|, \dots, |Y_{\frac{1}{2}N-1}| \sim (\frac{1}{2}N)^{\frac{1}{2}} \rho \chi_2$ are independent and identically distributed random variables having a scaled χ distribution with 2 degrees of freedom with cumulative distribution function $F_{|Y_j|}(t) = \mathcal{P}\left(|Y_j| \leq t\right) = 1 - \exp\left(-\frac{t^2}{N\rho^2}\right)$ where $t > 0$. Thus $\|Y\|_\infty$ has distribution function $F_{\|Y\|_\infty}(t) = \mathcal{P}(\|Y\|_\infty \leq t) = \mathcal{P}\left(|Y_j| \leq t\right)^{\frac{1}{2}N} = \left(1 - \exp\left(-\frac{t^2}{N\rho^2}\right)\right)^{\frac{1}{2}N}$. For a bound B on $\|Y\|_\infty$, the failure probability α is thus given by $\alpha = \mathcal{P}(\|Y\|_\infty > B) = 1 - \mathcal{P}(\|Y\|_\infty \leq t) = 1 - \left(1 - \exp\left(-\frac{B^2}{N\rho^2}\right)\right)^{\frac{1}{2}N}$, so we derive a $1-\alpha$ probability on $\|Y\|_\infty$ as $B = \sqrt{N}\rho\left(-\ln(1-(1-\alpha)^{\frac{2}{N}})\right)^{\frac{1}{2}}$, which is also a $1-\alpha$ probability bound on $\|Z\|^{\mathsf{can}}_\infty = \|\tau(Z)\|_\infty = \|Y\|_\infty$. □

B Distribution of noise polynomials

In this section, we derive distributional results about CKKS noise polynomials. We give results for the Textbook CKKS scheme [10]; as well as for the RNS variant [8] where the distributional results are the same as for the Textbook case. In particular, these distributional results give us the variances of the noise polynomials after each operation.

Lemma 15. [Fresh – Textbook and RNS] *The noise in a fresh ciphertext has a Normal distribution given by $\epsilon_{\mathsf{fresh}} \sim \mathrm{N}(\mathbf{0}; \rho^2_{\mathsf{fresh}} I_N)$ with component variance $\rho^2_{\mathsf{fresh}} = (\|v\|_2^2 + \|s\|_2^2 + 1)\sigma^2$.*

Proof. We give a proof for the RNS case; the Textbook case is very similar. For all $0 \leq j \leq L$, the j^{th} RNS representative of $\mathtt{ct} = (\mathtt{ct}^{(j)})_{0 \leq j \leq L}$ that is output by encryption has the form $\mathtt{ct}^{(j)} := (b^{(j)}v + e_0 + m, a^{(j)}v + e_1) \in R^2_{q_j}$. The same ephemeral secret $v \leftarrow \chi_{\text{enc}}$ and ephemeral Ring-LWE errors $e_0 \leftarrow \chi_{\text{err}}$, $e_1 \leftarrow \chi_{\text{err}}$ are used in each of the j RNS representatives. The noise in a fresh ciphertext can then be seen to be given by the same expression as for the noise in the analogous Textbook ciphertext:

$$\begin{aligned} \left\langle \mathtt{ct}^{(0)}, \mathtt{sk} \right\rangle &= -a^{(0)}vs + ev + e_0 + m + a^{(0)}vs + e_1 s \mod q_0 \\ &= m + ev + e_0 + e_1 s \mod q_0 \,. \end{aligned}$$

In particular, the noise is given by the polynomial $\epsilon_{\mathtt{fresh}} := ev + e_0 + e_1 s$ where e, e_0, $e_1 \sim N(\mathbf{0}, \sigma^2 I_N)$, and the ephemeral secret $v \leftarrow \chi_{\text{enc}}$ and secret key $s \leftarrow \chi_{\text{key}}$ are fixed vectors. Thus we have $\epsilon_{\mathtt{fresh}} \sim \mathrm{N}(0; \rho^2_{\text{fresh}} I_n)$, where $\rho^2_{\text{fresh}} = (\|v\|_2^2 + \|s\|_2^2 + 1)\sigma^2$. □

For example, if $v \leftarrow \mathcal{ZO}(0.5)$ and s is ternary with fixed Hamming weight h as in [10], this variance is $\rho^2_{\mathsf{fresh}} = (\frac{N}{2} + h + 1)\sigma^2$.

Lemma 16. **[Add – Textbook and RNS]** *Let $\mathtt{ct}_1$, $\mathtt{ct}_2$ be two independent ciphertexts encrypting m_1 and m_2 with noises $\epsilon_1 \sim \mathrm{N}(\mathbf{0}; \rho_1^2 I_N)$ and $\epsilon_2 \sim \mathrm{N}(\mathbf{0}; \rho_2^2 I_N)$ respectively. Let $\mathtt{ct}_{\mathsf{add}}$ with noise ϵ_{add} be the ciphertext resulting from the homomorphic addition of the ciphertexts $\mathtt{ct}_1$ and $\mathtt{ct}_2$. Then $\epsilon_{\mathsf{add}} \sim \mathrm{N}(\mathbf{0}; \rho_{\mathsf{add}}^2 I_N)$ where $\rho_{\mathsf{add}}^2 = \rho_1^2 + \rho_2^2$.*

Proof. Follows directly from Corollary 5. □

Lemma 17. **[Pre-Mult – Textbook and RNS]** *Let $\mathtt{ct}_1$, $\mathtt{ct}_2$ be two independent ciphertexts encrypting m_1 and m_2 with noises $\epsilon_1 \sim \mathrm{N}(\mathbf{0}; \rho_1^2 I_N)$ and $\epsilon_2 \sim \mathrm{N}(\mathbf{0}; \rho_2^2 I_N)$ respectively. Let $\mathtt{ct}_{\mathsf{pre\text{-}mult}}$ with noise $\epsilon_{\mathsf{pre\text{-}mult}}$ be the ciphertext resulting from applying pre-multiplication to the ciphertexts $\mathtt{ct}_1$ and $\mathtt{ct}_2$. If the Small-S assumption is valid for the product distribution $(m_1+\epsilon_1)(m_2+\epsilon_2)$, then $\epsilon_{\mathsf{pre\text{-}mult}} \sim \mathrm{N}(\mathbf{0}; \rho_{\mathsf{pre\text{-}mult}}^2 I_N)$ and the noise variance $\rho_{\mathsf{pre\text{-}mult}}^2$ is given by:*

$$\rho_{\mathsf{pre\text{-}mult}}^2 = N\rho_1^2\rho_2^2 + \rho_2^2 \left\| m_1 \right\|_2^2 + \rho_1^2 \left\| m_2 \right\|_2^2 .$$

Proof. We give a proof for the RNS case; the Textbook case is very similar. For $0 \le j \le \ell$, for input ciphertexts $\mathtt{ct}_1 = \{\mathtt{ct}_1^{(j)}\} = \left\{\left(c_0^{(j)}, c_1^{(j)}\right)\right\}$ and $\mathtt{ct}_2 = \{\mathtt{ct}_2^{(j)}\} = \left\{\left(C_0^{(j)}, C_1^{(j)}\right)\right\}$, the output $\mathtt{ct}_{\text{pre-mult}} = \{\mathtt{ct}_{\text{pre-mult}}^{(j)}\}$ has representatives $(d_0^{(j)}, d_1^{(j)}, d_2^{(j)})$ given by $d_0^{(j)} = c_0^{(j)} C_0^{(j)} \mod q_j$, $d_1^{(j)} = c_0^{(j)} C_1^{(j)} + c_1^{(j)} C_0^{(j)} \mod q_j$, and $d_2^{(j)} = c_1^{(j)} C_1^{(j)} \mod q_j$. For input ciphertexts $\{\mathtt{ct}_1^{(j)}\}_{0 \le j \le \ell}$ encrypting m_1 with noise ϵ_1 and $\{\mathtt{ct}_2^{(j)}\}_{0 \le j \le \ell}$ encrypting m_2 with noise ϵ_2, we have:

$$\begin{aligned}
\left\langle \mathtt{ct}_{\text{pre-mult}}^{(0)}, \mathtt{sk} \right\rangle &= d_0^{(0)} + d_1^{(0)} s + d_2^{(0)} s^2 = (c_0^{(0)} + c_1^{(0)} s) \cdot (C_0^{(0)} + C_1^{(0)} s) \mod q_0 \\
&= \left\langle \mathtt{ct}_1^{(0)}, \mathtt{sk} \right\rangle \cdot \left\langle \mathtt{ct}_2^{(0)}, \mathtt{sk} \right\rangle = (m_1 + \epsilon_1) \cdot (m_2 + \epsilon_2) \mod q_0 \\
&= m_1 m_2 + \epsilon_1 m_2 + \epsilon_2 m_1 + \epsilon_1 \epsilon_2 \mod q_0 .
\end{aligned}$$

so $\{\mathtt{ct}_{\text{pre-mult}}^{(j))}\}_{0 \le j \le \ell}$ encrypts $m_1 m_2$ with noise $\epsilon_{\mathtt{pre-mult}} := \epsilon_1 m_2 + \epsilon_2 m_1 + \epsilon_1 \epsilon_2$. Given that ϵ_1 and ϵ_2 are distributed Normally, and that the Small-S assumption holds for $m_1 + \epsilon_1$ and $m_2 + \epsilon_2$, then $\epsilon_{\mathtt{pre-mult}}$ is also distributed Normally by Corollary 5. □

Lemma 18. **[Round – Textbook and RNS]** *The component-wise rounding of a ciphertext $(\mathtt{ct}_0, \mathtt{ct}_1) \in (\mathbb{R}[X]/(X^N+1))^2$ in $\mathcal{R}_Q$ introduces an additive rounding error ε_{round}, which has a Normal distribution given by $\varepsilon_{\mathsf{round}} \sim \mathrm{N}(\mathbf{0}, \eta_{\mathsf{round}}^2 I_N)$ with component variance $\eta_{\mathsf{round}}^2 = \frac{1}{12}(\left\| s \right\|_2^2 + 1)$.*

Proof. Let $\mathtt{ct} = (\mathtt{ct}_0, \mathtt{ct}_1)$ be a ciphertext encrypting a message m with noise ε. Let $\mathtt{ct}_{\mathsf{round}} = (\mathtt{ct}_0', \mathtt{ct}_1')$ be the result of applying component-wise rounding to $\mathtt{ct}$, so $\mathtt{ct}_0' + \mathtt{ct}_1' s = \mathtt{ct}_0 + \tau_0 + (\mathtt{ct}_1 + \tau_1)s = \mathtt{ct}_0 + \mathtt{ct}_1 s + \tau_0 + \tau_1 s = m + \varepsilon + \tau_0 + \tau_1 s$, where τ_i is the rounding error introduced in each component. The rounding process thus introduces an additive noise $\epsilon := \tau_0 + \tau_1 s$, where τ_0 and τ_1 are modelled as being drawn uniformly at random with components in $\left[-\frac{1}{2}, \frac{1}{2}\right]$. Thus the j^{th} component of this additive error is given by $\epsilon_j = \tau_{0,j} + \sum_{k=0}^{N-1} \xi(k-j) s_k \; \tau_{1,j-k}$, where ξ is a modified sign function given by $\xi(z) = \text{Sign}(z)$ for $z \neq 0$ and $\xi(0) = 1$ arising from the multiplication modulo $X^n + 1$ (also see Theorem 3). This component ϵ_j therefore has mean 0 and variance satisfying $\eta^2_{\mathsf{round}} = \text{Var}(\epsilon_j) = \text{Var}(\tau_{0,j}) + \sum_{k=0}^{N-1} (\xi(k-j)s_k)^2 \; \text{Var}(\tau_{1,j-k}) = \frac{1}{12} + \|s\|_2^2 \cdot \frac{1}{12} = \frac{1}{12}(\|s\|_2^2 + 1)$. Similarly, we can show for $j' \neq j$ that $\text{Cov}(\epsilon_j, \epsilon_{j'}) = \mathbf{E}(\epsilon_j \epsilon_{j'}) = \sum_{k=0}^{N-1} \xi(k-j)^2 s_k s_{j'-j+k} \; \text{Var}(\tau_{1,j-k}) = \frac{1}{12} \sum_{k=0}^{N-1} s_k s_{j'-j+k}$, as s_i are uniformly distributed over $\{-1, 0, 1\}$. Thus $\text{Cov}(\epsilon_j, \epsilon_{j'})$ can be modelled by a Normal $\text{N}(0, \frac{1}{72}N)$ distribution, so is far smaller than η^2_{round}, and we can regard $\text{Cov}(\epsilon_j, \epsilon_{j'}) \approx 0$.

We have now found the mean and variance and established that the covariance is negligible. It remains to show normality. As shown above, the error component ϵ_j is the sum of the independent random variables $\tau_{0,j}$ and $\xi(k-j)s_k\tau_{1,j-k}$ (for $k = 0, \ldots, N-1$) which are each independent random variables uniformly distributed on $\left[-\frac{1}{2}, \frac{1}{2}\right]$ (when $s_k \neq 0$). Thus, ϵ_j is the sum of $(\|s\|_2 + 1)$ independent and identically distributed random variables with mean 0 and variance $\frac{1}{12}$, so the Central Limit Theorem shows that ϵ_j has an approximate Normal $\text{N}(0, \eta^2_{\mathsf{round}})$ distribution. Thus ϵ can be modelled as a multivariate Normal $\text{N}(\mathbf{0}; \eta^2_{\mathsf{round}} I_N)$ distribution. □

For example, if the secret is sparse with fixed Hamming weight h as in [10], we have $\eta^2_{\mathsf{round}} = \frac{1}{12}(h+1)$.

Lemma 19. [Key Switch – Textbook] *The output noise after the* Textbook CKKS *key switch operation is given by $\epsilon_{\mathsf{ks}} := \epsilon + \varepsilon_{\mathsf{ks}}$ if the input noise is given by ϵ. The additive error $\varepsilon_{\mathsf{ks}}$ has a Normal distribution given by $\varepsilon_{\mathsf{ks}} \sim \text{N}(\mathbf{0}, \eta^2_{\mathsf{ks}} I_N)$, where $\eta^2_{\mathsf{ks}} = \frac{1}{12}\left(P^{-2}NQ_\ell^2\sigma^2 + \mathbb{1}_{P \nmid Q_\ell}(\|s\|_2^2 + 1)\right)$.*

Proof. The keyswitch operation introduces an additive error $\epsilon_{\mathsf{ks}} = P^{-1}Q_\ell \cdot d_2 e' + \epsilon_{\mathsf{round}}$ (see for example [10, Lemma 3]). In this expression, $e' \sim \text{N}(\mathbf{0}; \sigma^2 I_N)$ is the Ring-LWE noise term in the evaluation key, $d_2 = [\mathtt{ct}_0[1]\mathtt{ct}_1[1]]_{Q_\ell}$ is a component from the output of pre-multiply, and $\epsilon_{\mathrm{round}}$ is a possible rounding error created by dividing by P^{-1}. We can regard a component of d_2 given by $d_{2,i} = \sum_{j=0}^{N-1} \xi(i-j)\mathtt{ct}_0[1]_j \mathtt{ct}_1[1]_{i-j} \bmod Q_\ell$ as having a uniform distribution on $(-\frac{1}{2}Q_\ell, \frac{1}{2}Q_\ell)^N$ (including the "squaring" case $\mathtt{ct}_0[1] = \mathtt{ct}_1[1]$), so $d_{2,i}$ has mean 0 and variance $\frac{1}{12}Q_\ell^2$. A Central Limit argument similar to the proof of Lemma 18 then shows that $P^{-1}(d_2 e') \sim \text{N}(\mathbf{0}\; ; \; \frac{1}{12}P^{-2}NQ_\ell^2\sigma^2 I_N)$.

The additional rounding error e_{round} is required when P does not divide Q_ℓ, in which case Lemma 18 shows that $e_{\mathsf{round}} \sim \text{N}(\mathbf{0}; \frac{1}{12}(\|s\|_2^2 + 1)I_N)$. In these circumstances, ϵ_{ks} is the sum of two independent multivariate Normal distributions, so has a multivariate Normal distribution itself with mean $\mathbf{0}$ and component variance $\frac{1}{12}P^{-2}NQ_\ell^2\sigma^2 + \frac{1}{12}(\|s\|_2^2 + 1)$. □

For example, if the secret is sparse with fixed Hamming weight h as in [10], we have $\eta^2_{\mathsf{ks}} = \frac{1}{12}P^{-2}NQ_\ell^2\sigma^2 + \mathbb{1}_{P \nmid Q_\ell}\frac{1}{12}(h+1)$.

Lemma 20. [Rescale – Textbook] *Let* $\mathtt{ct}$ *be a ciphertext encrypting* m *with noise* $\epsilon \sim N(\mathbf{0}; \rho^2 I_N)$. *Let* $\mathtt{ct}_{\mathtt{rs}}$ *encrypting* m *be the ciphertext with noise* $\epsilon_{\mathtt{rs}}$ *resulting from the* $\mathtt{Rescale}$ *operation. Then* $\epsilon_{\mathtt{rs}} \sim N(\mathbf{0}; \rho^2_{\mathtt{rs}} I_N)$, *where the component variance* $\rho^2_{\mathtt{rs}}$ *is given by* $\rho^2_{\mathtt{rs}} = \frac{1}{\Delta^2}\rho^2 + \frac{1}{12}(\|s\|_2^2 + 1)$.

Proof. The ciphertexts after the $\mathtt{Rescale}$ operation are $\mathtt{ct}'_i = \left[\left\lfloor \frac{1}{\Delta}\mathtt{ct}_i \right\rceil\right]_{Q_{\ell-1}}$ $(i = 0, 1)$, so we have $\mathtt{ct}'_0 + s\mathtt{ct}'_1 = \left[\left\lfloor \dfrac{1}{\Delta}\mathtt{ct}_0 \right\rceil\right]_{Q_{\ell-1}} + s\left[\left\lfloor \dfrac{1}{\Delta}\mathtt{ct}_1 \right\rceil\right]_{Q_{\ell-1}} = \dfrac{1}{\Delta}(\mathtt{ct}_0 + s\mathtt{ct}_1) + (\tau_0 + s\tau_1)$ where τ_0, τ_1 are rounding random variables as in Lemma 18 with $\epsilon_{\mathtt{round}} = \tau_0 + s\tau_1$. Thus Lemma 18 shows that $\epsilon_{\mathtt{rs}} = \frac{1}{\Delta}\epsilon + \epsilon_{\mathtt{round}}$, where $\epsilon_{\mathtt{round}} \sim \mathrm{N}(\mathbf{0}; \eta^2_{\mathtt{ks}} I_N)$ and $\eta^2_{\mathtt{ks}} = \frac{1}{12}(\|s\|_2^2 + 1)$, so $\epsilon_{\mathtt{rs}} \sim \mathrm{N}(\mathbf{0}; \rho^2_{\mathtt{rs}})$, where $\rho^2_{\mathtt{rs}} = \frac{1}{\Delta^2}\rho^2 + \eta^2_{\mathtt{ks}} = \frac{1}{\Delta^2}\rho^2 + \left(\frac{1}{12}(\|s\|_2^2 + 1)\right)$. □

C Justification of worst-case noise bounds

In this section, we justify the worst-case noise bounds presented in Section 4.4. For illustration, we give the bounds for the noise after fresh encryption, after the rounding operation, and after the key switching variant in [10]. Bounds for other variants of key switching, e.g. as in [8] can be proved similarly. Worst-case bounds on addition and multiplication are proved in prior work for the canonical embedding case [10, 8], and the bound in the ring for these operations is proved analogously, so we omit presenting them in detail. Similarly, worst-case bounds for the rescale operation can also be proved in the same manner as prior work [10, 8] using Lemma 22.

In more detail, we use Theorem 2 and the variances derived in Supplementary Material Section B to give worst-case bounds on the noise ϵ in a ciphertext obtained after each homomorphic evaluation operation. We give probabilistic bounds, defined by the parameter α, both in the ring (i.e., bounding $\|\epsilon\|_\infty$), and in the canonical embedding (i.e., bounding $\|\epsilon\|_\infty^{\mathsf{can}}$).

Lemma 21. [Fresh Bound - Textbook and RNS] *Let* $\mathtt{ct}$ *be a fresh ciphertext encrypting a message* m *with noise* ϵ. *In the canonical embedding, with probability* $1 - \alpha$, *we have that* $\|\epsilon\|_\infty^{\mathsf{can}} \leq B_{\mathit{fresh}}$, *where*

$$B_{\mathit{fresh}} = \sqrt{N\rho^2_{\mathit{fresh}}}\, H_{\mathbb{C}}(\alpha, N) = \sigma\sqrt{N}\sqrt{\|v\|_2^2 + \|s\|_2^2 + 1}\, H_{\mathbb{C}}(\alpha, N).$$

In the ring, with probability $1 - \alpha$, *we have that* $\|\epsilon\|_\infty \leq B_{\mathit{fresh}}$, *where*

$$B_{\mathit{fresh}} = \sqrt{2}\rho^2_{\mathit{fresh}}\, H_{\mathbb{R}}(\alpha, N) = \sigma\sqrt{2}\sqrt{\|v\|_2^2 + \|s\|_2^2 + 1}\, H_{\mathbb{R}}(\alpha, N).$$

Proof. Lemma 15 shows that the encryption noise has a multivariate Normal $\mathrm{N}(\mathbf{0}; \rho^2_{\mathsf{fresh}} I_N)$ distribution, where $\rho^2_{\mathsf{fresh}} = (\|v\|_2^2 + \|s\|_2^2 + 1)\sigma^2$. The worst case bounds on the encryption noise under the canonical embedding norm and in the ring then follow from Theorem 2 (3) and Theorem 2 (1) respectively. □

Lemma 22. **[Round Bound - Textbook and RNS]** *Consider rounding a non-integer ciphertext. In the canonical embedding, a worst-case bound on the additive noise introduced by this process, that holds with probability $1-\alpha$, is given by*

$$B_{\mathsf{round}} = \sqrt{N\eta^2_{\mathsf{round}}}\, H_{\mathbb{C}}(\alpha, N) = \sqrt{\tfrac{N}{12}(\|s\|_2^2 + 1)}\, H_{\mathbb{C}}(\alpha, N).$$

In the ring, a worst-case bound on the additive noise introduced by this process, that holds with probability $1-\alpha$, is given by

$$B_{\mathsf{round}} = \sqrt{2\eta^2_{\mathsf{round}}}\, H_{\mathbb{R}}(\alpha, N) = \sqrt{\tfrac{1}{6}(\|s\|_2^2 + 1)}\, H_{\mathbb{R}}(\alpha, N).$$

Proof. Lemma 18 shows that the rounding operation introduces an additive noise with a multivariate Normal $\mathrm{N}(\mathbf{0}; \eta^2_{\mathsf{round}} I_N)$ distribution, where $\eta^2_{\mathsf{round}} = \frac{1}{12}(\|s\|_2^2 + 1)$. The worst case bounds on the rounding noise under the canonical embedding norm and in the ring then follow from Theorem 2 (3) and Theorem 2 (1) respectively. □

Lemma 23. **[Keyswitch Bound - Textbook]** *Let $\mathtt{ct}_{\mathsf{mult}}$ be the ciphertext resulting from the key switch operation applied on the ciphertext $\mathtt{ct}$. This operation introduces an additive error term ϵ_{ks} that can be bounded as follows. In the canonical embedding, with probability $1-\alpha$, we have that $\|\epsilon_{\mathsf{ks}}\|_\infty^{\mathsf{can}} \le B_{\mathsf{ks}}$, where*

$$\begin{aligned} B_{\mathsf{ks}} &= \sqrt{N\eta^2_{\mathsf{ks}}}\, H_{\mathbb{C}}(\alpha, N) \\ &= \sqrt{N\left(\tfrac{1}{12}P^{-2}NQ_\ell^2\sigma^2 + \mathbb{1}_{P\nmid q_\ell}\left(\tfrac{1}{12}(\|s\|_2^2 + 1)\right)\right)} H_{\mathbb{C}}(\alpha, N). \end{aligned}$$

In the ring, with probability $1-\alpha$, we have that $\|\epsilon_{\mathsf{ks}}\|_\infty \le B_{\mathsf{ks}}$, where

$$\begin{aligned} B_{\mathsf{ks}} &= \sqrt{2\eta^2_{\mathsf{ks}}} H_{\mathbb{R}}(\alpha, N) \\ &= \sqrt{\tfrac{1}{6}\left(P^{-2}NQ_\ell^2\sigma^2 + \mathbb{1}_{P\nmid q_\ell}\left(\|s\|_2^2 + 1\right)\right)} H_{\mathbb{R}}(\alpha, N). \end{aligned}$$

Proof. Lemma 19 shows that the rounding operation introduces an additive noise with a multivariate Normal $\mathrm{N}(\mathbf{0}; \eta^2_{\mathsf{ks}} I_N)$ distribution, where

$$\eta^2_{\mathsf{ks}} = \tfrac{1}{12}\left(P^{-2}NQ_\ell^2\sigma^2 + \mathbb{1}_{P\nmid Q_\ell}(\|s\|_2^2 + 1)\right).$$

The worst case bounds on the rounding noise under the canonical embedding norm and in the ring then follow from Theorem 2 (3) and Theorem 2 (1) respectively. □

D Canonical embedding norm bounds in prior work

To support a comparison to prior work for each of our experiments, this section summarises the prior worst-case canonical embedding norm noise bounds used in prior noise analyses of variants of CKKS [10, 7, 8, 28], developed using the methodology in [20, 13]. Table 6 gives the prior noise bounds for Textbook CKKS [10] and Table 7 gives the prior noise bounds for the RNS variant [8].

Operation	Bound
Fresh	$8\sqrt{2}\sigma N + 6\sigma\sqrt{N} + 16\sigma\sqrt{hN}$
Add	$B_1 + B_2$
Pre-Mult	$\nu_1 B_2 + \nu_2 B_1 + B_1 B_2$
Key Switch	$B + \frac{8}{\sqrt{3}} P^{-1} N Q_\ell \sigma$
Rescale	$\Delta^{-1} B + \sqrt{3N} + \frac{8}{\sqrt{3}}\sqrt{hN}$

Table 6: Prior canonical embedding bounds for Textbook CKKS [10] (see [10, Appendix B]). The notation ν_i denotes an upper bound on the message m_i.

Operation	Bound
Fresh	$8\sqrt{2}\sigma N + 6\sigma\sqrt{N} + 16\sigma\sqrt{hN}$
Add	$B_1 + B_2$
Pre-Mult	$\nu_1 B_2 + \nu_2 B_1 + B_1 B_2$
Key Switch	$B + \frac{8}{\sqrt{3}} N\sigma \frac{Q}{P}\sqrt{\ell^2+1} + (k+1)\left(\sqrt{3N} + \frac{8}{\sqrt{3}}\sqrt{hN}\right)$
Rescale	$q_\ell^{-1} B + \sqrt{3N} + \frac{8}{\sqrt{3}}\sqrt{hN}$

Table 7: Prior canonical embedding bounds for the RNS variant [8] (see [8, Appendix A.2]). The notation ν_i denotes an upper bound on the message m_i.

E Proofs of noise distribution results in Section 5.2

E.1 Proof of Lemma 9

We will prove Lemma 9 in terms of the 'full' output ciphertext $\texttt{ct}_{\mathrm{ks}}$, rather than its RNS representatives. This follows the proof structure of [8], and eases the analysis. We first recall the structure of the evaluation key, and relevant details from [8] about the functionality of the `Conv`, `ModUp`, and `ModDown` algorithms. We then give the proof.

For full details of the key switching process, including a specification of the `Conv`, `ModUp`, and `ModDown` algorithms, we refer the reader to [8]. We denote by $\mathcal{B}$ the set of moduli $\{p_0, \ldots, p_k\}$, by $\mathcal{C}$ the set of moduli $\{q_0, \ldots, q_L\}$, and by $\mathcal{D}$ the set $\mathcal{D} = \mathcal{B} \cup \mathcal{C}$.

Evaluation key. The evaluation key is given by the RNS representatives in the basis $\mathcal{D}$ of (b', a') where $b' = -a's + e' + Ps^2$. Making explicit the reduction modulo PQ, this gives, (for a suitable integer λ): $b' + a's = e' + Ps^2 + \lambda PQ$.

The `ModUp` algorithm. We recall that $\texttt{ModUp}_{\mathcal{C}\to\mathcal{D}}([z]_{\mathcal{C}}) = (\texttt{Conv}_{\mathcal{C}\to\mathcal{B}}([z]_{\mathcal{C}}), [z]_{\mathcal{C}})$. We give the detail of the implementation of `Conv` needed to analyse the noise. In particular, the computation of $\texttt{Conv}_{\mathcal{C}\to\mathcal{B}}$ involves reducing the following sum modulo p_i for each $0 \leq i \leq k$:

$$S = \sum_{j=0}^{\ell-1} [a^{(j)} \widehat{q_j}^{-1}]_{q_j} \widehat{q}_j \, .$$

We consider modulo reduction centred on 0 and write $X_j = [a^{(j)} \widehat{q_j}^{-1}]_{q_j} \in \{-\frac{1}{2}(q_j - 1), \ldots, \frac{1}{2}(q_j - 1)\}$. We can therefore express the sum as

$$S = \sum_{j=0}^{\ell-1} [a^{(j)} \widehat{q_j}^{-1}]_{q_j} \widehat{q}_j = a + \nu Q,$$

where ν is a small integer. We now address the variance of this small integer ν. In [8, Section 3.1], it is suggested that the primes $q_0, \ldots, q_{\ell-1} \approx q$ are all of comparable size q, so we can approximate $Q \approx q^\ell$ and $\widehat{q}_0, \ldots, \widehat{q}_{\ell-1} \approx q^{\ell-1}$. We can therefore approximate $X_0, \ldots, X_{\ell-1}$ as independent Uniform random variables on $(-\frac{1}{2}q, \frac{1}{2}q)$ with mean $\mathbf{E}(X_i) = 0$ and variance $\mathrm{Var}(X_i) = \frac{1}{12}q$. Thus we obtain the sum $S = a + \nu Q = \sum_{j=0}^{\ell-1} [a^{(j)} \widehat{q_j}^{-1}]_{q_j} \widehat{q}_j \approx \sum_{j=0}^{\ell-1} X_i q^{\ell-1} = q^{\ell-1} \sum_{j=0}^{\ell-1} X_i$, with variance $\mathrm{Var}(S) \approx q^{2(\ell-1)} \sum_{j=0}^{\ell-1} \mathrm{Var}(X_i) = q^{2(\ell-1)} \ell \frac{1}{12} q^2 = \frac{1}{12} \ell q^{2\ell} = \frac{1}{12} \ell Q^2$. This gives an approximation to the variance of $\nu \approx \dfrac{S - a}{Q}$ as $\mathrm{Var}(\nu) \approx \frac{\mathrm{Var}(S)}{Q^2} \approx \frac{1}{12}\ell$.

The `ModDown` algorithm. At a high level, the inputs to `ModDown` are the RNS representations of $\widetilde{c}_0 := (d_2 + Qe)b' \mod PQ$ and $\widetilde{c}_1 := (d_2 + Qe)a' \mod PQ$, and the outputs are RNS representations of $\hat{c}_0$ and $\hat{c}_1$ such that $\hat{c}_0 \approx P^{-1}(d_2 + Qe)b'$ and $\hat{c}_1 \approx P^{-1}(d_2 + Qe)a'$. In more detail, for $\iota \in \{0, 1\}$, suppose we can find a small $\widetilde{\epsilon_\iota} \mod PQ$ such that $\widetilde{\epsilon_\iota} \equiv \widetilde{c}_\iota \mod P$. This would imply that $\widetilde{c}_\iota - \widetilde{\epsilon_\iota}$ is divisible by P and so we can take $\hat{c}_\iota = P^{-1}(\widetilde{c}_\iota - \widetilde{\epsilon_\iota}) \mod Q$. In particular, $\hat{c}_\iota \approx P^{-1}\widetilde{c}_\iota$ with approximation error $P^{-1}\widetilde{\epsilon_\iota}$. For $\iota \in \{0, 1\}$, we let $\epsilon_\iota := [\widetilde{c}_\iota]_P$. We observe that the RNS representation of ϵ_ι in the basis $\mathcal{B}$ is exactly those elements $[\widetilde{c}_\iota]_{\mathcal{B}}$. We then compute the RNS representatives of $\widetilde{\epsilon_\iota} = \epsilon_\iota + P e''_\iota$ in the basis $\mathcal{C}$ via $[\widetilde{\epsilon_\iota}]_{\mathcal{C}} = \mathtt{Conv}_{\mathcal{B} \to \mathcal{C}}\left([\epsilon_\iota]_{\mathcal{B}}\right) = \mathtt{Conv}_{\mathcal{B} \to \mathcal{C}}\left([\widetilde{c}_\iota]_{\mathcal{B}}\right)$.

Proof of Lemma 9. Consider the ciphertext $\mathtt{ct}_{\mathrm{ks}}$ that is the output of key switching on input $\left(d_0^{(j)}, d_1^{(j)}, d_2^{(j)}\right)_{0 \leq j \leq \ell}$, the RNS representation of a ciphertext $\mathtt{ct}_{\mathrm{mult}} = (d_0, d_1, d_2)$ at level ℓ that encrypts m with noise ϵ. We have:

$$\begin{aligned}
\langle \mathtt{ct}_{\mathrm{ks}}, \mathtt{sk} \rangle &= d_0 + \widehat{c}_0 + (d_1 + \widehat{c}_1)\, s = (d_0 + d_1 s) + (\widehat{c}_0 + \widehat{c}_1 s) \\
&= (d_0 + d_1 s) + \left(\frac{(\widetilde{c}_0 + \widetilde{c}_1 s) - (\widetilde{\epsilon}_0 + \widetilde{\epsilon}_1 s)}{P} \right) \\
&= (d_0 + d_1 s) + \left(\frac{(d_2 + Qe)(b' + a' s) - (\widetilde{\epsilon}_0 + \widetilde{\epsilon}_1 s)}{P} \right) \\
&= (d_0 + d_1 s) + (d_2 + Qe)(s^2 + \lambda Q) + \left(\frac{(d_2 + Qe)e' - (\widetilde{\epsilon}_0 + \widetilde{\epsilon}_1 s)}{P} \right)
\end{aligned}$$

$$= (d_0 + d_1 s + d_2 s^2) + \left(\frac{(d_2 + Qe)e' - (\widetilde{\epsilon_0} + \widetilde{\epsilon_1} s)}{P} \right) \mod Q\,.$$

Thus, the additive error term introduced in key switching is the integer value

$$\varepsilon_{\mathtt{ks}} = \left(\frac{(d_2 + Qe)e'}{P} \right) - \left(\frac{\widetilde{\epsilon_0} + \widetilde{\epsilon_1} s}{P} \right).$$

To simplify the second term, by definition, $\widetilde{\epsilon}_0 = \epsilon_0 + Pe_0''$; and $\widetilde{\epsilon}_1 = \epsilon_1 + Pe_1''$. So, $\frac{\widetilde{\epsilon_0}+\widetilde{\epsilon_1} s}{P} = \frac{\epsilon_0}{P} + \frac{\epsilon_1}{P} s + e_0'' + e_1'' s = u_0 + u_1 s + e_0'' + e_1'' s$, where we define $u_0 := \frac{\epsilon_0}{P}$ and $u_1 := \frac{\epsilon_1}{P}$. By definition, we have that $\epsilon_0 := [\widetilde{c_0}]_P$ and $\epsilon_1 := [\widetilde{c_1}]_P$ and so we can model ϵ_0 and ϵ_1 as uniform with coefficients in $\left[-\frac{P}{2}, \frac{P}{2}\right]$. Hence, we can model u_0 as uniform with coefficients in $\left[-\frac{1}{2}, \frac{1}{2}\right]$, and similarly for u_1. This leads to the final noise expression $\langle \mathtt{ct}_{\mathrm{ks}}, \mathtt{sk} \rangle = m + \epsilon + \left(\frac{(d_2+Qe)e'}{P} \right) - (u_0 + u_1 s) - (e_0'' + e_1'' s)$.

In the above expression, $e = (\nu_0, \ldots, \nu_{N-1})$ is a multivariate version of the univariate ν described in the `ModUp` paragraph above, having independent components $\nu_0, \ldots, \nu_{N-1}$. The random variable $d_2 + Qe$ has mean 0 and component variance $\frac{1}{12}Q^2 + \frac{1}{12}Q^2\ell^2 = \frac{1}{12}Q^2(\ell^2 + 1)$. A Central Limit argument similar to the proof of Lemma 18 shows that the additive noise term $\frac{(d_2+Qe)e'}{P} \sim \mathrm{N}(\mathbf{0}; \frac{1}{12}P^{-2}NQ^2(\ell^2+1)\sigma^2 I_N)$. By the same argument as Lemma 18, the additive noise term $u_0 + u_1 s$ can be modelled as a multivariate Normal $\mathrm{N}(\mathbf{0}; \eta^2_{\mathsf{round}} I_N)$ distribution where $\eta^2_{\mathsf{round}} = \frac{1}{12}(\|s\|_2^2 + 1)$. The `ModDown` errors e_0'' and e_1'' can be analysed in the same way as the `ModUp` errors so their coefficients have variance $\frac{k^2}{12}$. The component variance of the additive noise term given by $e_0'' + e_1'' s$ is then given by: $\frac{k^2}{12}(\|s\|_2^2 + 1)$, using the same argument as in the proof of Lemma 18. The above means the overall η^2_{ks} is given by

$$\begin{aligned} \eta^2_{\mathsf{ks}} &= \tfrac{1}{12}P^{-2}NQ^2(\ell^2+1)\sigma^2 + \tfrac{1}{12}(\|s\|_2^2+1) + \tfrac{k^2}{12}(\|s\|_2^2+1) \\ &= \tfrac{1}{12}P^{-2}NQ^2(\ell^2+1)\sigma^2 + \tfrac{1}{12}(k^2+1)(\|s\|_2^2+1). \end{aligned}$$

E.2 Proof of Lemma 10

Proof. We consider the ciphertext (c_0, c_1) with RNS components given by $c_0 = (c_0^{(0)}, \ldots, c_0^{(\ell)})$ and $c_1 = (c_1^{(0)}, \ldots, c_1^{(\ell)})$. The value $c_0 + sc_1$ has RNS components $c_0^{(j)} + sc_1^{(j)}$ $(j = 0, \ldots, \ell)$ satisfying $c_0^{(j)} + sc_1^{(j)} = m + \epsilon \mod q_j$, where the error ϵ is small and the modulus q_0 is used for decryption. However, these decryption components are small and constant for all j, so we have $c_0 + sc_1 = m + \epsilon \bmod Q$.

The two-part ciphertext (c_0', c_1') obtained after the `Rescale` operation can be expressed in terms dividing c_0 and c_1 by q_ℓ and then rounding to the nearest integer [28], so $c_0' = \left\lfloor \frac{c_0}{q_\ell} \right\rceil = \frac{c_0}{q_\ell} + U_0$ and $c_1' = \left\lfloor \frac{c_1}{q_\ell} \right\rceil = \frac{c_1}{q_\ell} + U_1$, where $U_0 = (U_{0,0}, \ldots, U_{0,\ell-1})$ and $U_1 = (U_{1,0}, \ldots, U_{1,\ell-1})$ are vectors of independent rounding $U_{i,j} \sim \mathrm{Uni}((-\frac{1}{2}, \frac{1}{2}))$ random variables with mean $\mathbf{E}(U_{i,j}) = 0$ and variance $\mathrm{Var}(U_{i,j}) = \frac{1}{12}$. Thus we have $c_0' + sc_1' = \frac{c_0 + sc_1}{q_\ell} + (U_0 + sU_1) = \frac{m}{q_\ell} + \frac{\epsilon}{q_\ell} + (U_0 + sU_1)$, where $s = (s_0, \ldots, s_{\ell-1})$. As the righthand side is small, the reduction modulo q_0 gives $c_0'^{(0)} + sc_1'^{(0)} = \frac{m}{q_\ell} + \frac{\epsilon}{q_\ell} + (U_0 + sU_1)$. The additive error given by the `Rescale` operation is therefore given by $\epsilon_{\texttt{rs}} = \frac{\epsilon}{q_\ell} + (U_0 + sU_1)$, where the first term $\dfrac{\epsilon}{q_\ell} \sim \mathrm{N}\left(\mathbf{0}; \dfrac{\rho^2}{q_\ell^2} I_N\right)$. For the second term, Lemma 18 gives the Normal approximation $\epsilon_{\texttt{rs}} = U_0 + sU_1 \sim \mathrm{N}(0; \eta^2_{\texttt{round}} I_\ell)$, where $\eta^2_{\texttt{round}} = \frac{1}{12}(\|s\|_2^2 + 1)$. Thus we have $\epsilon_{\texttt{rs}} \sim \mathrm{N}(\mathbf{0}; \rho^2_{\texttt{rs}} I_N)$, where $\rho^2_{\texttt{rs}} = \dfrac{\rho^2}{q_\ell^2} + (\frac{1}{12}(\|s\|_2^2 + 1))$. □

F Proofs of results in Section 5.4

F.1 Proof of Lemma 11

The noise in the ciphertext output by the `Adjust` by one level operation is given by

$$\begin{aligned}
\langle \texttt{ct}_{\text{adj}}, s\rangle &= \left\lceil \frac{1}{q_{\ell+1}} \left[\left\lceil \frac{\Delta_\ell \cdot q_{\ell+1}}{\Delta_{\ell+1}} c_0 \right\rfloor \right]_{Q_{\ell+1}} \right\rfloor \\
&\quad + \left(\left\lceil \frac{1}{q_{\ell+1}} \left[\left\lceil \frac{\Delta_\ell \cdot q_{\ell+1}}{\Delta_{\ell+1}} c_1 \right\rfloor \right]_{Q_{\ell+1}} \right\rfloor \right) s + k' Q_\ell \\
&= \left\lceil \frac{1}{q_{\ell+1}} \left(\frac{\Delta_\ell \cdot q_{\ell+1}}{\Delta_{\ell+1}} c_0 + \epsilon_0 + k_0 Q_{\ell+1} \right) \right\rfloor \\
&\quad + \left(\left\lceil \frac{1}{q_{\ell+1}} \left(\frac{\Delta_\ell \cdot q_{\ell+1}}{\Delta_{\ell+1}} c_1 + \epsilon_1 + k_1 Q_{\ell+1} \right) \right\rfloor \right) s + k Q_\ell \\
&= \frac{\Delta_\ell}{\Delta_{\ell+1}} c_0 + \frac{1}{q_{\ell+1}} \epsilon_0 + \frac{1}{q_{\ell+1}} k_0 Q_{\ell+1} + \epsilon'_0 \\
&\quad + \left(\frac{\Delta_\ell}{\Delta_{\ell+1}} c_1 + \frac{1}{q_{\ell+1}} \epsilon_1 + \frac{1}{q_{\ell+1}} k_1 Q_{\ell+1} + \epsilon'_1 \right) s + k Q_\ell \\
&= \frac{\Delta_\ell}{\Delta_{\ell+1}} (c_0 + c_1 s) + \frac{1}{q_{\ell+1}} (\epsilon_0 + \epsilon_1 s) + \epsilon'_0 + \epsilon'_1 s + (k_0 + k_1 s + k) Q_\ell \\
&= \Delta_\ell m + \frac{\Delta_\ell}{\Delta_{\ell+1}} \epsilon + \frac{1}{q_{\ell+1}} (\epsilon_0 + \epsilon_1 s) + \epsilon'_0 + \epsilon'_1 s + (k_0 + k_1 s + k) Q_\ell \,.
\end{aligned}$$

Thus the output ciphertext $\texttt{ct}_{\text{adj}}$ encrypts m at level ℓ with scale Δ_ℓ and noise

$$\epsilon_{\text{adj}} = \frac{\Delta_\ell}{\Delta_{\ell+1}} \epsilon + \frac{1}{q_{\ell+1}} \epsilon_{\text{round}} + \epsilon'_{\text{round}},$$

where ϵ, $\epsilon_{\text{round}} = \epsilon_0 + \epsilon_1 s$ and $\epsilon'_{\text{round}} = \epsilon'_0 + \epsilon'_1 s$ are independent random variables. We therefore have $\dfrac{\Delta_\ell}{\Delta_{\ell+1}} \epsilon \sim \mathrm{N}\left(\mathbf{0}; \dfrac{\Delta_\ell^2}{\Delta_{\ell+1}^2} \rho^2 I_N\right)$ and, by Lemma 18,

$$\frac{\epsilon_{\text{round}}}{q_{\ell+1}} \sim \mathrm{N}\left(\mathbf{0}; \frac{\frac{1}{12}(\|s\|_2^2 + 1)}{q_{\ell+1}^2} I_N\right) \quad \text{and} \quad \epsilon'_{\text{round}} \sim \mathrm{N}\left(\mathbf{0}; \left(\tfrac{1}{12}(\|s\|_2^2 + 1)\right) I_N\right) .$$

This shows that $\dfrac{\epsilon_{\text{round}}}{q_{\ell+1}}$ is a negligible random variable, giving

$$\epsilon_{\text{adj}} \sim \mathrm{N}(\mathbf{0}; \rho^2_{\text{adj}} I_N), \qquad \text{where } \rho^2_{\text{adj}} = \frac{\Delta^2_\ell}{\Delta^2_{\ell+1}} \rho^2 + \tfrac{1}{12}(\|s\|^2_2 + 1).$$

F.2 Proof of Lemma 12

Let $\mathtt{ct}'_1$ and $\mathtt{ct}'_2$ be the outputs of Rescale applied to $\mathtt{ct}_1$ and $\mathtt{ct}_2$ respectively. Then $\mathtt{ct}'_1$ encrypts $m'_1 := \left(\Delta^2_\ell / q_\ell\right) m_1 = \Delta_{\ell-1} m_1$ with noise ϵ'_1 and $\mathtt{ct}'_2$ encrypts $m'_2 := \left(\Delta^2_\ell / q_\ell\right) m_2 = \Delta_{\ell-1} m_2$ with noise ϵ'_2. Then $\mathtt{ct}_{\text{Mult}'}$ encrypts $m'_1 m'_2 = \left(\Delta^4_\ell / q^2_\ell\right) m_1 m_2 = \Delta^2_{\ell-1} \cdot m_1 m_2$ with noise $\epsilon_{\text{Mult}'}$. By Lemma 10, $\epsilon'_1 \sim N(\mathbf{0}, \rho^2_{\prime_1})$ where $\rho^2_{\prime_1} = \frac{\rho^2_1}{q^2_\ell} + \left(\frac{1}{18}N + \frac{1}{12}\right)$ and $\epsilon'_2 \sim N(\mathbf{0}, \rho^2_{\prime_2})$ where $\rho^2_{\prime_2} = \frac{\rho^2_2}{q^2_\ell} + \left(\frac{1}{18}N + \frac{1}{12}\right)$. Then, if the conditions of Lemma 17 hold, $\epsilon_{\text{Mult}'} \sim N(\mathbf{0}, \rho^2_{\text{Mult}'})$ where

$$\begin{aligned}
\rho^2_{\text{Mult}'} &= N\rho^2_{\prime_1}\rho^2_{\prime_2} + \rho^2_{\prime_2}\left\|m'_1\right\|^2_2 + \rho^2_{\prime_1}\left\|m'_2\right\|^2_2 \\
&= N\left(\frac{\rho^2_1}{q^2_\ell} + \frac{1}{18}N + \frac{1}{12}\right)\left(\frac{\rho^2_2}{q^2_\ell} + \frac{1}{18}N + \frac{1}{12}\right) \\
&\quad + \left(\frac{\rho^2_2}{q^2_\ell} + \frac{1}{18}N + \frac{1}{12}\right)\Delta_{\ell-1}\left\|m_1\right\|^2_2 \\
&\quad + \left(\frac{\rho^2_1}{q^2_\ell} + \frac{1}{18}N + \frac{1}{12}\right)\Delta_{\ell-1}\left\|m_2\right\|^2_2 \\
&= \frac{N\rho^2_1\rho^2_2}{q^4_\ell} + \left(\frac{N^2}{18q^2_\ell} + \frac{N}{12q_\ell}\right)(\rho^2_1 + \rho^2_2) + \frac{\Delta_{\ell-1}}{q^2_\ell}\left(\rho^2_2\left\|m_1\right\|^2_2 + \rho^2_1\left\|m_2\right\|^2_2\right) \\
&\quad + \left(\frac{\Delta_{\ell-1}N}{18} + \frac{\Delta_{\ell-1}}{12}\right)\left(\left\|m_1\right\|^2_2 + \left\|m_2\right\|^2_2\right) + \frac{N^3}{18} + \frac{N^2}{108} + \frac{N}{144}\,.
\end{aligned}$$