Improving and Automating BFV Parameters Selection: An Average-Case Approach

Beatrice Biasioli[1], Chiara Marcolla[1], Marco Calderini[2], and Johannes Mono[3]

[1] Technology Innovation Institute, Abu Dhabi, United Arab Emirates
[2] Universitá degli studi di Trento, Italy
[3] Ruhr University Bochum, Bochum, Germany

Abstract. The Brakerski/Fan-Vercauteren (BFV) scheme is a state-of-the-art scheme in Fully Homomorphic Encryption based on the Ring Learning with Errors (RLWE) problem. Thus, ciphertexts contain an error that increases with each homomorphic operation and has to stay below a certain threshold for correctness. This can be achieved by setting the ciphertext modulus big enough. On the other hand, a larger ciphertext modulus decreases the level of security and computational efficiency, making parameter selection challenging. Our work aims to improve the bound on the ciphertext modulus, minimizing it.
Our main contributions are the following. Primarily, we perform the first average-case analysis of the error growth for the BFV scheme, significantly improving its estimation. For a circuit with a multiplicative depth of only 5, our bounds are up to 25.2 bits tighter than previous analyses and within 1.2 bits of the experimentally observed values. Secondly, we give a general way to bound the ciphertext modulus for correct decryption that allows closed formulas. Finally, we use our theoretical advances and propose the first parameter generation tool for the BFV scheme. Here, we add support for arbitrary but use-case-specific circuits, as well as the ability to generate easy-to-use code snippets, making our theoretical work accessible to both researchers and practitioners.

Keywords: Fully Homomorphic Encryption, BFV, Parameter Generation, Average-Case Noise Analysis, OpenFHE

1 Introduction

Data privacy concerns are increasing significantly in the context of future-generation networking, such as Internet of Things, cloud services, edge computing and artificial intelligence applications. Homomorphic encryption enables privacy-preserving data processing, namely data manipulation in the encrypted domain without decryption. More specifically, fully homomorphic encryption (FHE) schemes allows ciphertext operations that correspond to additions and multiplications on the underlying plaintext.

The first Fully Homomorphic Encryption (FHE) scheme was introduced in 2009 by Gentry in [26]. In his PhD thesis, Gentry provided a method for constructing a general FHE scheme from a scheme with limited but sufficient homomorphic evaluation capacity. Since then, novel constructions on FHE have been proposed following his idea, BGV [8], BFV [7,24], TFHE [16,17] which improves the FHEW scheme [23], and CKKS [14,15] some of the most representative. The reader interested in FHE and its applications will find some introductory material in [1,13,34,35].

The security of most of the FHE schemes is based on the presumed intractability of the decision Learning with Errors (LWE) problem, [38], and its ring variant (RLWE), [33]. Informally, they consist of distinguishing equations perturbed by small error from random tuples. In the following, our focus will be on the ring version. Let $\mathcal{R}_q = \mathbb{Z}_q[x]/\langle x^n + 1\rangle$ where q is a positive integer. The decision RLWE problem consists of distinguishing with *non-negligible advantage* between independent and uniformly random samples in $\mathcal{R}_q \times \mathcal{R}_q$ and the same number of independent RLWE instances. These instances are represented as $(a, b = s \cdot a + e) \in \mathcal{R}_q \times \mathcal{R}_q$, where $a, s \in \mathcal{R}_q$ are randomly chosen and $e \in \mathcal{R}_q$ is the error sampled from a distribution χ.

The problem arising from this construction is that the error (also called noise) grows progressively as operations are performed, particularly when homomorphic multiplications are involved. In order to guarantee correct decryption, the error has to be small. Specifically, its maximal coefficient must be smaller than a quantity depending on the ciphertext modulus q. One approach to accommodating more operations is increasing the ciphertext modulus q. However, a larger modulus also decreases the security level of the underlying scheme. To maintain an equivalent level of security, we must require a larger polynomial degree n at the cost of efficiency. This delicate balance between security (achieved with a small ciphertext modulus) and error margin (associated with a large ciphertext modulus) illustrates the difficulty of finding an optimal set of parameters for a specific FHE scheme.

Addressing this challenge is crucial to reach widespread adoption of FHE. For this reason, the FHE community has made significant efforts by providing automated and user-friendly methods for choosing an appropriate set of parameters. For instance, Mono *et al.* [36] developed an interactive parameters generator for the leveled BGV scheme that supports arbitrary circuit models. Bergerat *et al.* [5] proposed a framework for efficiently selecting parameters in TFHE-like schemes. Moreover, for all FHE schemes, the Homomorphic Encryption Standard [3] provides lookup tables (recently updated in [6]) that allow to determine the maximum ciphertext modulus q required to achieve a desired security level λ given fixed polynomial degrees n. To provide λ, the Homomorphic Encryption Standard used the Lattice Estimator[4], a software tool to determine the security level of LWE instances against the known attacks. Finally, in [31], starting from a theoretical analysis of lattice attacks, Kirshanova *et al.* present closed and precise formulas for two key tasks: 1) deriving the security parameter λ given the

[4] https://github.com/malb/lattice-estimator.

secret distribution χ_s, the polynomial degree n, and the ciphertext modulus q, 2) determining n as a function of λ, q, and χ_s. Additionally, they offer a practical tool to compute these formulas, making their approach both rigorous and accessible.

It is worth noting that to achieve the goal of finding an optimal set of parameters is also essential to provide an accurate approximation of the noise size. Indeed, an underestimation not only results in an incorrect plaintext recovery but also renders the FHE schemes vulnerable to attacks, as evidenced by recent papers [10,12]. While a substantial overestimation would significantly affect security and efficiency.

Over the past few years, several methods to compute a bound for the error have been proposed, from the Euclidean [8] and infinity norms [24,30] to the canonical norm (called *worst-case* analysis) [18,20,27,29,36]. The prevailing trend in the current literature adopts the *average-case* analysis, where the coefficients of the polynomial error are treated as random variables. With this approach, it is possible to compute a tight *probabilistic* upper bound, taking into consideration the Gaussian distribution of the error coefficients, their mean, and variance. However, the state-of-the-art is limited to the case where the ciphertexts are computed independently. Firstly employed in the TFHE [16] scheme, it has been successively used for CKKS [19] and BGV [21,37].

In this context, we want to emphasize that applying these established average-case methods to the BFV scheme results in incorrect bounds. For this reason, the state-of-the-art still employs either the infinity [30] or the canonical norm [29,18,20]. In this paper, we propose the first average-case approach that successfully applies to the BFV, because it takes in account the dependency among its error coefficients. This yields correct, accurate and secure bound.

Our contributions. This paper presents three main contributions, significantly improving the current state of BFV parameters selection.

Firstly, we present an innovative and accurate approach for the noise analysis of the BFV scheme based on the average-case. Our method significantly differs from the previously proposed for the BGV [37] and CKKS schemes [19], as we take into consideration the fact that the error coefficients are not independent among each other, making it impossible to apply the Central Limit Theorem. As a result, our analysis is more intricate, particularly for homomorphic multiplication, where we have to introduce a function to "correct" the product of the variances. This results in more accurate and secure bounds. In contrast, the heuristics used in the BGV [37] and CKKS [19] schemes often *underestimate* the noise growth due to the assumption of noise coefficient independence, leading to *imprecise bounds*, as also pointed out in [19,37].

With our approach, we significantly improve the understanding of the noise growth arising from the encryption and homomorphic operations in BFV, offering exceptionally tight bounds. This improvement is validated by experimental results (Tables 7 to 9), where our bounds are exceptionally close, differing by no more than 2.5 bits from the experimentally observed values. The experiments

are done by computing the error growth in different circuits using OpenFHE [2] library, a well-known and used open-source FHE library. Moreover, we conduct a comprehensive comparison of our bounds with the state-of-the-art noise analysis based on the canonical norm. Notably, for a circuit with multiplicative depth only 5, our bounds are up to 25.2 bits tighter than the previous one, showing the substantial improvement our approach brings to noise analysis.

Secondly, we provide closed formulas to compute a minimal bound on the ciphertext modulus that guarantees correct decryption. Specifically, we introduce a method for the computation of the ciphertext modulus in any circuit. Moreover, we focus on the most common ones illustrated in Figure 3, for which we explicitly provide closed formulas for q. Thanks to our findings, we compare our results with those obtained using the worst-case approach. This highlights the importance of having a tighter bound for the error. For instance, for a simple circuit with multiplicative depth 3, the ciphertext modulus decreases by at least 13.5% (Figure 4a).

In addition, we develop an interactive parameters generator, which makes use of our theoretical results and the security formula proposed in [36]. This tool provides flexibility, allowing users to choose the desired security level, the degree of the arithmetic function to be evaluated homomorphically, and the error and secret distributions, among other parameters. This user-friendly tool is designed to be accessible to individuals with different levels of expertise, ensuring that even those who are not FHE experts can utilize it. By providing an accessible mean for generating parameters tailored to the BFV scheme, we aim to contribute to the widespread adoption of FHE.

Finally, we conduct the first study of specific circuits with *dependent* ciphertexts. In addition to our comparison of dependent and independent cases, this analysis establishes a groundwork for understanding error bounds and highlights the significant differences between the two scenarios. In particular, when comparing Figure 7b (dependent-computed ciphertexts) with Figure 4b (independent case) for a simple circuit with a multiplicative depth of 5, we observe at least a 5.3% increase in the ciphertext modulus. These findings point out the critical importance of selecting the parameters according to the correct case to ensure both correctness and security. Specifically, using parameters optimized for independent ciphertexts in the dependent case can result in decryption failures and thus expose the system to key recovery attacks [12].

The structure of the paper is the following:

- To facilitate the understanding of the paper, we present the notation and mathematical background required in Section 2.
- In Section 3, we comprehensively analyze and compute invariant noise after any operation in the BFV scheme.
- The core of the paper is Section 4, where we introduce our average-case approach.
- In Section 5, we exploit the novel error analysis to provide a general way to compute a minimal bound on the ciphertext modulus, focusing on practical-

used circuits. Additionally, we introduce our parameter generator to facilitate the selection of optimal parameters for the BFV scheme.
– In Section 6, we compare our average-case approach with prior bounds of BFV noise growth as well as with experimental results.
– In Section 7, we explore a general setting where ciphertexts can be dependently computed. This section establishes a baseline for understanding error bounds in this context and highlights the differences with the independent case.
– Finally, Section 8 draws some conclusions and open problems.

2 Preliminaries

In this section, we first define the general notations that we will use in the remainder of the work, then we provide the mathematical background for the secret and error distributions, as well as their analysis.

2.1 Notation

Let $f(x)$ be a monic irreducible polynomial of degree n, in particular, we take $f(x) = x^n + 1$ with n a power of 2. We denote by $\mathcal{R} = \mathbb{Z}[x]/\langle f(x)\rangle$ and with $\mathcal{K} = \mathbb{Q}[x]/\langle f(x)\rangle$. Let $a \in \mathcal{K}$, we denote by $a|_i$ the coefficient of x^i. Note that, for $a, b \in \mathcal{K}$ we have that

$$(ab)|_i = \sum_{j=0}^{n-1} \xi(i,j)\, a|_j\, b|_{i-j}, \tag{1}$$

where $i-j$ is computed mod n and $\xi(i,j)$ is defined as 1 if $i-j \in [0,n)$ and -1 otherwise. For a positive integer p, $\mathbb{Z}_p$ denotes the set of integers in $(-p/2, p/2]$ and by $\mathcal{R}_p$ the set of polynomials in $\mathcal{R}$ with coefficients in $\mathbb{Z}_p$. Let $z \in \mathbb{Z}$, we write $[z]_p \in \mathbb{Z}_p$ for the centered representative of z mod p. For polynomials in $\mathcal{R}$, it denotes the element in $\mathcal{R}_p$ where $[\cdot]_p$ is applied coefficient-wise. Let $x \in \mathbb{Q}$, $\lfloor x \rceil$ be the rounding to the nearest integer. The same holds coefficient-wise for polynomials in $\mathcal{K}$.

The integer $t > 1$ denotes the plaintext modulus and with $\mathcal{R}_t$ the plaintext space. We further require $t \equiv 1 \pmod{2n}$. Analogously, we denote the ciphertext modulus by $q = \prod_{i=1}^{k} r_i$, and the ciphertext space follows as $\mathcal{R}_q$. $r_i > 1$ are pairwise coprime of approximately the same size, coprime with t and such that $r_i \equiv 1$ mod $2n$. Moreover, for the BGV-like circuit case explained in Section 5.2, we need $L = M + 1$ sub-moduli p_j defined analogously to q, where M is the multiplicative depth of the circuit. For any ℓ, we denote by $q_\ell = \prod_{j=1}^{\ell} p_j = \prod_{i=1}^{k_\ell} r_i$, the initial ciphertext is $q = q_L$, or q_{ms} to distinguish it.

Finally, we recall the *characteristic function* of a subset A is defined as

$$\mathbf{1}_{x\in A} = \begin{cases} 1 & \text{if } x \in A \ , \\ 0 & \text{if } x \notin A \ . \end{cases}$$

Note that if $A = \{a\}$, we can also denoted the characteristic function as $\mathbf{1}_{x=a}$

2.2 Secret and Error Distributions

Let χ be a probabilistic distribution and $a \in \mathcal{R}$, we write $a \leftarrow \chi$ when sampling each coefficient of a independently from χ. We use the following distributions.

- $\mathcal{DG}(0, \sigma^2)$, the discrete Gaussian distribution centered in 0 with standard deviation σ.
- $\mathcal{U}_p$, the uniform distribution over $\mathbb{Z}_p$, where p is a positive integer.
- $\mathcal{U}_I$, the uniform distribution over a real interval $I \subset \mathbb{R}$.
- $\mathcal{ZO}(\rho)$, a distribution over the ternary set $\{0, \pm 1\}$ with probability $\rho/2$ for ± 1 and probability $1 - \rho$ for 0 with $\rho \in [0, 1]$.

Finally, the distributions $\mathcal{HWT}(h)$ chooses a vector uniformly at random from $\{0, \pm 1\}^n$ with exactly h nonzero entries, where $h \leq n$ positive integer. Let χ_s, χ_u be secret key distributions, χ_e an error distribution from the Learning with Errors over Rings (RLWE) problem and V_s, V_u, V_e the associated variances. Tipically, we have $\chi_e = \mathcal{DG}(0, \sigma^2)$, with $\sigma = 3.19$ and $\chi_s = \chi_u = \mathcal{U}_3$ [3]. Other common options for χ_s are $\mathcal{ZO}(0.5)$, $\mathcal{DG}(0, (3.19)^2)$ and $\mathcal{HWT}(64)$. A variable with any of the above distributions or from the uniform over a centered interval is symmetric, thus with mean 0, and has variance as follows.

- If $X \leftarrow \mathcal{DG}(0, \sigma^2)$ then $\mathsf{Var}(X) = \sigma^2$.
- If $X \leftarrow \mathcal{U}_p$ then $\mathsf{Var}(X) = (p^2 - 1)/12$. In particular,
  - If $X \leftarrow \mathcal{U}_q$ then $\mathsf{Var}(X) \approx q^2/12$.
  - If $X \leftarrow \mathcal{U}_3$ then $\mathsf{Var}(X) = 2/3$.
- If $X \leftarrow \mathcal{U}_{(-1/2,1/2]}$ then $\mathsf{Var}(X) = 1/12$.
- If $X \leftarrow \mathcal{ZO}(0.5)$ then $\mathsf{Var}(X) = 1/2$.
- If $X \leftarrow \mathcal{HWT}(64)$ then $\mathsf{Var}(X) = 64/n$.

2.3 Properties of probabilistic operators

In the following statement, we recall some useful properties of the most common probabilistic operators.

Fact 1 *Let $\mathbb{E}$ be the expected value, Var be the variance, and Cov be the covariance. Let a, b be constants and X, Y, Z, W be random variables. Then*

(a) The expected value $\mathbb{E}$ is linear: $\mathbb{E}[X + Y] = \mathbb{E}[X] + \mathbb{E}[Y]$, $\mathbb{E}[aX] = a\mathbb{E}[X]$.
(b) $\mathbb{E}$ is monotonic: if $X \leq Y$ (a.s.) and $\mathbb{E}[X], \mathbb{E}[Y]$ exist, then $\mathbb{E}[X] \leq \mathbb{E}[Y]$.
(c) $\mathsf{Cov}(X, Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$.
(d) If X and Y are independent, then $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$, i.e. $\mathsf{Cov}(X, Y) = 0$.

(e) *The covariance is bilinear:* $\mathsf{Cov}(aX + bY, Z) = a\mathsf{Cov}(X, Z) + b\mathsf{Cov}(Y, Z)$
(f) Cov *is symmetric:* $\mathsf{Cov}(X, Y) = \mathsf{Cov}(Y, X)$.
(g) *If* $X, Y \perp\!\!\!\perp Z, W$, $\mathsf{Cov}(X, Y) = 0$ *and* $\mathbb{E}[X] = 0$, *then* $\mathsf{Cov}(XZ, YW) = 0$.
Indeed, by (c) and (d), we have

$$\begin{aligned}\mathsf{Cov}(XZ, YW) =& \mathbb{E}[XYZW] - \mathbb{E}[XZ]\mathbb{E}[YW] \\ =& \mathbb{E}[XY]\mathbb{E}[ZW] - \mathbb{E}[X]\mathbb{E}[Z]\mathbb{E}[YW] \\ =& \mathbb{E}[XY]\mathbb{E}[ZW] = (\mathbb{E}[X]\mathbb{E}[Y] + \mathsf{Cov}(X, Y))\mathbb{E}[ZW] = 0\end{aligned}$$

(h) *If* $X \perp\!\!\!\perp Y, Z, W$ *and* $\mathbb{E}[X] = 0$, *then* $\mathsf{Cov}(XZ, YW) = 0$.
Analogously to (g), we have

$$\begin{aligned}\mathsf{Cov}(XZ, YW) =& \mathbb{E}[XYZW] - \mathbb{E}[XZ]\mathbb{E}[YW] \\ =& \mathbb{E}[X](\mathbb{E}[YZW] - \mathbb{E}[Z]\mathbb{E}[YW]) = 0\end{aligned}$$

(i) $\mathsf{Var}(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2$, $\mathsf{Var}(X) \geq 0$ *and* $V(X) = \mathsf{Cov}(X, X)$.
(j) $\mathsf{Var}(X + a) = \mathsf{Var}(X)$ *and* $\mathsf{Var}(aX) = a^2\mathsf{Var}(X)$.
(k) $\mathsf{Var}(X + Y) = \mathsf{Var}(X) + \mathsf{Var}(Y) + 2\mathsf{Cov}(X, Y)$, *in general* $\mathsf{Var}(\sum_i X_i) = \sum_{i=1}^{N} \mathsf{Var}(X_i) + \sum_{i_1 \neq i_2} \mathsf{Cov}(X_{i_1}, X_{i_2})$.
(l) $\mathsf{Var}(XY) = (\mathsf{Var}(X) + \mathbb{E}[X]^2)(\mathsf{Var}(Y) + \mathbb{E}[Y]^2) + \mathsf{Cov}(X^2, Y^2) - (\mathsf{Cov}(X, Y) + \mathbb{E}[X]\mathbb{E}[Y])^2$, *in particular*
  – *if* $\mathbb{E}[X] = \mathbb{E}[Y] = 0$, $\mathsf{Var}(XY) = \mathsf{Var}(X)\mathsf{Var}(Y) + \mathsf{Cov}(X^2, Y^2) - \mathsf{Cov}(X, Y)^2$
  – *if* $\mathbb{E}[X] = \mathbb{E}[Y] = 0$ *and* X *and* Y *are independent,* $\mathsf{Var}(XY) = \mathsf{Var}(X)\mathsf{Var}(Y)$
(m) *If* $\mathsf{Var}(X), \mathsf{Var}(Y)$ *are finite, then* $|\mathsf{Cov}(X, Y)| \leq \sqrt{\mathsf{Var}(X)\mathsf{Var}(Y)}$.

Coverage probability for Gaussian-distributed variables. Let X be a random variable (r.v.) from a Gaussian distribution centred in 0 of variance V, then

$$\begin{aligned}\mathbb{P}\Big(|X| \leq x\Big) &= \mathbb{P}\Big(X \leq x\Big) - \mathbb{P}\Big(X \leq -x\Big) = \\ &= \frac{1}{2}\Big(1 + \mathrm{erf}\Big(\frac{x}{\sqrt{2V}}\Big)\Big) - \frac{1}{2}\Big(1 + \mathrm{erf}\Big(\frac{-x}{\sqrt{2V}}\Big)\Big) = \mathrm{erf}\Big(\frac{x}{\sqrt{2V}}\Big).\end{aligned} \tag{2}$$

Suppose now that we want to study the infinity norm of a vector. If its entries are independent, then $\mathbb{P}\Big(||\mathbf{X}||_\infty \leq x\Big) = \mathbb{P}\Big(|X| \leq x\Big)^n$. In general, we can give an upper bound on the complementary probability:

$$\mathbb{P}\Big(||\mathbf{X}||_\infty > x\Big) \leq n\mathbb{P}\Big(|X| > x\Big) = n\Big(1 - \mathrm{erf}\Big(\frac{x}{\sqrt{2V}}\Big)\Big). \tag{3}$$

Canonical embedding and norm. We recall the results of [18,29,20]. The *canonical embedding* of $a \in \mathcal{R}$ is the vector obtained by evaluating a in the primitive $2n$-th roots of unity. The *canonical embedding norm* of a is defined as the infinity norm of the canonical embedding.

Let us consider a random polynomial $a \in R$ where each coefficient is sampled independently from a zero-mean distribution, then $||a||^{can} \leq D\sqrt{nV_a}$ with high probability [18].

We now want to estimate the probability that the canonical norm of a random polynomial exceeds a certain value x.

Let us consider the case where the coefficients in a, $a|_0, ..., a|_{n-1}$, are independent and identically distributed (i.i.d.) with 0 mean and variance V_a, and suppose $\mathbb{E}(|a|_i|^{2+\delta}) < \infty$ for all i and for some fixed $\delta > 0$ (this last condition it is not restrictive in our case). As shown in [22], using the Lyapunov Central Limit Theorem, it is possible to prove that for any root of unity $\zeta = \cos(\alpha) + i\sin(\alpha)$, the r.v. $a(\zeta)$ is a complex r.v. which can be approximated by a complex Gaussian r.v.. That is, $a(\zeta)$ is approximated by a bivariate Normal distributed r.v. (X, Y). Moreover, X and Y are Normal distributed with variance $V_X = V_a(\sum_{j=0}^{n-1}\cos^2(j\alpha))$ and $V_Y = V_a(\sum_{j=0}^{n-1}\sin^2(j\alpha)) = nV_a - V_X$, respectively.

Let C be the diagonal matrix with the standard deviation of X and Y over the diagonal. We have that $(X, Y)^t = C \cdot (Z, Z')^t$ with Z and Z' i.i.d. standard Gaussian r.v.'s. Therefore,

$$\mathbb{P}\left(|a(\zeta)| < x\right) = \mathbb{P}\left(||(X, Y)||_2 < x\right) \geq \mathbb{P}\left(||C||_2||(Z, Z')||_2 < x\right).$$

Let M be the maximum between V_X and V_Y (note that $\frac{n}{2}V_a \leq M \leq nV_a$). The 2-norm of the matrix C is $\sqrt{M}$. Thus, $\mathbb{P}\left(||C||_2||(Z, Z')||_2 < x\right) = \mathbb{P}\left(||(Z, Z')||_2^2 < \frac{x^2}{M}\right)$. Since Z, Z' are independent standard Gaussian r.v., $||(Z, Z')||_2^2$ is Chi-squared distributed and $\mathbb{P}\left(||(Z, Z')||_2^2 < \frac{x^2}{M}\right) = 1 - e^{-\frac{x^2}{2M}} \geq 1 - e^{-\frac{x^2}{nV_a}}$, implying $\mathbb{P}\left(|a(\zeta_m)| > x\right) \leq e^{-\frac{x^2}{nV_a}}$. Therefore,

$$\mathbb{P}\left(||a||^{can} > x\right) \leq ne^{-\frac{x^2}{nV_a}}. \tag{4}$$

3 The BFV Scheme

The following describes the BFV scheme [7,24], a cutting-edge FHE scheme whose security relies on the hardness of the ring learning with errors (RLWE) problem. We consider the latest enhancements proposed in [30]. In particular, the authors revised the encryption algorithm replacing the term $\Delta m = \lfloor \frac{q}{t} \rfloor m$ with $\lfloor \frac{q}{t} m \rceil$, which eliminates the noise gap with respect to the BGV scheme.

KeyGen(λ, L)

Define parameters and distributions accordingly to λ and L. Sample $s \leftarrow \chi_s$, $a \leftarrow \mathcal{U}_q$ and $e \leftarrow \chi_e$. Output $\mathsf{sk} = s$ and $\mathsf{pk} = (b, a) = ([-as + e]_q, a)$.

Enc(m, pk)

Receive the plaintext $m \in \mathcal{R}_t$ and $\mathsf{pk} = (b, a)$. Sample $u \leftarrow \chi_u$ and $e_0, e_1 \leftarrow \chi_e$. Output $\mathfrak{c} = (\mathbf{c}, q, \nu_{\mathsf{clean}})$ with $\mathbf{c} = (c_0, c_1) = \left(\left[\lfloor \frac{q}{t} m \rceil + ub + e_0\right]_q, [ua + e_1]_q\right)$.

Dec($\mathfrak{c}$, sk)

Receive the extended ciphertext $\mathfrak{c}$ for $\mathsf{sk} = s$. Output $\left[\left\lfloor \frac{t}{q_\ell}[c_0 + c_1 s]_{q_\ell} \right\rceil\right]_t$.

Let $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$ be the *extended ciphertext*, where $\mathbf{c}$ is a ciphertext, q_ℓ denotes the ciphertext modulus and ν the *invariant noise*. The invariant noise [29] is the minimal $\nu \in \mathcal{K}$ such that

$$\frac{t}{q_\ell}[c_0 + c_1 s]_{q_\ell} = m + \nu + kt$$

for some $k \in \mathcal{R}$. Therefore, $\left[\left\lfloor \frac{t}{q}[c_0 + c_1 s]_q \right\rceil\right]_t = [\lfloor m + \nu + kt \rceil]_t = [m + \lfloor \nu \rceil]_t$. Hence, the decryption works properly as long as ν is small enough. In particular, it is correct when the coefficients of ν belong to the interval $(-\frac{1}{2}, \frac{1}{2}]$. After the encryption operation, the invariant noise is

$$\nu_{\mathsf{clean}} = \frac{t}{q}(\varepsilon + eu + e_0 + e_1 s) \tag{5}$$

where $\varepsilon = \left\lfloor \frac{q}{t} m \right\rceil - \frac{q}{t} m = -\frac{[qm]_t}{t}$, [30].

Addition & Constant Multiplication.

Add($\mathfrak{c}, \mathfrak{c}'$)

Receive extended ciphertexts $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$ and $\mathfrak{c}' = (\mathbf{c}', q_\ell, \nu')$.
Output $(\mathbf{c}_{\mathsf{add}}, q_\ell, \nu_{\mathsf{add}})$ with $\mathbf{c}_{\mathsf{add}} = ([c_0 + c'_0]_{q_\ell}, [c_1 + c'_1]_{q_\ell})$.

MulConst($\alpha, \mathfrak{c}$)

Receive constant polynomial $\alpha \in \mathcal{R}_t$ and extended ciphertext $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$.
Output $(\mathbf{c}_{\mathsf{const}}, q_\ell, \nu_{\mathsf{const}})$ with $\mathbf{c}_{\mathsf{const}} = ([\alpha c_0]_{q_\ell}, [\alpha c_1]_{q_\ell})$.

By the definition of invariant noise, for some $k \in \mathcal{R}$, we have

$$\frac{t}{q_\ell}[c_0 + c_1 s + c'_0 + c'_1 s]_{q_\ell} = [m + m']_t + \nu + \nu' + kt \implies \nu_{\mathsf{add}} = \nu + \nu' \tag{6}$$

$$\frac{t}{q_\ell}[\alpha c_0 + \alpha c_1 s]_{q_\ell} = [\alpha m]_t + \alpha\nu + kt \implies \nu_{\mathsf{const}} = \alpha\nu, \tag{7}$$

Multiplication & Modulus switching. In this section, we are going to see the multiplication algorithm presented in [30], which, before multiplying two ciphertexts, applies to one of them a modulus switch. This is done in order to make the Residue Number System (RNS) representation more efficient. The modulus switch technique was first introduced for the BGV scheme in [9] to reduce the error associated with a ciphertext. In the BFV scheme, this error reduction is made implicitly, so the purpose of the modulus switch is only to shift to a different ciphertext modulus.

ModSwitch$(\mathfrak{c}, q'_\ell)$

Receive the extended ciphertext $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$ and the target modulo q'_ℓ. Output $\mathfrak{c}' = (\mathbf{c}', q'_\ell, \nu + \nu_{\mathsf{ms}}(q'_\ell))$ with $\mathbf{c}' = \left(\left[\left\lfloor \frac{q'_\ell}{q_\ell} c_0 \right\rceil \right]_{q'_\ell}, \left[\left\lfloor \frac{q'_\ell}{q_\ell} c_1 \right\rceil \right]_{q'_\ell} \right)$.

The noise added by the modulo switch operation is

$$\nu_{\mathsf{ms}}(q'_\ell) = \frac{t}{q'_\ell}(\varepsilon_0 + \varepsilon_1 s), \text{ with } \varepsilon_i = -\frac{[q'_\ell c_i]_{q_\ell}}{q_\ell}. \tag{8}$$

The multiplication algorithm takes as input two extended ciphertexts $\mathfrak{c}$ and $\mathfrak{c}'$, where one of the ciphertexts, say $\mathfrak{c}'$, is the result of a modulo switch to q'_ℓ. The new modulus q'_ℓ is required to be of approximately the same size of q_ℓ, to satisfy $q'_\ell \equiv 1 \pmod{2n}$ and $(t, q'_\ell) = (q_\ell, q'_\ell) = 1$.

Ten$(\mathfrak{c}, \mathfrak{c}')$

Receive the extended ciphertexts $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$ and $\mathfrak{c}' = (\mathbf{c}', q'_\ell, \nu')$. Output $\mathfrak{d} = (\mathbf{d}, q_\ell, \nu_{\mathsf{mul}}(q_\ell))$ with

$$\mathbf{d} = (d_0, d_1, d_2) = \left(\left[\left\lfloor \frac{t}{q'_\ell} c_0 c'_0 \right\rceil \right]_{q_\ell}, \left[\left\lfloor \frac{t}{q'_\ell} (c_0 c'_1 + c_1 c'_0) \right\rceil \right]_{q_\ell}, \left[\left\lfloor \frac{t}{q'_\ell} c_1 c'_1 \right\rceil \right]_{q_\ell} \right).$$

The multiplication output is a polynomial $\mathcal{R}_q^3$ that can be decrypted in the following way: $\left\lfloor \frac{t}{q_\ell} [d_0 + d_1 s + d_2 s^2]_{q_\ell} \right\rceil$. Let $\frac{t}{q_\ell}(c_0 + c_1 s) = m + \nu + ht$ and $\frac{t}{q'_\ell}(c'_0 + c'_1 s) = m' + \nu' + h't$, as per definition of invariant noise. Thus,

$$\begin{aligned}
&\tfrac{t}{q_\ell} \left[\left\lfloor \tfrac{t}{q'_\ell} c_0 c'_0 \right\rceil + \left\lfloor \tfrac{t}{q'_\ell} (c_0 c'_1 + c'_0 c_1) \right\rceil s + \left\lfloor \tfrac{t}{q'_\ell} c_1 c'_1 \right\rceil s^2 \right]_{q_\ell} \\
&= \tfrac{t}{q_\ell}(c_0 + c_1 s) \cdot \tfrac{t}{q'_\ell}(c'_0 + c'_1 s) + \tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2) + h''t \\
&= [mm']_t + \nu(m' + h't) + \nu'(m + ht) + \nu\nu' + \tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2) + kt \\
&= [mm']_t + \nu_{\mathsf{mul}}(q_\ell) + kt,
\end{aligned}$$

where the noise after the multiplication is

$$\nu_{\mathsf{mul}}(q_\ell) = -\nu\nu' + \nu \frac{t}{q'_\ell}(c'_0 + c'_1 s) + \nu' \frac{t}{q_\ell}(c_0 + c_1 s) + \frac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2). \tag{9}$$

Finally, the multiplication output needs to be transformed back to a ciphertext in $\mathcal{R}_q^2$. This is done by encrypting its last term d_2 via key switching (see Section 3.1), also called relinearization.

3.1 Key Switching

The key switch is used for (i) reducing the degree of a ciphertext polynomial, usually the multiplication output, or (ii) changing the key after a rotation. In the multiplication case, the term $d_2 \cdot s^2$ is converted into a polynomial $c_0^{\mathsf{ks}} + c_1^{\mathsf{ks}} \cdot s$ and the two components are added, obtaining the equivalent $\mathbf{c}' = (d_0 + c_0^{\mathsf{ks}}, d_1 + c_1^{\mathsf{ks}})$.

In the rotation, where we need to go back to the original key s from $\mathrm{rot}(s)$, we convert the ciphertext term $c_1 \cdot \mathrm{rot}(s)$ into $c_0^{\mathsf{ks}} + c_1^{\mathsf{ks}} \cdot s$. In the following, we will only analyze the first case.

The idea is to encrypt the extra term s^2 under the secret key. However, in doing so, the resulting error would be too significant. Hence several variants exist to reduce its growth. This work considers the three main ones: Brakerski Vaikuntanathan (BV), Gentry Halevi Smart (GHS), and Hybrid. For the sake of simplicity, we present directly the variants compatible with the RNS representation [4,28,30]. The RNS method makes the scheme implementation substantially faster and allows parallelization.

Brakerski-Vaikuntanathan The strategy is exploiting the Chinese Remainder Theorem (CRT) to decompose d_2 in the k_ℓ moduli $r_i \approx \sqrt[k]{q}$.

KeySwitchGen$^{\mathsf{BV}}$ (s, s^2)

Sample $a_i \leftarrow \mathcal{U}_q$, $e_i \leftarrow \chi_e$ and set $(b_i, a_i) = \left(\left[\left[\left(\frac{q}{r_i} \right)^{-1} \right]_{r_i} \frac{q}{r_i} s^2 - a_i s + e_i \right]_q, a_i \right)$ for $i = 1, \ldots, k$. Output $\mathsf{ks}^{\mathsf{BV}} = \{(b_i, a_i)\}$.

KeySwitch$^{\mathsf{BV}}$$(\mathbf{ks}^{\mathsf{BV}}, \mathfrak{c})$

Receive $\mathfrak{d} = (\mathbf{d}, q_\ell, \nu)$ with $\mathbf{d} = (d_0, d_1, d_2)$ and $\mathsf{ks}^{\mathsf{BV}} = \{(b_i, a_i)\}$. Output $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu + \nu_{\mathsf{ks}}^{\mathsf{BV}}(q_\ell))$ where $\mathbf{c} = \left(\left[d_0 + \sum_{i=1}^{k_\ell} [d_2]_{r_i} b_i \right]_{q_\ell}, \left[d_1 + \sum_{i=1}^{k_\ell} [d_2]_{r_i} a_i \right]_{q_\ell} \right)$.

The error after the BV key switching is $\nu + \nu_{\mathsf{ks}}^{\mathsf{BV}}(q_\ell)$ where

$$\nu_{\mathsf{ks}}^{\mathsf{BV}}(q_\ell) = \frac{t}{q_\ell} \sum_{i=1}^{k_\ell} [d_2]_{r_i} e_i. \tag{10}$$

Gentry-Halevi-Smart An alternative is encrypting Ps^2 instead of s^2 with P a large number, usually of approximately the same size as q. In this way, the error quantity added is divided by P.

KeySwitchGen$^{\mathsf{GHS}}$$(s, s^2)$

Sample $a' \leftarrow \mathcal{U}_{qP}$, $e' \leftarrow \chi_e$ and output the key switching key

$$\mathsf{ks}^{\mathsf{GHS}} = (b', a') = ([Ps^2 - a's + e']_{qP}, a').$$

KeySwitch$^{\mathsf{GHS}}$$(\mathbf{ks}, \mathfrak{c})$

Receive extended ciphertext $\mathfrak{d} = (\mathbf{d}, q_\ell, \nu)$ and key switching key $\mathsf{ks}^{\mathsf{GHS}}$.
Output $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu + \nu_{\mathsf{ks}}^{\mathsf{GHS}}((q_\ell))$ with $\mathbf{c} = \left(\left[d_0 + \left\lfloor \frac{d_2 b'}{P} \right\rceil \right]_{q_\ell}, \left[d_1 + \left\lfloor \frac{d_2 a'}{P} \right\rceil \right]_{q_\ell} \right)$.

The noise after the GHS key switching is $\nu + \nu_{\mathsf{ks}}^{\mathsf{GHS}}(q_\ell)$ where

$$\nu_{\mathsf{ks}}^{\mathsf{GHS}}(q_\ell) = \frac{t}{q_\ell} \left(\frac{d_2 e'}{P} + \varepsilon_0 + \varepsilon_1 s \right). \tag{11}$$

GHS-RNS In practice, d_2 in base $q_\ell P$ is computed with the `FastBaseExtension` technique [30], for better efficiency, which gives an approximate result $d_2 + uq_\ell$:

$$\sum_{i=1}^{k_\ell} \left[[d_2]_{r_i} \left[\left(\frac{q_\ell}{r_i} \right)^{-1} \right]_{r_i} \right]_{r_i} \frac{q_\ell}{r_i} = d_2 + uq_\ell, \quad u = \left\lfloor \sum_{i=1}^{k_\ell} \left[[d_2]_{r_i} \left[\left(\frac{q_\ell}{r_i} \right)^{-1} \right]_{r_i} \right]_{r_i} \frac{1}{r_i} \right\rceil.$$

Therefore, the added error becomes

$$\nu_{\mathsf{ks}}^{\mathsf{GHS-RNS}}(q_\ell) = \frac{t}{q_\ell} \left(\frac{(d_2 + uq_\ell)e'}{P} + \varepsilon_0 + \varepsilon_1 s \right). \tag{12}$$

Hybrid The Hybrid variant offers a trade-off between efficiency and security from the two previous variants. Indeed, the downside of the first one is the inefficiency due to a large number of multiplications to be performed. In contrast, the issue with the second one is that its security relies on the RLWE assumption with a larger factor $q_\ell P$, instead of q_ℓ. This larger factor means that to achieve the same level of security, the modulus q_ℓ must be smaller, which limits the depth of the circuit that can be evaluated homomorphically. In the Hybrid relinearization, the modulus is split in a smaller number of elements ω by gathering the r_i in chunks $\tilde{r_i}$, and the division is done considering $P \approx \sqrt[\omega]{q}$. For further information see [30,27].

KeySwitchGen$^{\mathsf{Hyb}}(s, s^2)$

Sample $a_i \leftarrow \mathcal{U}_{qP}$, $e_i \leftarrow \chi_e$ and output $\mathsf{ks}^{\mathsf{Hyb}} = \{(b_i, a_i)\}_{i=1,\dots,\omega}$ with

$$(b_i, a_i) = \left(\left[P \left[\left(\frac{q}{\tilde{r_i}} \right)^{-1} \right]_{\tilde{r_i}} \frac{q}{\tilde{r_i}} s^2 - a_i s + e_i \right]_{qP}, a_i \right).$$

KeySwitch$^{\mathsf{Hyb}}(\mathbf{ks}^{\mathsf{Hyb}}, \mathfrak{c})$

Receive extended ciphertext $\mathfrak{d} = (\mathbf{d}, q_\ell, \nu)$ and key switching key $\mathsf{ks}^{\mathsf{Hyb}}$.
Output $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu + \nu_{\mathsf{ks}}^{\mathsf{Hyb}}(q_\ell))$ with

$$\mathbf{c} = \left(\left[d_0 + \left\lfloor \frac{\sum_{i=1}^{\omega} [d_2]_{\tilde{r_i}} b_i}{P} \right\rceil \right]_{q_\ell}, \left[d_1 + \left\lfloor \frac{\sum_{i=1}^{\omega} [d_2]_{\tilde{r_i}} a_i}{P} \right\rceil \right]_{q_\ell} \right).$$

The noise after the Hybrid key switching is $\nu + \nu_{\mathsf{ks}}^{\mathsf{Hyb}}(q_\ell)$, where

$$\nu_{\mathsf{ks}}^{\mathsf{Hyb}}(q_\ell) = \frac{t}{q_\ell} \left(\frac{\sum_{i=1}^{\omega} [d_2]_{\tilde{r_i}} e_i}{P} + \varepsilon_0 + \varepsilon_1 s \right). \tag{13}$$

Hyb-RNS Here, the FastBaseExtension is eventually applied to the terms $[d_2]_{\tilde{r_i}}$,

$$\sum_{r_j | \tilde{r_i}} \left[[d_2]_{r_j} \left[\left(\frac{\tilde{r_i}}{r_j} \right)^{-1} \right]_{r_j} \right]_{r_j} \frac{\tilde{r_i}}{r_j} = [d_2]_{\tilde{r_i}} + u_i \tilde{r_i}, \; u_i = \left\lfloor \sum_{r_j | \tilde{r_i}} \left[[d_2]_{r_j} \left[\left(\frac{\tilde{r_i}}{r_j} \right)^{-1} \right]_{r_j} \right]_{r_j} \frac{1}{r_j} \right\rceil.$$

Therefore, the error added becomes

$$\nu_{\mathsf{ks}}^{\mathsf{Hyb-RNS}}(q_\ell) = \frac{t}{q_\ell} \left(\frac{\sum_{i=1}^{\omega} ([d_2]_{\tilde{r_i}} + u_i \tilde{r_i}) e_i}{P} + \varepsilon_0 + \varepsilon_1 s \right). \tag{14}$$

4 Average-Case Noise Analysis for BFV

The purpose of this section is to investigate the error behaviour during homomorphic operations among independently computed ciphertexts. The goal is to find a small ciphertext modulus ensuring correct decryption. More specifically, it has to make the error coefficients lie in $(-\frac{1}{2}, \frac{1}{2}]$ with overwhelming probability.

We observed that the distributions of the noise coefficients are well-approximated by identical distributed Gaussian centred in 0, but not independent. Therefore, we can bound the maximum error coefficient in absolute value with high probability just by limiting their variance V. In particular, by Equation (3), setting $V \leq \frac{1}{8D^2}$, i.e. $D \leq \frac{1}{2\sqrt{2V}}$, the probability of failure for the decryption is

$$\mathbb{P}\Big(||\nu||_\infty > \frac{1}{2}\Big) \leq n\Big(1 - \mathrm{erf}\big(\frac{1}{2\sqrt{2V}}\big)\Big) \leq n(1 - \mathrm{erf}(D)),$$

Usually $D = 6$. So, for example, for $n = 2^{13}$, we have $n(1 - \mathrm{erf}(D)) \approx 2^{-42}$.

In the following, we denote with ν the invariant noise of any ciphertext and with $\nu|_i$ the i-th coefficient of ν. Moreover, we indicate with a_ι the ι-th element of ν when written as a polynomial in s, i.e. $\nu = \sum_\iota a_\iota s^\iota$. Note that the element a_ι is a polynomial in $\mathcal{K}$ itself, then $a_\iota|_i$ is its i-th coefficient. Finally, the ciphertexts we are considering are computed independently. In other words, each time we perform addition and multiplication operations, we use ciphertexts that either encrypt two different messages or are the results of different circuits, and there are no shared messages between them.

4.1 Distribution Analysis

Our study of coefficients distribution has been performed computationally. We used the OpenFHE library [2] to compute 10000 error samples, then analysed their coefficients with the Python **fitter** package[5]. We obtained that their distributions can be well-approximated by Gaussians with confidence level 95%, indeed the resulting p-value is ≥ 0.05.

In Figure 1, we show the outcome for circuits of multiplicative depth 0, 1 and 2, in particular of the first coefficient. As parameters, we took $t = 65537$, $n = 2^{13}$, q as computed by the library to have at least 128 bit security, $\chi_s = \chi_u = \mathcal{U}_3$, and $\chi_e = \mathcal{DG}(0, \sigma^2)$ with $\sigma = 3.19$. We used the Hybrid key switching and `HPSPOVERQ` multiplication.

4.2 Mean Analysis

We will prove that the error coefficients always have mean 0.

Lemma 1. *Let $\nu = \sum_\iota a_\iota s^\iota$ be any invariant noise, a_ι the ι-th element of ν as a polynomial in s, and $a_\iota|_i$ its i-th coefficient. Then, $\mathbb{E}[a_\iota|_i] = 0, \forall \iota, i \in \mathbb{N}_{>0}$.*

See Appendix A for the proof of Lemma 1.

[5] https://fitter.readthedocs.io/en/latest/

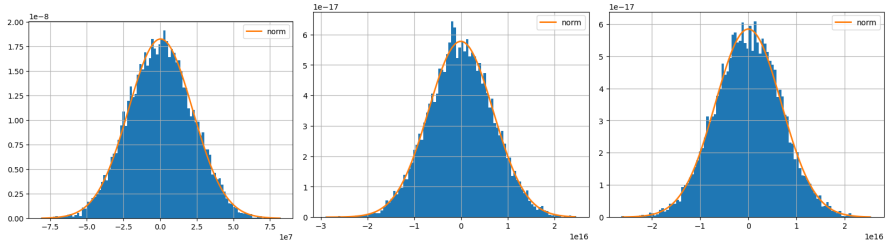


Fig. 1: (i) ks_{pval} 0.588918; (ii) ks_{pval} 0.596218; and (iii) ks_{pval} 0.744975.

Proposition 1. *Let ν be any invariant noise and $\nu|_i$ the i-th coefficient, then the average value of its coefficients is 0, i.e. $\mathbb{E}[\nu|_i] = 0$.*

Proof. Let us write the invariant noise as a polynomial in s, i.e., $\nu = \sum_\iota a_\iota s^\iota$. Then, by Equation (1), $\nu|_i = \sum_\iota (a_\iota s^\iota)|_i = a_0|_i + \sum_{\iota>0} \sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|_{i-j}$. Hence, by the linearity of the expected value and the independence between $a_\iota|_j$ and $s^\iota|_{i-j}$, properties (a) and (d) of Fact 1, $\mathbb{E}[\nu|_i] = \mathbb{E}[a_0|_i] + \sum_{\iota>0} \sum_{j=0}^{n-1} \xi(i,j) \cdot \mathbb{E}[a_\iota|_j]\mathbb{E}[s^\iota|_{i-j}] = 0$, by Lemma 1. □

4.3 Variance Analysis

In this section, we show how the variance of the error coefficients changes as homomorphic operations are performed. To do this, we need the following results.

Lemma 2. *Let $\nu = \sum_\iota a_\iota s^\iota$ be an invariant noise written as a polynomial in s, and $a_{\iota_1}|_{i_1}$, $a_{\iota_2}|_{i_2}$ any two coefficients i_1, i_2 of elements of ν . It follows that $\mathsf{Cov}(a_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) = 0$, if either $\iota_1 \neq \iota_2$ or $i_1 \neq i_2$, and that $\mathsf{Var}(a_\iota|_i)$ does not depend on the coefficient i. Moreover, we have that*

- ***Encryption.*** *The invariant noise ν_{clean} of a fresh ciphertext can be write as $\nu_{\mathsf{clean}} = a_0^{\mathsf{clean}} + a_1^{\mathsf{clean}} s$, where the coefficient variances are*

$$\mathsf{Var}(a_0^{\mathsf{clean}}|_i) = \frac{t^2}{q^2}\left(\frac{1}{12} + nV_eV_u + V_e\right) \text{ and } \mathsf{Var}(a_1^{\mathsf{clean}}|_i) = \frac{t^2}{q^2}V_e.$$

- ***Addition.*** *Let ν, ν' be the invariant noise of two independently-computed ciphertexts, then $\mathsf{Var}(a_\iota^{\mathsf{add}}|_i) = \mathsf{Var}(a_\iota|_i) + \mathsf{Var}(a'_\iota|_i)$;*
- ***Constant Multiplication.*** *Let ν_{const} be the invariant noise after a multiplication between a constant $\alpha \in \mathcal{R}_t$ and a ciphertext with noise ν, then $\mathsf{Var}(a_\iota^{\mathsf{const}}|_i) = \frac{(t^2-1)n}{12}\mathsf{Var}(a_\iota|_i)$;*
- ***Modulo Switch to q'_ℓ.*** *Let $\nu^{\mathsf{ms}}(q'_\ell) = \nu + \nu_{\mathsf{ms}}(q'_\ell)$ be the noise after the modulo switch operation to q'_ℓ, then $\mathsf{Var}(a_\iota^{\mathsf{ms}}(q'_\ell)|_i) = \mathsf{Var}(a_\iota|_i) + \frac{t^2}{12q_\ell'^2}\mathbf{1}_{\iota\in\{0,1\}}$;*

– ***Key Switch.*** *Let* $\nu^{\mathsf{ks}} = \nu + \nu_{\mathsf{ks}}$ *be the error after the key switch operation, then* $\mathsf{Var}(a_\iota^{\mathsf{ks}}|_i) \leq \mathsf{Var}(a_\iota|_i) + V_{\mathsf{ks}}^\iota(q_\ell)$*, where*

$$V_{\mathsf{ks}}^\iota(q_\ell) = \begin{cases} \frac{t^2}{12q_\ell^2} k_\ell \sqrt[k]{q^2} n V_e \mathbf{1}_{\iota=0} & \textit{for BV} \\ \frac{t^2}{12q_\ell^2}\big((nV_e+1)\mathbf{1}_{\iota=0} + \mathbf{1}_{\iota=1}\big) & \textit{for GHS} \\ \frac{t^2}{12q_\ell^2}\big(((k+2)nV_e+1)\mathbf{1}_{\iota=0} + \mathbf{1}_{\iota=1}\big) & \textit{for GHS-RNS} \\ \frac{t^2}{12q_\ell^2}\big((\omega nV_e+1)\mathbf{1}_{\iota=0} + \mathbf{1}_{\iota=1}\big) & \textit{for Hybrid} \\ \frac{t^2}{12q_\ell^2}\big(((k+2\omega)nV_e+1)\mathbf{1}_{\iota=0} + \mathbf{1}_{\iota=1}\big) & \textit{for Hybrid-RNS} \end{cases}$$

– ***Multiplication.*** *Let* $\nu = \sum_{\iota_1=0}^{T_1} a_{\iota_1} s^{\iota_1}, \nu' = \sum_{\iota_2=0}^{T_2} a'_{\iota_2} s^{\iota_2}$ *be the noises of two independently-computed ciphertexts, then*

$$\begin{aligned}\mathsf{Var}(a_\iota^{\mathsf{mul}}(q_\ell)|_i) = n \textstyle\sum_{\iota_1=0}^{T_1}\sum_{\iota_2=0}^{T_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)\mathbf{1}_{\iota_1+\iota_2=\iota} \\ + \tfrac{t^2 n}{12}\Big(\mathsf{Var}(a_\iota|_i)\mathbf{1}_{0\leq\iota\leq T_1} + \mathsf{Var}(a_{\iota-1}|_i)\mathbf{1}_{1\leq\iota\leq T_1+1} \\ + \mathsf{Var}(a'_\iota|_i)\mathbf{1}_{0\leq\iota\leq T_2} + \mathsf{Var}(a'_{\iota-1}|_i)\mathbf{1}_{1\leq\iota\leq T_2+1}\Big) + \tfrac{t^2}{12q_\ell^2}\mathbf{1}_{\iota\in\{0,1,2\}}.\end{aligned}$$

See Appendix A for the proof of the lemma.

We can finally state our results on the variance computation for operations, dedicating a special section to the multiplication.

Proposition 2. *Let* $\nu = \sum_\iota a_\iota s^\iota$ *be any invariant noise written as a polynomial in* s*,* $\nu|_i$ *its* i*-th coefficients and* $a_\iota|_i$ *the* i*-th coefficients of the* ι*-th element of* ν *written as a polynomial in* s*. Then, the variance of the noise coefficients is*

$$\mathsf{Var}(\nu|_i) = \mathsf{Var}(a_0|_i) + \sum_{\iota>0} \mathsf{Var}(a_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}\big[s^\iota|_j^2\big]. \tag{15}$$

Proof. To simplify the notation, we write $\nu|_i = \sum_\iota \sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|_{i-j}$, instead of $\nu|_i = a_0|_i + \sum_{\iota>0}\sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|_{i-j}$. By Equation (1) and the properties (k), (j) and (e) of Fact 1, the variance of the noise invariant's i-th coefficient is

$$\begin{aligned}\mathsf{Var}(\nu|_i) =& \mathsf{Var}\Big(\sum_\iota \sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|_{i-j}\Big) = \sum_\iota \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j s^\iota|_{i-j}) + \\ &+ \sum_{\substack{\iota_1\neq\iota_2 \text{ or} \\ j_1\neq j_2}} \xi(i,j_1)\xi(i,j_2)\mathsf{Cov}(a_{\iota_1}|_{j_1} s^{\iota_1}|_{i-j_1}, a_{\iota_2}|_{j_2} s^{\iota_2}|_{i-j_2}).\end{aligned}$$

By definition of covariance and the independence of a_ι and s, properties (c), (d) Fact 1, we can write $\mathsf{Cov}(a_{\iota_1}|_{j_1} s^{\iota_1}|_{i-j_1}, a_{\iota_2}|_{j_2} s^{\iota_2}|_{i-j_2})$ as

$$\mathbb{E}[a_{\iota_1}|_{j_1} a_{\iota_2}|_{j_2}]\mathbb{E}[s^{\iota_1}|_{i-j_1} s^{\iota_2}|_{i-j_2}] - \mathbb{E}[a_{\iota_1}|_{j_1}]\mathbb{E}[s^{\iota_1}|_{i-j_1}]\mathbb{E}[a_{\iota_2}|_{j_2}]\mathbb{E}[s^{\iota_2}|_{i-j_2}],$$

which, by Lemma 1, becomes

$$\mathbb{E}[a_{\iota_1}|_{j_1} a_{\iota_2}|_{j_2}]\mathbb{E}[s^{\iota_1}|_{i-j_1} s^{\iota_2}|_{i-j_2}] = \mathsf{Cov}(a_{\iota_1}|_{j_1}, a_{\iota_2}|_{j_2})\mathbb{E}[s^{\iota_1}|_{i-j_1} s^{\iota_2}|_{i-j_2}].$$

Since $\mathsf{Cov}(a_{\iota_1}|_{j_1}, a_{\iota_2}|_{j_2}) = 0$, thanks to Lemma 2, it follows that

$$\mathsf{Var}(\nu|_i) = \sum_{\iota} \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j s^\iota|_{i-j}).$$

Then, since a_ι and s are independent and, by Lemma 1, $\mathbb{E}[a_\iota|_j] = 0$ for any j, then we can apply the property (l) of Fact 1 obtaining,

$$\mathsf{Var}(a_\iota|_j s^\iota|_{i-j}) = \mathsf{Var}(a_\iota|_j)\mathbb{E}[s^\iota|^2_{i-j}].$$

Finally, by Lemma 2, $\mathsf{Var}(a_\iota|_j)$ does not depend on j, hence

$$\begin{aligned}\mathsf{Var}(\nu|_i) &= \mathsf{Var}(a_0|_i) + \textstyle\sum_{\iota>0} \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j)\mathbb{E}[s^\iota|^2_{i-j}] \\ &= \mathsf{Var}(a_0|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}(a_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|^2_{i-j}].\end{aligned}$$

□

Proposition 3 (Encryption). *The invariant noise ν_{clean} of a fresh ciphertext has coefficient variance*

$$V_{\mathsf{clean}} = \mathsf{Var}(\nu_{\mathsf{clean}}|_i) = \tfrac{t^2}{q^2}\left(\tfrac{1}{12} + nV_eV_u + V_e + nV_eV_s\right). \tag{16}$$

Proof. By Equation (5), the fresh error ν_{clean} can be written as $\nu_{\mathsf{clean}} = a_0^{\mathsf{clean}} + a_1^{\mathsf{clean}} s$. Thus, thanks to Equation (15), we have

$$\mathsf{Var}(\nu_{\mathsf{clean}}|_i) = \mathsf{Var}(a_0^{\mathsf{clean}}|_i) + \mathsf{Var}(a_1^{\mathsf{clean}}|_i) \sum_{j=0}^{n-1} \mathbb{E}[s|^2_j].$$

Moreover, $\mathbb{E}[s|^2_j] = \mathsf{Var}(s|_j) = V_s$ since $\mathbb{E}[s|_j] = 0$ (property (i) of Fact 1). Then, by Lemma 2, $\mathsf{Var}(\nu_{\mathsf{clean}}|_i) = \frac{t^2}{q^2}(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s)$. □

Proposition 4 (Addition & Constant Multiplication). *Let $\alpha \in \mathcal{R}_t$ be a constant and $\mathbf{c}, \mathbf{c}'$ be two independently-computed ciphertexts with invariant noises ν, ν', respectively. Then the variance of the error coefficients*

- *resulting from the addition of $\mathbf{c}$ and $\mathbf{c}'$ is*

$$\mathsf{Var}(\nu_{\mathsf{add}}|_i) = \mathsf{Var}(\nu|_i) + \mathsf{Var}(\nu'|_i). \tag{17}$$

- *after a multiplication between α and $\mathbf{c}$ is*

$$\mathsf{Var}(\nu_{\mathsf{const}}|_i) = \tfrac{(t^2-1)n}{12}\mathsf{Var}(\nu|_i). \tag{18}$$

Proof. By Equation (15) and Lemma 2, we have the variance of the error coefficients after

- the addition is

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{add}}|_i) &= \mathsf{Var}(a_0|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}(a_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &\quad + \mathsf{Var}(a'_0|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}(a'_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &= \mathsf{Var}(\nu|_i) + \mathsf{Var}(\nu'|_i).\end{aligned}$$

- the constant multiplication is

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{const}}|_i) &= \mathsf{Var}(a_0^{\mathsf{const}}|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}(a_\iota^{\mathsf{const}}|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &= \tfrac{(t^2-1)n}{12} \big(\mathsf{Var}(a_0|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}(a_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2]\big) \\ &= \tfrac{(t^2-1)n}{12} \mathsf{Var}(\nu|_i).\end{aligned}$$

□

Proposition 5 (Modulo Switch). *Let $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu)$ be an extended ciphertext. The variance of the error coefficients after the modulo switch to the target modulo q'_ℓ is*

$$V((\nu + \nu_{\mathsf{ms}}(q'_\ell))|_i) = \mathsf{Var}(\nu|_i) + \tfrac{t^2}{12q_\ell'^2}(1 + nV_s). \tag{19}$$

Proof. By Equation (15) and Lemma 2, since $\mathbb{E}[s|_j^2] = \mathsf{Var}(s|_j) = V_s$, we have

$$\begin{aligned}\mathsf{Var}((\nu + \nu_{\mathsf{ms}}(q'_\ell))|_i) &= \mathsf{Var}(a_0^{\mathsf{ms}}(q'_\ell)|_i) + \sum_{\iota>0} \mathsf{Var}\big(a_\iota^{\mathsf{ms}}(q'_\ell)|_i\big) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &= \mathsf{Var}(a_0|_i) + \sum_{\iota>0} \mathsf{Var}(a_\iota|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] + \frac{t^2}{12q_\ell'^2}\Big(1 + \sum_{j=0}^{n-1} \mathbb{E}[s|_j^2]\Big) \\ &= \mathsf{Var}(\nu|_i) + \frac{t^2}{12q_\ell'^2}(1 + nV_s)\end{aligned}$$

□

Proposition 6 (Key Switch). *Let $\mathfrak{d} = (\mathbf{d}, q_\ell, \nu)$ be an extended ciphertext, the variance of the error coefficients after the key switching is*

$$\mathsf{Var}((\nu + \nu_{\mathsf{ks}}(q_\ell))|_i) \le \mathsf{Var}(\nu|_i) + V_{\mathsf{ks}}(q_\ell), \tag{20}$$

where

$$V_{\mathsf{ks}}(q_\ell) = \begin{cases} \frac{t^2}{12q_\ell^2} k_\ell \sqrt[k]{q^2} n V_e & \text{for } BV \\ \frac{t^2}{12q_\ell^2} (nV_e + 1 + nV_s) & \text{for } GHS \\ \frac{t^2}{12q_\ell^2} (n(k+2)V_e + 1 + nV_s) & \text{for } GHS\text{-}RNS \\ \frac{t^2}{12q_\ell^2} (\omega n V_e + 1 + nV_s) & \text{for } Hybrid \\ \frac{t^2}{12q_\ell^2} ((k+2\omega)nV_e + 1 + nV_s) & \text{for } Hybrid\text{-}RNS \end{cases} \tag{21}$$

and ω, k and k_ℓ are defined as in Section 3.1.

Proof. Analogously, the result follows by Equation (15) and Lemma 2. Indeed, since $\mathbb{E}[s|_j^2] = \mathsf{Var}(s|_j) = V_s$, then

$$\begin{aligned}\mathsf{Var}((\nu + \nu_{\mathsf{ks}}(q_\ell)) &= \mathsf{Var}(a_0^{\mathsf{ks}}(q_\ell)|_i) + \textstyle\sum_{\iota>0} \mathsf{Var}\big(a_\iota^{\mathsf{ks}}(q_\ell)|_i\big) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &\le V(a_0|_i) + V_{\mathsf{ks}}^0(q_\ell) + \textstyle\sum_{\iota>0} \big(\mathsf{Var}(a_\iota|_i) + V_{\mathsf{ks}}^\iota(q_\ell)\big) \sum_{j=0}^{n-1} \mathbb{E}[s^\iota|_j^2] \\ &= \mathsf{Var}(\nu|_i) + V_{\mathsf{ks}}(q_\ell)\end{aligned}$$

□

Theorem 1 (Multiplication). *Let $\nu = \sum_{\iota_1=0}^{T_1} a_{\iota_1} s^{\iota_1}, \nu' = \sum_{\iota_2=0}^{T_2} a'_{\iota_2} s^{\iota_2}$ be the noises of two independently computed ciphertexts, the variance of the error coefficients after a multiplication is well-approximated by*

$$\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \le \frac{t^2 n^2 V_s}{12}\big(\mathsf{Var}(\nu|_i) f(T_1+1) + \mathsf{Var}(\nu'|_i) f(T_2+1)\big), \tag{22}$$

with $f(i)$ as in Heuristic 1.

The proof is given in the following section, due to its complexity.

4.4 On the Estimation of the Variance in the Multiplication

In this section, we prove Theorem 1. To do so, we analyze the coefficient variance $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ of the error resulting from the multiplication of two independently-computed ciphertexts, which noises are $\nu = \sum_{\iota_1=0}^{T_1} a_{\iota_1} s^{\iota_1}$ and $\nu' = \sum_{\iota_2=0}^{T_2} a'_{\iota_2} s^{\iota_2}$.

To give an idea of the problem that we are tackling in this section, we start by considering the term $\nu\nu'$ of the multiplication error. By Lemma 2 and Equation (15), we get $\mathsf{Var}\big((\nu\nu')|_i\big) = \mathsf{Var}\big(a_0^{\nu\nu'}|_i\big) + \sum_{\iota>0} \mathsf{Var}\big(a_\iota^{\nu\nu'}|_i\big) \sum_{j=0}^{n-1} \mathbb{E}\big[s^\iota|_j^2\big]$. Abusing notation for the sake of clarity, we have:

$$\begin{aligned}\mathsf{Var}\big((\nu\nu')|_i\big) &= \sum_\iota \mathsf{Var}\big(a_\iota^{\nu\nu'}|_i\big) \sum_{j=0}^{n-1} \mathbb{E}\big[s^\iota|_j^2\big] \\ &= n \sum_\iota \sum_{\iota_1,\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)\mathbf{1}_{\iota_1+\iota_2=\iota} \sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_1+\iota_2}|_j^2\big]\end{aligned}$$

Note that, computing $n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)$ (with the same abuse of notation), we obtain almost $\mathsf{Var}\big((\nu\nu')|_i\big)$. Indeed,

$$\begin{aligned}n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i) &= n\Big(\sum_{\iota_1} \mathsf{Var}(a_{\iota_1}|_i) \sum_{j_1=0}^{n-1} \mathbb{E}\big[s^{\iota_1}|_{j_1}^2\big]\Big)\Big(\sum_{\iota_2} \mathsf{Var}(a_{\iota_2}|_i) \sum_{j_2=0}^{n-1} \mathbb{E}\big[s^{\iota}|_{j_2}^2\big]\Big) = \\ &= n \sum_\iota \sum_{\iota_1,\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)\mathbf{1}_{\iota_1+\iota_2=\iota} \sum_{j_1=0}^{n-1} \mathbb{E}\big[s^{\iota_1}|_{j_1}^2\big] \sum_{j_2=0}^{n-1} \mathbb{E}\big[s^{\iota}|_{j_2}^2\big],\end{aligned}$$

where the unique difference is that

$$\sum_{j=0}^{n-1} \mathbb{E}\left[s^{\iota_1+\iota_2}|_j^2\right] \neq \sum_{j_1=0}^{n-1} \mathbb{E}\left[s^{\iota}|_{j_1}^2\right] \sum_{j_2=0}^{n-1} \mathbb{E}\left[s^{\iota}|_{j_2}^2\right].$$

Therefore, the goal of this section is to find a *correction function* F such that

$$\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota_1+\iota_2}|_j^2] \approx F(\iota_1,\iota_2) \sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota_1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s^{\iota_2}|_{j_2}^2]. \tag{23}$$

In this way, we can compute $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ from $\mathsf{Var}(\nu|_i)$ and $\mathsf{Var}(\nu'|_i)$ using a simple formula.

We achieve this as follows:

1. We start by computing the correction function $F(\iota_1,\iota_2)$ for a specific case, namely for $\iota_2 = 1$, by defining $f(\iota) := F(\iota-1,1)$.
2. Using this function f and exploiting Lemma 3, we define the *correction function* $F(\iota_1,\iota_2)$ for any ι_1 and ι_2 (Corollary 1).
3. In Lemma 4, we prove some properties of $F(\iota_1,\iota_2)$ that will be used in the proof of Theorem 1.
4. Finally, we prove Theorem 1, providing a method for the computation of $\mathsf{Var}(\nu_{\mathsf{mul}}|_i)$.

The function $f(\iota)$. As mentioned, the first step is finding a function $f(\iota)$ that approximates the following special case of the correction function F, for $\iota \geq 2$:

$$f(\iota) := F(\iota-1,1) \approx \frac{\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota}|_j^2]}{\sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota-1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s|_{j_2}^2]}. \tag{24}$$

Note that the function $f(\iota)$ only depends on the distribution χ_s and the ring dimension n.

We compute the values that $f(\iota)$ has to approximate, namely the right-hand side of Equation (24), experimentally for $\iota \leq 300$, considering 25000 samples (for more details see Appendix B).

Heuristic 1 *A function $f(\iota)$, verifying Equation (24) for $\iota \geq 2$, is*

$$f(\iota) = -e^{\alpha-\beta\iota-\gamma\iota^2} + \delta. \tag{25}$$

where α, β, γ, and δ depend on the distribution χ_s and the ring dimension n. In Table 1 we show the constant values when $\chi_s = \mathcal{U}_3$. The list of all the other values obtained for the function $f(\iota)$ across different n and χ_s, can be found in Appendix B.

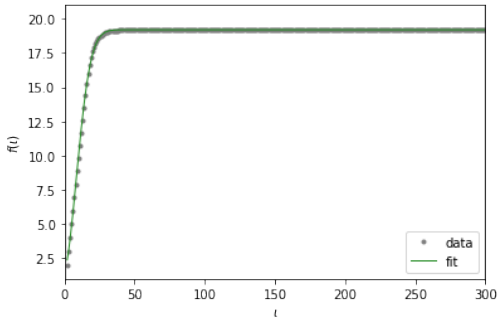
In Figures 2a and 2b we give an example with $n = 2^{12}$ and $\chi_s = \mathcal{U}_3$.

n	α	β	γ	δ
2^{12}	2.8732	0.0160	0.0049	19.1895
2^{13}	2.9644	0.0196	0.0046	20.4747
2^{14}	2.9578	0.0386	0.0032	19.5755
2^{15}	2.9765	0.0197	0.0043	20.7760

Table 1: $\chi_s = \mathcal{U}_3$

ι	(24)	$f(\iota)$
2	2.0003	2.3858
3	2.9991	3.0527
4	3.9966	3.8443
5	4.9911	4.7394
6	5.9789	5.7150
7	6.9545	6.7472
8	7.9126	7.8124
⋮	⋮	⋮
91	19.1894	19.1895
92	19.1895	19.1895
⋮	⋮	⋮
300	19.1895	19.1895

(a) Calculated values of the right-hand side of Equation (24) and $f(\iota)$ as ι varies.



(b) The grey dots represent the values from Tables (2a) (i.e., the right-hand side of Equation (24)), while the green line shows the approximation of $f(\iota)$ according to Heuristic 1.

Fig. 2: Example of the function $f(\iota)$ for $n = 2^{12}$ and $\chi_s = \mathcal{U}_3$.

The correction function $F(\iota_1, \iota_2)$. Under Heuristic 1, we are able to define the correction function F, thanks to the following result.

Lemma 3. *Let $g(\iota) = \prod_{i=0}^{\iota} f(i)$ with $f(i)$ as in Equation (24) and $f(0) = f(1) = 1$. Then for $\iota \geq 1$, $\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota}|_j^2] \approx (nV_s)^{\iota} g(\iota)$.*

Proof. The proof is done by induction on ι.

- For $\iota = 1$, $\sum_{j=0}^{n-1} \mathbb{E}[s|_j^2] = nV_s = nV_s g(1)$.
- Moreover, if the thesis holds for all $\iota' < \iota$, then by Equation (24)

$$\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota}|_j^2] \approx f(\iota) \sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota-1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s|_{j_2}^2] \approx g(\iota)(nV_s)^{\iota}.$$

□

Note that for $\iota = 0$, we can also define $\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota}|_j^2] \approx (nV_s)^{\iota} g(\iota)$. Indeed, in this case, the term $\mathsf{Var}(a_0|_i)$ is only multiplied by $g(0)(nVs)^0 = 1$.

Corollary 1. *Let* $g(\iota) = \prod_{i=0}^{\iota} f(i)$ *with* $f(i)$ *as in Equation* (24) *and* $f(0) = f(1) = 1$. *Then, the correction function* F *is*

$$F(\iota_1, \iota_2) := \frac{g(\iota_1 + \iota_2)}{g(\iota_1)g(\iota_2)} \approx \frac{\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota_1+\iota_2}|_j^2]}{\sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota_1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s^{\iota_2}|_{j_2}^2]}. \tag{26}$$

To provide an estimation of $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$, we need the following technical lemma regarding some properties of $F(\iota_1, \iota_2)$.

Lemma 4. *Let* $F(\iota_1, \iota_2)$ *be the correction function as in Equation* (26), *then*

1. $F(\iota_1, \iota_2) \leq F(T_1, T_2)$ *for any* $\iota_1 \in \{0, \ldots, T_1\}$, *and* $\iota_2 \in \{0, \ldots, T_2\}$.
2. *Let* $T_1, T_2 \in \mathbb{N}$, *then* $\frac{F(T_1,T_2)}{f(T_1+1)f(T_2+1)} \leq K_n$ *with* K_n *is a finite constant. Specifically,* $K_n < 40\,n$, *for any* $n = 2^\kappa$, *where* $\kappa \in \{12, \ldots, 15\}$.

The proof of Lemma 4 is in Appendix C.

Proof of Theorem 1. Finally, we present the proof of our main result:

Theorem 1 (Multiplication). *Let* $\nu = \sum_{\iota_1=0}^{T_1} a_{\iota_1} s^{\iota_1}, \nu' = \sum_{\iota_2=0}^{T_2} a'_{\iota_2} s^{\iota_2}$ *be the noises of two independently computed ciphertexts, the variance of the error coefficients after a multiplication is well-approximated by*

$$\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq \frac{t^2 n^2 V_s}{12} \big(\mathsf{Var}(\nu|_i) f(T_1 + 1) + \mathsf{Var}(\nu'|_i) f(T_2 + 1)\big),$$

with $f(\iota)$ *as in Heuristic 1.*

The proof is divided into two parts. Firstly, we propose a formula for the computation of (an upper bound of) $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ from $\mathsf{Var}(\nu|_i)$ and $\mathsf{Var}(\nu'|_i)$. Secondly, we show that some terms in the obtained formula are negligible. This approach simplifies the computation of the ciphertext modulus q in a circuit (see Section 5).

Proof. Abusing notation for the sake of clarity, by Lemma 2 and Equation (15), we have

$$\begin{aligned}
\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) &= \sum_{\iota} \mathsf{Var}\big(a_\iota^{\mathsf{mul}(q_\ell)}|_i\big) \sum_{j=0}^{n-1} \mathbb{E}\big[s^\iota|_j^2\big] \\
&= n \sum_{\iota_1=0}^{T_1} \sum_{\iota_2=0}^{T_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i) \sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_1+\iota_2}|_j^2\big] \\
&\quad + \frac{t^2 n}{12} \sum_{\iota_1=0}^{T_1} \mathsf{Var}(a_{\iota_1}|_i)\Big(\sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_1}|_j^2\big] + \sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_1+1}|_j^2\big]\Big) \\
&\quad + \frac{t^2 n}{12} \sum_{\iota_2=0}^{T_2} \mathsf{Var}(a'_{\iota_2}|_i)\Big(\sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_2}|_j^2\big] + \sum_{j=0}^{n-1} \mathbb{E}\big[s^{\iota_2+1}|_j^2\big]\Big) \\
&\quad + \frac{t^2}{12q_\ell^2}\Big(1 + \sum_{j=0}^{n-1} \mathbb{E}\big[s|_j^2\big] + \sum_{j=0}^{n-1} \mathbb{E}\big[s^2|_j^2\big]\Big)
\end{aligned}$$

Thanks to Corollary 1, we approximate

$$\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota_1+\iota_2}|_j^2] \approx F(\iota_1,\iota_2) \sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota_1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s^{\iota_2}|_{j_2}^2].$$

Additionally, by Lemma 3, we have $\sum_{j=0}^{n-1} \mathbb{E}[s|_j^2] \approx nV_s$ and $\sum_{j=0}^{n-1} \mathbb{E}[s^2|_j^2] \approx (nV_s)^2 f(2)$, and, in general, $\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota+1}|_j^2] \approx f(\iota{+}1) \sum_{j_1=0}^{n-1} \mathbb{E}[s^\iota|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s|_{j_2}^2]$. Hence,

$$\begin{aligned}
\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \approx n &\sum_{\iota_1=0}^{T_1} \sum_{\iota_2=0}^{T_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i) \sum_{j_1,j_2=0}^{n-1} \mathbb{E}[s^{\iota_1}|_{j_1}^2]\mathbb{E}[s^{\iota_2}|_{j_2}^2]F(\iota_1,\iota_2) + \\
&+ \frac{t^2 n}{12} \sum_{\iota_1=0}^{T_1} \mathsf{Var}(a_{\iota_1}|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^{\iota_1}|_j^2]\big(1 + nV_s f(\iota_1+1)\big) + \\
&+ \frac{t^2 n}{12} \sum_{\iota_2=0}^{T_2} \mathsf{Var}(a'_{\iota_2}|_i) \sum_{j=0}^{n-1} \mathbb{E}[s^{\iota_2}|_j^2]\big(1 + nV_s f(\iota_2+1)\big) + \\
&+ \frac{t^2}{12q_\ell^2}\Big(1 + nV_s + n^2V_s^2 f(2)\Big).
\end{aligned}$$

Moreover, by point 1 of Lemma 4, we obtain the following bound

$$\begin{aligned}
\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \le\ & n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)F(T_1,T_2) + \frac{t^2 n}{12}\mathsf{Var}(\nu|_i)\big(1 + nV_s f(T_1+1)\big) \\
&+ \frac{t^2 n}{12}\mathsf{Var}(\nu'|_i)\big(1 + nV_s f(T_2+1)\big) + \frac{t^2}{12q_\ell^2}\Big(1 + nV_s + n^2V_s^2 f(2)\Big).
\end{aligned}$$

Now, we need to make some further simplifications to obtain the thesis.

First, recall that n is a power of 2, usually at least 2^{12}. Thus we can approximate the previous inequality as

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq & n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)F(T_1,T_2) + \frac{t^2n^2V_s}{12}\Big(\mathsf{Var}(\nu|_i)f(T_1+1)+ \\ & + \mathsf{Var}(\nu'|_i)f(T_2+1)\Big) + \frac{t^2n^2V_s^2f(2)}{12q_\ell^2}. \end{aligned} \tag{27}$$

Second, we prove that the first term is negligible compared to the rest, specifically to the second term. To do so, we exploit the fact that we are pursuing correctness, namely the bound (27) is set smaller than $1/8D^2$. In particular, it follows that

$$\frac{t^2n^2V_s}{12}\mathsf{Var}(\nu'|_i)f(T_2+1) < \frac{1}{8D^2}, \text{ i.e. } \mathsf{Var}(\nu'|_i) < \frac{3}{2D^2t^2n^2V_sf(T_2+1)}.$$

Then, by point 2 of Lemma 4,

$$\begin{aligned} n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)F(T_1,T_2) \leq & \ \frac{3}{2D^2t^2nV_s}\mathsf{Var}(\nu|_i)\frac{F(T_1,T_2)}{f(T_2+1)} \\ \leq & \ \frac{3K_n}{2D^2t^2nV_s}\mathsf{Var}(\nu|_i)f(T_1+1) \\ \ll & \ \frac{t^2n^2V_s}{12}\mathsf{Var}(\nu|_i)f(T_1+1). \end{aligned}$$

Hence, the bound (27) becomes

$$\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq \frac{t^2n^2V_s}{12}\Big(\mathsf{Var}(\nu|_i)f(T_1+1) + \mathsf{Var}(\nu'|_i)f(T_2+1)\Big) + \frac{t^2n^2V_s^2}{12q_\ell^2}f(2).$$

Finally, we prove that the last term is negligible respect the other terms. Let us consider the case in which the multiplication is performed in the modulus q. We know that $\mathsf{Var}(\nu|_i) \geq V_{\mathsf{clean}}$, since all the homomorphic operations performed increase the variance of the error coefficients. Therefore, by Equation (16), we get $\mathsf{Var}(\nu|_i) \geq t^2nV_eV_s/q^2$. It follows

$$\frac{t^2n^2V_s}{12}\mathsf{Var}(\nu|_i)f(T_1+1) \geq \frac{t^4n^3V_eV_s^2}{12q^2}f(T_1+1) \gg \frac{t^2n^2V_s^2}{12q^2}f(2).$$

Hence, the thesis.

Note that if a modulo switch to a different ciphertext modulus q_ℓ has been performed, the proof is analogous. Indeed, we only need to observe that $\mathsf{Var}(\nu|_i) \geq V_{\mathsf{ms}}(q_\ell)$, where $V_{\mathsf{ms}}(q_\ell) = \frac{t^2nV_s}{12q_\ell^2}$ by Equation (19).

□

5 Closed formulas for BFV Circuits

In this section, we exploit our theoretical work (Section 4) to improve the parameter generation for the BFV scheme, providing a method to derive closed formulas for determining the ciphertext modulus q and, eventually, its sub-moduli q_j. In particular, we analyze a specific circuit and we show how to track the error growth. These formulas are implemented in our tool, which provides automated parameter selection for non-FHE experts (Section 5.3).

Recall that the coefficients of any error follow a Gaussian distribution with mean 0 and variance computable using the formulas in Proposition 2. Therefore, a lower bound for correctness on the ciphertext modulus q or its sub-moduli p_j can always be determined by tracking the increasing variance and setting its final value $\leq 1/8D^2$. However, solving this inequality can be challenging in some cases. We will explain how this problem is addressed in the following sections.

Note that the homomorphic circuits where the BFV scheme is used are mainly of two kinds: the ones that employ the division into levels using the modulo switch, hence adopting smaller ciphertext moduli at each slot of operation, and those that use a fixed ciphertext-modulus. We analyze these two cases in Section 5.2 and Section 5.1, respectively.

The modulo switching technique was introduced in the BGV scheme to reduce the error associated with the ciphertext, and it is typically applied after heavy-on-the-error operations, like homomorphic multiplications. In BFV, even if this technique does not reduce the error, it can still be beneficial for efficiency. This is because homomorphic operations are computed over smaller moduli [24,18].

5.1 State-of-the-art: Fixed Ciphertext-Modulus Circuits

Since the modulo switch technique does not reduce the error, the state-of-the-art circuits for the BFV scheme are the fixed ciphertext-modulus ones. This means that all the homomorphic operations are performed modulus the same q. The only exception is during the multiplication algorithm, where one of the two ciphertexts is *temporarily* moved to q' (which has approximately the same size of q); however, the result obtained after the multiplication is again in modulo q.

In Table 2, we summarize the variance for each homomorphic operation. These results are coming from Propositions 3 to 6 and Theorem 1, where V, V', V'' are the error coefficients' variances of independently-computed ciphertexts $\mathbf{c}, \mathbf{c}'$ in modulo q and $\mathbf{c}''$ in modulo q'. Moreover, T_1, T_2 are the degrees of errors as polynomials in s of $\mathbf{c}, \mathbf{c}''$ and $f(\iota)$ is defined as Equation (25).

We point out that the variance can always be written as $\frac{B}{q^2}$, where B does not depend[6] on q. Therefore, to find q, we solve an inequality of the form $\frac{B}{q^2} \leq \frac{1}{8D^2}$, obtaining $q \geq \sqrt{8D^2B}$.

[6] The BV relinearization is the only exception, but its use is no longer common in practice (see Section 3.1).

Homom. operation	Variance	
`Enc`	$V_{\mathsf{enc}} = B_{\mathsf{clean}}/q^2$	$B_{\mathsf{clean}} = t^2(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s)$
`Add`$(\mathbf{c}, \mathbf{c}')$	$V + V'$	
`Const`$(\mathbf{c})$	$B_{\mathsf{const}}V$	$B_{\mathsf{const}} = \frac{(t^2-1)n}{12}$
`Mod Switch`(q')	$V + B_{\mathsf{ms}}/q'^2 \approx V + B_{\mathsf{ms}}/q^2$	$B_{\mathsf{ms}} = \frac{t^2(1+nV_s)}{12}$
`Key Switch`	$V + B_{\mathsf{ks}}/q^2$	$B_{\mathsf{ks}} = V_{\mathsf{ks}}(q)/q^2$, where $V_{\mathsf{ks}}(q)$ is as in (21)
`Mult`$(\mathbf{c}, \mathbf{c}'')$	$\frac{t^2n^2V_s}{12}\big(Vf(T_1+1) + V''f(T_2+1)\big)$	

Table 2: Variance depending on the homomorphic operations.

In the following example, we will track the error for a specific circuit (Figure 3 Model 2). We have chosen this circuit among those available in our tool because it has the most complex error growth to track. We believe that by providing this example, users will be able to generalize our approach for calculating the error in simpler cases. Additionally, this circuit exhibits the highest error growth compared to all other circuits.

In Section 5.2, we will refer back to this same example to calculate the ciphertext modulus q in both scenarios that involve the use of modulo switching. Finally, in Section 5.3, we also provide the closed formulas for determining q for all the other circuits in Figure 3.

Example 1. Consider the Model 2 circuit in Figure 3. This circuit has a multiplicative depth of $M = L - 1$, and we operate on η independently computed ciphertexts in parallel. In this circuit, we perform τ rotations followed by a constant multiplication. The resulting ciphertexts are then summed together and used as input for a multiplication with relinearization. This new ciphertext is subsequently used as an input for the same circuit. Considering this circuit, we are going to compute the ciphertext modulus q by tracing the variance's growth and setting its final value smaller or equal than $1/8D^2$.

Let us define $V_0 = V_{\mathsf{clean}}$ the variance after the fresh encryption (Equation (16)) and V_ℓ the variance after the ℓ-th multiplication (and relinearization), hence we will set $V_{L-1} \leq 1/8D^2$. From Table 2, we write the variance as $V = B/q^2$, where B is not depend on q, then we denote $V_0 = V_{\mathsf{clean}} = B_{\mathsf{clean}}/q^2$ and compute V_ℓ from $V_{\ell-1}$ with the following steps:

- We first apply τ rotations to each ciphertext, obtaining, by Equation (20), $V_{\ell-1} + \tau V_{\mathsf{ks}}(q) = V_{\ell-1} + \tau B_{\mathsf{ks}}/q^2$.
- Secondly, we have a constant multiplication. Thus, by Proposition 4, the variance is multiplied by $B_{\mathsf{const}} = \frac{(t^2-1)n}{12}$, becoming $(V_{\ell-1} + \tau\frac{B_{\mathsf{ks}}}{q^2})B_{\mathsf{const}}$.
- We add η ciphertexts, obtaining $\eta(V_{\ell-1} + \tau\frac{B_{\mathsf{ks}}}{q^2})B_{\mathsf{const}}$, thanks to Proposition 4.
- Finally, we perform a homomorphic multiplication between pairs of ciphertexts. Recall from Section 3 that, in each pair, a modulo switch to the modulus $q' \approx q$ is applied to one of the two ciphertexts. Hence, by Proposition 5, its variance becomes approximately $\eta(V_{\ell-1} + \tau\frac{B_{\mathsf{ks}}}{q^2})B_{\mathsf{const}} + B_{\mathsf{ms}}/q^2$, with $B_{\mathsf{ms}} = t^2(1 + nV_s)/12$. Performing also the multiplication and relin-

earization, we get

$$V_\ell \approx \frac{t^2 n^2 V_s}{12}\left(2\eta(V_{\ell-1}+\tau\tfrac{B_{\mathsf{ks}}}{q^2})B_{\mathsf{const}}+\tfrac{B_{\mathsf{ms}}}{q^2}\right)f(\ell+1)+\tfrac{B_{\mathsf{ks}}}{q^2}$$
$$\approx \frac{t^2 n^2 V_s}{12}\left(2\eta(V_{\ell-1}+\tau\tfrac{B_{\mathsf{ks}}}{q^2})B_{\mathsf{const}}+\tfrac{B_{\mathsf{ms}}}{q^2}\right)f(\ell+1). \tag{28}$$

Recursively, we have that $V_\ell = B_\ell/q^2$ with B_ℓ independent of q and, in particular,

$$V_\ell = \frac{B_\ell}{q^2} \approx \frac{(AB_{\ell-1}+C)f(\ell+1)}{q^2} \tag{29}$$

where $A = \eta\frac{t^2n^2V_s}{6}B_{\mathsf{const}}$ and $C = \frac{t^2n^2V_s}{12}(2\eta\tau B_{\mathsf{ks}}B_{\mathsf{const}} + B_{\mathsf{ms}})$. Hence,

$$V_{L-1} = \frac{B_{L-1}}{q^2} \approx \frac{(AB_{L-2}+C)f(L)}{q^2} \approx \frac{A(AB_{L-3}+C)f(L-1)f(L)}{q^2}$$
$$\approx \cdots \approx \frac{A^{L-2}(AB_{\mathsf{clean}}+C)g(L)}{q^2},$$

and, setting $V_{L-1} \leq 1/8D^2$, we obtain

$$q^2 \geq 8D^2A^{L-2}(AB_{\mathsf{clean}}+C)g(L). \tag{30}$$

5.2 Circuits Exploiting the Modulo Switch

In this section, we study the case where modulus switching to smaller moduli is applied in two different ways: the first follows the BGV-like approach (see [18]), the second was proposed by Kim *et al.* in [30]. Moreover, we propose a set of parameters to limit the error growth difference compared to the non-leveled circuits (Section 5.1), focusing on the same circuit as in Example 1.

BGV-like circuit In BGV [9], the noise is managed for the first time without the bootstrapping by using the modulo switching technique. Hence, the encryption modulus is a product $q = \prod_{j=1}^{L} p_j$, and the operations are performed progressively in the sub-moduli $q_L, q_{L-1}, \ldots, q_1$ with $q_\ell = \prod_{j=1}^{\ell} p_j$. Specifically, in BGV, the modulus is switched from q_ℓ to $q_{\ell-1}$ when the noise level approaches the threshold that permits correct decryption, typically after a homomorphic multiplication.

Considering the same circuit as in Example 1, the p_j can be divided into three types:

- The top one p_L is smaller than the others, since since no operations are performed between the enctyption and the first modulo switch.
- The middle ones, p_ℓ where $2 \leq \ell \leq L-1$, are approximately of the same size. The modulo switch from q_ℓ to $q_{\ell-1}$ is executed after each round of operations (i.e., after τ rotations, a constant multiplication, η additions and a homomorphic multiplication with relinearization), continuing until the last modulus $q_1 = p_1$.

– For the bottom modulus p_1, the same operations are performed. However, instead of performing a key switch and a modulus switch after the final multiplication, we decrypt right after this multiplication using s and s^2, reducing the overall amount of operations performed. For this reason and to guarantee correctness p_1 is bigger than the previous p_i.

These types of circuits were also utilized for the BFV scheme [18]. We provide a clearer explanation by demonstrating the parameter settings in the circuit example from the previous section.

Example 2. The circuit is the same as Example 1. Using the same argument, we compute the noise variance starting from $V_0^{\mathsf{ms}} = V_{\mathsf{clean}}$ and we only need to ensure that the final variance, V_{L-1}^{ms}, is bounded. The analysis differs for the presence of many moduli. At the ℓ-th level we switch from $q_{L-\ell+1}$ to $q_{L-\ell}$, yielding

$$V_{\ell-1}^{\mathsf{ms}} + V_{\mathsf{ms}}(q_{L-\ell}) = V_{\ell-1}^{\mathsf{ms}} + \frac{B_{\mathsf{ms}}}{q_{L-\ell}^2}$$

with B_{ms} as in Table 2 and the errors of the next operations are divided by $q_{L-\ell}^2$ as well. Thus, similarly to Equation (28), we have

$$V_{\ell}^{\mathsf{ms}} \approx \frac{t^2 n^2 V_s}{12}\left(2\eta\Big(V_{\ell-1}^{\mathsf{ms}} + \frac{B_{\mathsf{ms}} + \tau B_{\mathsf{ks}}}{q_{L-\ell}^2}\Big)B_{\mathsf{const}} + \frac{B_{\mathsf{ms}}}{q_{L-\ell}^2}\right) f(\ell+1).$$

Therefore

$$V_{\ell}^{\mathsf{ms}} \approx \left(A_{\mathsf{ms}} V_{\ell-1} + \frac{C_{\mathsf{ms}}}{q_{L-\ell}^2}\right) f(\ell+1), \tag{31}$$

where $A_{\mathsf{ms}} = \frac{\eta t^2 n^2 V_s}{6} B_{\mathsf{const}}$ and $C_{\mathsf{ms}} = \frac{t^2 n^2 V_s}{12}\big(2\eta\tau B_{\mathsf{ks}} B_{\mathsf{const}} + (2\eta B_{\mathsf{const}} + 1)B_{\mathsf{ms}}\big)$. Note that $A_{\mathsf{ms}} = A$ and $C_{\mathsf{ms}} > C$, where A, C are as in Example 1. Thanks to Equation (31), we can recursively compute the variance V_{L-1}^{ms} as

$$\begin{aligned} V_{L-1}^{\mathsf{ms}} &\approx A V_{L-2}^{\mathsf{ms}} f(L) + \tfrac{C_{\mathsf{ms}}}{q_1^2} f(L) \approx \\ &\approx A^2 V_{L-3}^{\mathsf{ms}} f(L-1) f(L) + \tfrac{A C_{\mathsf{ms}}}{q_2^2} f(L-1) f(L) + \tfrac{C_{\mathsf{ms}}}{q_1^2} f(L) \approx \cdots \approx \\ &\approx A^{L-1} V_0^{\mathsf{ms}} f(2) \cdots f(L) + \textstyle\sum_{i=1}^{L-1} \tfrac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} f(L-i+1) \cdots f(L), \end{aligned}$$

therefore,

$$\frac{A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L) + \sum_{i=1}^{L-1} \frac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \leq \frac{1}{8D^2}. \tag{32}$$

Observe that, since $C_{\mathsf{ms}} > C$ and $q_\ell \leq q_L$, $V_{L-1}^{\mathsf{ms}} > V_{L-1}$. This implies that the ciphertext modulus obtained with the modulus switch technique, $q_{\mathsf{ms}} = q_L$, is bigger than the modulus q obtained in Equation (30). However, we can select specific sub-moduli for them to be close, improving efficiency.

Fact 2 *An optimal choice for the p_j's, maximizing the efficiency while keeping the ciphertext modulus close to the one gotten without modulus-switching, is*

obtained when the addends in Equation (32) are approximately of the same size, namely when

$$p_1^2 \approx 8D^2 L C_{\mathsf{ms}} f(L), \quad p_\ell^2 \approx A f(L-\ell-1), \quad p_L^2 \approx \frac{A B_{\mathsf{clean}}}{C_{\mathsf{ms}}}.$$

Then $q_{\mathsf{ms}}^2 \approx 8D^2 L A^{L-1} B_{\mathsf{clean}} g(L)$*, which means that* q_{ms} *is approximately* $\sqrt{L}$ *times the ciphertext modulus* q *in Equation (30).*

Proof. We begin our proof by contradiction, assuming that there exists at least one index i in Equation (32) such that

$$\frac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \gg \frac{A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L), \tag{33}$$

Then, called $N \geq 1$ the number such indices, we get from Equation (32)

$$V_{L-1}^{\mathsf{ms}} \approx \frac{N A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \leq \frac{1}{8D^2}$$

and, consequently, $q_i^2 \geq 8D^2 N A^{i-1} C_{\mathsf{ms}} \frac{g(L)}{g(L-i)}$. From Equation (33), it also follows $\frac{q_L^2}{q_i^2} \gg \frac{A^{L-i} B_{\mathsf{clean}}}{C_{\mathsf{ms}}} g(L-i)$, which implies $q_{\mathsf{ms}}^2 \gg 8D^2 N A^{L-1} B_{\mathsf{clean}} g(L)$, much larger than the bound for q given by (30).

Thus, we now suppose that, for any index i, we have

$$\frac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \leq \frac{A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L). \tag{34}$$

So that

$$V_{L-1}^{\mathsf{ms}} \leq \frac{L A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L), \tag{35}$$

namely, $q_{\mathsf{ms}}^2 \geq 8D^2 L A^{L-1} B_{\mathsf{clean}} g(L)$. From Equation (34) we get

$$p_L^2 \leq \frac{A B_{\mathsf{clean}}}{C_{\mathsf{ms}}}, \; p_{L-1}^2 p_L^2 \leq \frac{A^2 B_{\mathsf{clean}}}{C_{\mathsf{ms}}} g(2), \; \ldots, \; p_2^2 \cdots p_L^2 \leq \frac{A^{L-1} B_{\mathsf{clean}}}{C_{\mathsf{ms}}} g(L-1).$$

Moreover, from Equation (35), we take $p_1^2 \cdots p_L^2 \approx 8D^2 L A^{L-1} B_{\mathsf{clean}} g(L)$. For maximal efficiency, we choose p_1 to be as small as possible by setting $p_2^2 \cdots p_L^2$ the largest, i.e. satisfying $p_2^2 \cdots p_L^2 \approx A^{L-1} B_{\mathsf{clean}} g(L-1)/C_{\mathsf{ms}}$. This yields $p_1^2 \approx 8D^2 L C_{\mathsf{ms}} f(L)$. We can apply the same argument iteratively to $p_2, \ldots, p_{L-1}$, obtaining the values of the thesis, i.e. $p_\ell^2 \approx A f(L-\ell-1)$, for $\ell = 2, \ldots, L-1$. Finally, from these values and $p_1^2 \cdots p_L^2 \approx 8D^2 L A^{L-1} B_{\mathsf{clean}} g(L)$, we get $p_L^2 \approx A B_{\mathsf{clean}} / C_{\mathsf{ms}}$. □

Kim et al. circuit In [30], the authors proposed an alternative approach that applies the modulo switch only during multiplications, which is the most expensive operation. Specifically, the ciphertexts modulo q is internally switched to a smaller modulus q_{lev}, the homomorphic multiplication is performed, and then the results are scaled back up to q. Although the error remains larger compared to the fixed ciphertext-modulus case, it can be kept relatively close.

Example 3. Let us consider the same circuit as in Examples 1 and 2. In this case, we obtain

$$V_\ell \approx 2\left[\eta\left(V_{\ell-1}+\frac{\tau B_{\mathsf{ks}}}{q^2}\right)B_{\mathsf{const}}+\frac{B_{\mathsf{ms}}}{q_{lev}^2}\right]\frac{t^2n^2V_s}{12}f(\ell+1)+\frac{B_{\mathsf{ms}}}{q^2}+\frac{B_{\mathsf{ks}}}{q^2},$$

which, written as $V_\ell \approx AV_{\ell-1}f(\ell+1)+\frac{C_1f(\ell+1)+C_2}{q^2}+\frac{Ef(\ell+1)}{q_{lev}^2}$, yields

$$V_{L-1} \approx A^{L-2}g(L)\left[\frac{AB_{\mathsf{clean}}+C_1+C_2/f(2)}{q^2}+\frac{E}{q_{lev}^2}\right] \leq \frac{1}{8D^2},$$

with $A=\frac{\eta t^2n^2V_s}{6}B_{\mathsf{const}}$, $C_1=\eta\tau\frac{t^2n^2V_s}{6}B_{\mathsf{const}}B_{\mathsf{ks}}$, $C_2=B_{\mathsf{ms}}+B_{\mathsf{ks}}$, $E=\frac{t^2n^2V_s}{6}B_{\mathsf{ms}}$. To have a level of security similar to the previous one, we can take q_{lev} such that $\frac{AB_{\mathsf{clean}}+C_1+C_2/f(2)}{q^2} \approx \frac{E}{q_{lev}^2}$, i.e. $q_{lev}^2 \approx \frac{E}{AB_{\mathsf{clean}}+C_1+C_2/f(2)}q^2$. Then the bound on q become approximately

$$q^2 \geq 16D^2A^{L-2}g(L)(AB_{\mathsf{clean}}+C_1+C_2/f(2)).$$

5.3 A Parameter Generator for BFV

To make our work more valuable and approachable for practical purposes, we provide automated parameter generation implemented in Python and publicly available on GitHub [7]. We integrated our theoretical work for the BFV scheme in the tool of Mono *et al.* [36], combining the correctness analysis developed in the previous sections with the formula for security in their paper.

In this tool, the focus is on the circuit models proposed by Mono *et al.* [36]. The circuit models chosen perform a list of operations on η ciphertexts c_i in parallel, as illustrated in Figure 3. The resulting ciphertexts are homomorphically multiplied with other ones computed analogously. This sequence is repeated M times.

Base model This is a simplified version of the other models, performing constant multiplications on the ciphertexts and summing them afterwards, before the homomorphic multiplication. It is mainly used to make the analysis easier, and it is equal to Model 1 and 2 with $\tau = 0$.

[7] https://github.com/Crypto-TII/fhegen

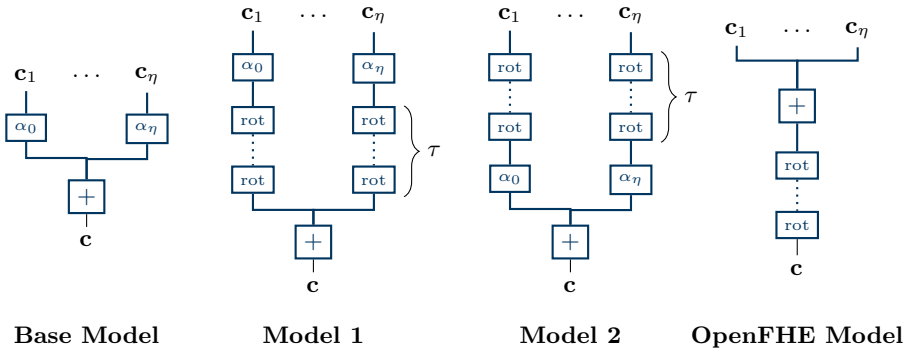


Fig. 3: Sequences of operations in the different models.

Model 1 & **2** Models 1 and 2 extend the Base Model performing τ rotations either after or before the constant multiplications, respectively.

OpenFHE Model For comparison with previous work, we also define the model used in the OpenFHE library [30,2]. Here the first operation to be performed is a homomorphic multiplication, then η additions and τ rotations are carried out.

As for the rest of the paper, in the parameters generation, we assume that the input ciphertexts for each circuit encrypt different messages, therefore all the operation involve independently computed ciphertexts. For the Base Model and Model 1, the modulus q can be computed using Equation (30):

$$q^2 \geq 8D^2A^{L-2}(AB_{\mathsf{clean}} + C)g(L),$$

analogously to Example 1 (Model 2).

We make slight modifications for the OpenFHE Model, where the multiplication occurs at the beginning of the circuit. In this case, we approximate $V_\ell = \frac{AB_{\ell-1}f(\ell+1)+C}{q^2}$, hence

$$q^2 \geq 8D^2A^{L-2}(AB_{\mathsf{clean}} + C/f(2))g(L). \tag{36}$$

In Table 3, we list the resulting A and C depending on the models.

Model	A	C
Base Model	$\frac{\eta t^2 n^2 V_s}{6} B_{\mathsf{const}}$	$\frac{t^2 n^2 V_s}{12} B_{\mathsf{ms}}$
Model 1	$\frac{\eta t^2 n^2 V_s}{6} B_{\mathsf{const}}$	$\frac{t^2 n^2 V_s}{12}(2\eta\tau B_{\mathsf{ks}} + B_{\mathsf{ms}})$
Model 2	$\frac{\eta t^2 n^2 V_s}{6} B_{\mathsf{const}}$	$\frac{t^2 n^2 V_s}{12}(2\eta\tau B_{\mathsf{ks}} B_{\mathsf{const}} + B_{\mathsf{ms}})$
OpenFHE Model	$\frac{\eta t^2 n^2 V_s}{6}$	$(\eta + \tau)B_{\mathsf{ks}}$

Table 3: A, C to compute q with either (30), or (36) for the OpenFHE one.

The generator interacts with the user by asking a series of questions and presenting a list of mandatory and optional inputs, then generating code snippets based on the obtained parameters. We provide a list of required inputs in the first part and optional inputs in the second part of Table 4.

Model	`'Base'`, `'Model1'`, `'Model2'`, `'OpenFHE'`
t or $\log t$	any integer ≥ 2
λ or m	any integer ≥ 40 or ≥ 4, respectively
M, η	any integer > 0
τ	any integer ≥ 0
Library	`'None'`, `'OpenFHE'`, `'PALISADE'`, `'SEAL'`
Full Batching	full batching with t, `'True'` or `'False'`
Secret Distribution	`'Ternary'`, `'Error'`
Key Switching	`'Hybrid'`, `'BV'`, `'GHS'`
β	any integer ≥ 2
ω	any integer ≥ 1

Table 4: Required and optional inputs to the parameter generator

This approach ensures high versatility and comprehensiveness, supporting multiple state-of-the-art libraries and all the circuits in Figure 3. Moreover, its implementation is easily adaptable to any sequence of operations.

To support arbitrary circuit models, we adapt Mono *et al.* approach for the key switching noise estimation to our average-case analysis. We use fixed values for β and ω, per default $\beta = 2^{10}$ and $\omega = 3$. When applicable, we set the key switching modulus P approximately equal to the ciphertext modulus q in the GHS variant, and to the submoduli $\tilde{r}_i$ that split it in the Hybrid one, and scale it by a constant K, per default $K = 100$. Using this estimate for the extension modulus, we compute the noise bound programmatically.

Note that, although this approach slightly overestimates the error, the noise growth due to key switching is relatively small compared to other operations. Thus, this estimation still results in valid parameter sets. This generalization extends our theoretical work to arbitrary, use-case-specific circuit models with a user-friendly interface.

6 Comparison with Previous Works

In this section, we demonstrate the efficacy of our average-case approach by comparing it to the state-of-the-art works [29,18,20,30] and the practical errors arising from OpenFHE [2].

In particular, we conduct this analysis for the basic homomorphic operations: Encryption of a fresh ciphertext and Addition and Multiplication between 2 fresh independently computed ciphertexts (Table 7). Moreover, in Tables 8 and 9 we focus on circuits. Specifically, Table 8 examines the Base Model circuits (Figure 3) with $\eta = 8$ and depth 2 and 3, while the same circuit is analyzed with

depths 4 and 5 in Table 9. Obviously, we can apply our analysis to any circuit evaluated on independent ciphertexts.

To ensure clarity, we summarize the main results needed for the comparison. The bounds with the canonical norm are computed following the latest work by Costache *et al.* [20], and Iliashenko [29], taking into account the modifications we made to the encryption and multiplication algorithms based on the work of Kim *et al.* [30]. Moreover, we recall our formulas from Sections 4 and 5.

Canonical norm. In contrast to our approach, the latest works establishing theoretical bounds on the BFV noise growth propose a worst-case analysis employing either the infinity norm [30] or the canonical norm [29,18,20]. The canonical norm is known to result in better parameters.

In Table 5 we summarize how the error behaves when the homomorphic operations are performed considering the error bounds using the canonical norm.

Homomorphic operation	Error bounds with canonical norm
`Enc`	$\|\|\nu_{\mathsf{clean}}\|\|^{\mathrm{can}} \leq D\frac{t}{q}\sqrt{n\left(\frac{1}{12}+nV_eV_u+V_e+nV_eV_s\right)}$
`Mod Switch`(q')	$\|\|\nu+\nu_{\mathsf{ms}}(q')\|\|^{\mathrm{can}} \leq \|\|\nu\|\|^{\mathrm{can}}+\frac{D\sqrt{nB_{\mathsf{ms}}}}{q'}$
`Key switch`(q)	$\|\|\nu+\nu_{\mathsf{ks}}\|\|^{\mathrm{can}} \leq \|\|\nu\|\|^{\mathrm{can}}+D\sqrt{nV_{\mathsf{ks}}}$
`Add`$(\mathbf{c},\mathbf{c}')$	$\|\|\nu+\nu'\|\|^{\mathrm{can}} \leq \|\|\nu\|\|^{\mathrm{can}}+\|\|\nu'\|\|^{\mathrm{can}}$
`Const`$(\mathbf{c})$	$\|\|\alpha\nu\|\|^{\mathrm{can}} \leq D\sqrt{n\frac{(t^2-1)}{12}}\|\|\nu\|\|^{\mathrm{can}}$
`Mult`$(\mathbf{c},\mathbf{c}')$	$\|\|\nu_{\mathrm{mul}}\|\|^{\mathrm{can}} \leq \left(2\|\|\nu\|\|^{\mathrm{can}}+D\sqrt{nV_{\mathsf{ms}}(q)}\right)Dt\sqrt{\frac{n}{12}(1+nV_s)}$

Table 5: Canonical norm depending on the homomorphic operations.

In [18], the authors, assuming independence among the coefficients, used the bound $||a||^{can} \leq D\sqrt{nV_a}$ for polynomials $a \in \mathcal{R}$, where D is an integer in general set equal to 6 and V_a is the variance of the coefficients of a. With the same hypothesis, we can bound the canonical norm of the invariant noise ν with $||\nu||^{can} \leq D\sqrt{nV}$, whose probability is greater or equal to $1-ne^{-D^2}$, by Equation (4). In line with the previous works, we set $D=6$ which guarantees the bound with probability at least $1-2^{-36}$. It's worth noting that, in a practical scenario is better to choose $D=8$ since the probability of failure is limited to 2^{-77} (for n smaller than 2^{15}).
Applying the same argument of Section 5.1, we get that the following bound on the final error of a Base Model circuit: $||\nu_{L-1}||^{\mathrm{can}} \leq A^{L-2}\left(AD\sqrt{nB_{\mathsf{clean}}}+C\right)/q$, with $A = D\eta t\sqrt{\frac{n}{3}(1+nV_s)}$ and $C = \frac{D^2t^2n}{12}(1+nV_s)$. Since the norm has to satisfy $||\nu_{L-1}||^{\mathrm{can}} \leq 1/2$, it follows that

$$q \geq 2A^{L-2}\left(AD\sqrt{nB_{\mathsf{clean}}}+C\right). \tag{37}$$

Average-case bounds. In the average-case approach, we set $||\nu||_\infty \leq D\sqrt{2V}$ with V variance of each coefficient of ν. Thanks to Equation (3), the bound holds with probability at least $1 - n\,(1 - \mathrm{erf}(D))$, which for $D = 6$ is at least $1 - 2^{-40}$.

Summarizing the results of Section 4, let ν, ν' be the invariant noises associated with the ciphertexts $\mathbf{c}$ and $\mathbf{c}'$, results of independent circuits of depth $\ell - 1$. Let V be the variance of their coefficients, in Table 2 we recall how it changes depending on the homomorphic operations.

Homomorphic operation	Variance
$\texttt{Enc}$	$\frac{t^2}{q^2}\left(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s\right)$
$\texttt{Mod Switch}(q')$	$V + \frac{t^2(1+nV_s)}{12q'^2}$
$\texttt{Key switch}(q)$	$V + V_{\mathsf{ks}}(q)$
$\texttt{Add}(\mathbf{c}, \mathbf{c}')$	$2V$
$\texttt{Const}(\mathbf{c})$	$\frac{(t^2-1)n}{12}V$
$\texttt{Mult}(\mathbf{c}, \mathbf{c}')$	$\frac{t^2n^2V_s}{12}(2V + V_{\mathsf{ms}})f(\ell+1)$

Table 6: Variance depending on the homomorphic operations.

In Tables 7 to 9, we compare the error analysis performed using the canonical norm, using our method, and the experimental results obtained from the OpenFHE library. For readability, we do not show the bounds themselves, but their *noise budget*: $-\log_2(2\cdot||\nu||) = \log_2\left(\frac{1}{2}\right) - \log_2(||\nu||)$, [39]. Roughly speaking, this measures in bits the distance between the input and $\frac{1}{2}$, which is the limit for correct decryption.

The tag "can" denotes the state-of-the-art analysis carried out with the canonical norm, "our" presents the results obtained with the average-case approach presented in this paper, "exp" shows the observed values from OpenFHE [2] library with 2^{15} polynomial samples. We additionally display the average of the absolute error values under "mean", in Tables 8 and 9 we also present our estimation of this as $\sqrt{V}$, tagged as "our".

For parameters, we use $t = 65537$, $n = 2^{12}, \ldots, 2^{15}$ and q set by OpenFHE library[8]. We highlight the results in black in the tables when the security level is at least 128-bit, and in grey when it is below this threshold. We use Hybrid key switching and HPSPOVERQ multiplication and set $D = 6$, $\chi_s = \chi_u = \mathcal{U}_3$, and $\chi_e = \mathcal{DG}(0, \sigma^2)$, with $\sigma = 3.19$.

In Table 7, we display the results after only the encryption, an encryption followed by an addition or an encryption followed by a multiplication.

In Table 8, we consider the Base Model circuit (Figure 3) of depth 2 and 3, taking $\eta = 8$. In Table 9, the same circuit is analyzed with depths 4 and 5.

[8] Specifically, for Encryption and Addition $\log_2 q \approx 60$; for Multiplication or Multiplicative depth 2, $\log_2 q \approx 120$; for depth 3 or 4, $\log_2 q \approx 180$; and for 5, $\log_2 q \approx 240$.

	Encryption				Addition				Multiplication			
	maximum value			mean	maximum value			mean	maximum value			mean
n	can	our	exp	exp	can	our	exp	exp	can	our	exp	exp
2^{12}	26.5	32.0	33.0	35.4	25.5	31.5	32.3	34.9	57.0	65.0	66.4	68.7
2^{13}	25.5	31.5	32.5	35.0	24.5	31.0	32.0	34.4	55.0	63.6	64.8	67.2
2^{14}	24.5	31.0	32.0	34.4	23.5	30.5	31.4	33.9	53.0	62.1	63.1	65.7
2^{15}	23.5	30.5	31.5	33.9	22.5	30.0	31.2	33.4	51.0	60.5	61.9	64.2

Table 7: Encryption, addition and multiplication of fresh ciphertexts.

	Multiplicative depth 2					Multiplicative depth 3				
	maximum value			mean value		maximum value			mean value	
n	can	our	exp	our	exp	can	our	exp	our	exp
2^{12}	21.5	34.8	36.2	37.9	38.6	49.0	65.9	67.3	69.0	69.7
2^{13}	18.5	32.4	33.8	35.4	36.1	45.0	62.5	63.9	65.6	66.2
2^{14}	15.5	29.9	30.9	33.0	33.5	41.0	59.0	60.1	62.1	62.7
2^{15}	12.5	27.3	28.8	30.4	31.2	37.0	55.4	57.0	58.5	59.4

Table 8: Comparison in the Base Model of depth 2 and 3 with $\alpha = 1$ and $\eta = 8$.

	Multiplicative depth 4					Multiplicative depth 5				
	maximum value			mean value		maximum value			mean value	
n	can	our	exp	our	exp	can	our	exp	our	exp
2^{12}	16.5	36.8	38.5	39.9	40.9	44.0	67.7	68.9	70.8	71.6
2^{13}	11.5	32.4	33.5	35.5	36.1	38.0	62.2	63.8	65.3	66.2
2^{14}	6.5	28.0	29.3	31.1	31.8	32.0	56.8	57.7	59.9	59.9
2^{15}	1.5	23.4	24.9	26.5	27.2	26.0	51.2	52.2	54.3	54.3

Table 9: Comparison in the Base Model of depth 4 and 5 with $\alpha = 1$ and $\eta = 8$.

Tables 7 to 9 suggest that our approach is a promising method for analyzing noise in the BFV scheme. It provides more accurate results, very close to the experimentally observed ones, and it significantly improves upon previous works, especially as the multiplicative depth of the circuit grows. For example, for a circuit with multiplicative depth 3, our bounds are up to 18.4 bits tighter than the state-of-the-art, and up to 25.2 bits for circuits with depth 5, with a difference of less than 2 bits compared to the actual values.

Our last comparison is on the ciphertext modulus q. In Figure 4, we present the obtained bounds for $\log_2(q)$ following from the two theoretical approaches (Equation (37) and Equation (30)) when $\eta = 8$, $\alpha = 1$ and either $M = 3$ (4a) or $M = 5$ (4b). We set $D = 8$ to have a failure probability smaller than 2^{-80}, which is usually required in a practical scenario.

Here we can see the impact that a better noise analysis has on the scheme's efficiency and security, indeed for a simple circuit with multiplicative depth 5, the ciphertext modulus decreases by at least 12.6%.

n	2^{12}	2^{13}	2^{14}	2^{15}
can	132.6	136.6	140.6	144.6
our	114.5	117.9	121.4	125.0

(a) Circuit of depth 3 and $\eta = 8$.

n	2^{12}	2^{13}	2^{14}	2^{15}
can	198.5	204.5	210.5	216.5
our	172.7	178.2	183.6	189.2

(b) Circuit of depth 5 and $\eta = 8$.

Fig. 4: Comparison of $\log_2(q)$ in the Base Model circuit (setting $D = 8$).

Finally, in Figure 5, we graphically compare our parameter generation with the OpenFHE one, based on theoretical work with the infinity norm [30]. We compare our generated bounds with the size of the ciphertext modulus generated for $\lambda = 128$.

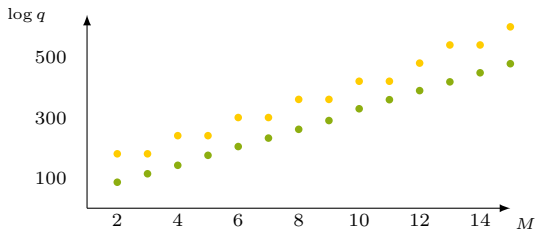


Fig. 5: Comparison of modulus sizes across multiplicative depths M with $\lambda = 128$ and $t = 2^{16} + 1$ for OpenFHE ● and our ● parameter generation.

7 Comparing Error Bounds for Dependent and Independent Ciphertexts

In this section, we consider a more general setting where ciphertexts can be dependently computed. Unlike the independent case analyzed in the previous sections, here we allow ciphertext inputs to be repeated, as for example in the homomorphic computation of $x^2 + x + 1$. This introduces additional complexity, as dependencies between ciphertexts create new challenges.

The aim of this section is to establish a baseline for understanding the error bounds in the dependent ciphertext setting and to highlight the significant differences compared to the independent case. To demonstrate this, we present theoretical results under specific hypotheses and support our findings with experimental evidence. However, our current methods do not allow us to generalize these theorems to all circuits, and addressing this may require a different approach for future research.

Distribution The coefficients distribution is analyzed as in Section 4.1. Also in this case it is well-approximated by a Gaussian distribution. In Figure 6, we show the outcome for circuits of multiplicative depth 2 with (i) all the input

ciphertexts equal, (ii) the 16 input ciphertexts taken randomly from a set of 5 and (iii) independently-computed ciphertexts. As parameters, we took $t = 3$, $n = 2^{13}$, $\sigma = 3.19$, $q = 2^{149} + 1$, $\chi_s = \mathcal{U}_3$, $\eta = 2$.

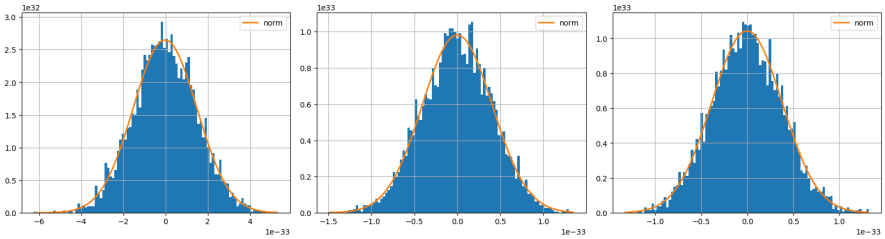


Fig. 6: (i) ks_{pval} 0.915409; (ii) ks_{pval} 0.50072; and (iii) ks_{pval} 0.92972.

Mean & Variance Analysis In the general case where the ciphertexts can be computed dependently, the computation of mean and variance changes only when it comes to additions and multiplication. We believe that it is still possible to prove Proposition 1, i.e. that the expected value of the coefficients of the invariant noise is 0, and experimental results confirm it. Note that this proposition is used in Proposition 8.

Regarding the variance computation, we have

Proposition 7 (Addition). *Let $\mathbf{c}, \mathbf{c}'$ be any two ciphertexts with invariant noises ν, ν', respectively. Then the variance of the error coefficients resulting from the addition of $\mathbf{c}$ and $\mathbf{c}'$ is*

$$\begin{aligned}\mathsf{Var}((\nu+\nu')|_i) &= \mathsf{Var}(\nu|_i) + \mathsf{Var}(\nu'|_i) + 2\mathsf{Cov}(\nu|_i, \nu'|_i) \\ &\leq \mathsf{Var}(\nu|_i) + \mathsf{Var}(\nu'|_i) + 2\sqrt{\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)} \\ &\leq 4\max(\mathsf{Var}(\nu|_i), \mathsf{Var}(\nu'|_i))\end{aligned}$$

Proof. The proof follows by the properties (k) and (m) of variance and covariance in Fact 1. □

Proposition 8 (Multiplication). *Let $\mathbf{c}, \mathbf{c}'$ be any two ciphertexts with invariant noises $\nu = \sum_{\iota_1=0}^{T_1} a_{\iota_1} s^{\iota_1}, \nu' = \sum_{\iota_2=0}^{T_2} a'_{\iota_2} s^{\iota_2}$, respectively. Assuming that $\mathbb{E}[\nu|_i] = \mathbb{E}[\nu'|_i] = 0$ for all i, the variance of the error coefficients resulting from the multiplication of $\mathbf{c}$ and $\mathbf{c}'$ satisfies*

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq \frac{t^2 n^2 V_s}{12}\Big(&\mathsf{Var}(\nu|_i) f(T_1+1) + \mathsf{Var}(\nu'|_i) f(T_2+1) + \\ &+2\sqrt{\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i) f(T_1+1) f(T_2+1)}\Big) + \mathsf{Var}((\nu\nu')|_i)\end{aligned} \tag{38}$$

with $f(i)$ as in Heuristic 1.

Proof. Recall that $\nu_{\mathsf{mul}}(q_\ell) = -\nu\nu' + \nu\frac{t}{q'_\ell}(c'_0 + c'_1 s) + \nu'\frac{t}{q_\ell}(c_0 + c_1 s) + \frac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2)$, by Equation (9), hence by property (k) of Fact 1, we get $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ is equal to

$$\begin{aligned}
&\mathsf{Var}((\nu\nu')|_i) + \mathsf{Var}\big((\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i\big) + \mathsf{Var}\big((\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i\big) \\
&+ \mathsf{Var}\big((\tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2))|_i\big) + 2\mathsf{Cov}((-\nu\nu')|_i, (\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i) \\
&+ 2\mathsf{Cov}((-\nu\nu')|_i, (\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i) + 2\mathsf{Cov}((-\nu\nu')|_i, \tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2)|_i) \\
&+ 2\mathsf{Cov}((\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, (\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i) \\
&+ 2\mathsf{Cov}((\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, \tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2)|_i) \\
&+ 2\mathsf{Cov}((\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i, \tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2)|_i)
\end{aligned}$$

Except $\mathsf{Cov}((\nu\frac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, (\nu'\frac{t}{q_\ell}(c_0 + c_1 s))|_i)$, all the covariances go to 0. Indeed, by Equation (1) and property (e) of Fact 1, they can be written as sums of covariances that satisfy the assumptions of property (h) of Fact 1. We get

$$\begin{aligned}
\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) = \mathsf{Var}((\nu\nu')|_i) + \mathsf{Var}\big((\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i\big) + \mathsf{Var}\big((\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i\big) \\
+ \mathsf{Var}\big((\tfrac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2))|_i\big) + 2\mathsf{Cov}((\nu\tfrac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, (\nu'\tfrac{t}{q_\ell}(c_0 + c_1 s))|_i).
\end{aligned}$$

Analogously to the proof of Theorem 1 (page 21), we get that the term $\mathsf{Var}\big((\frac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2))|_i\big)$ is negligible and we have the following bound $\mathsf{Var}\big((\nu\frac{t}{q'_\ell}(c'_0 + c'_1 s))|_i\big) + \mathsf{Var}\big((\nu'\frac{t}{q_\ell}(c_0 + c_1 s))|_i\big) \leq \frac{t^2 n^2 V_s}{12}\big(\mathsf{Var}(\nu|_i) f(T_1 + 1) + \mathsf{Var}(\nu'|_i) f(T_2 + 1)\big)$.

Finally, applying property (m) of Fact 1 to the covariance term, we obtain $\mathsf{Cov}((\nu\frac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, (\nu'\frac{t}{q_\ell}(c_0 + c_1 s))|_i) \leq \frac{t^2 n^2 V_s}{12}\sqrt{\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i) f(T_1+1) f(T_2+1)}$, hence the thesis

$$\begin{aligned}
\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq \frac{t^2 n^2 V_s}{12}\Big(\mathsf{Var}(\nu|_i) f(T_1 + 1) + \mathsf{Var}(\nu'|_i) f(T_2 + 1) + \\
+ 2\sqrt{\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i) f(T_1 + 1) f(T_2 + 1)}\Big) + \mathsf{Var}((\nu\nu')|_i).
\end{aligned}$$

□

Note that Equation (38) contains the same terms of Equation (22) with the addition of two terms: the last term, which comes from the covariance $\mathsf{Cov}((\nu\frac{t}{q'_\ell}(c'_0 + c'_1 s))|_i, (\nu'\frac{t}{q_\ell}(c_0 + c_1 s))|_i)$; and $\mathsf{Var}((\nu\nu')|_i)$. It is important to observe that, unlike the independent case, we are not able to give an estimation of $\mathsf{Var}((\nu\nu')|_i)$ starting from $\mathsf{Var}(\nu|_i)$ and $\mathsf{Var}(\nu'|_i)$.

Example: Error Bounds for Circuits in Figure 3 with Identical Inputs. Let us consider, as for the independent case, the circuits shown in Figure 3 with the modification that all input ciphertexts $\mathbf{c}_i$ are equal. In this case, we assume that

the variance term $\mathsf{Var}((\nu\nu')|_i)$ is negligible compared to the other terms. Thus, under this assumption, Equation (38) becomes

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) \leq \frac{t^2n^2V_s}{12}\big(&\mathsf{Var}(\nu|_i)f(T_1+1)+\mathsf{Var}(\nu'|_i)f(T_2+1)\\&+2\sqrt{\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)f(T_1+1)f(T_2+1)}\big)\\\leq \frac{t^2n^2V_s}{3}&\mathsf{Var}(\nu|_i)f(T_1+1). \qquad (39)\end{aligned}$$

Note that the last inequality arises from the fact that the input ciphertexts are equal; thus, we have $\nu = \nu'$ and $T_1 = T_2$.

This assumption is confirmed by the practical experiments shown in Tables 10 to 12, where we compare the error analysis and we provide the noise budget on the error bound. However, it is important to note that for different types of circuits, $\mathsf{Var}((\nu\nu')|_i)$ may not always be negligible.

In Tables 10 to 12, we present the noise budget of the error bounds. The tag "can" denotes the state-of-the-art analysis carried out with the canonical norm, "our" presents the results obtained with the average-case approach presented in this section under the hypothesis that $\mathsf{Var}((\nu\nu')|_i)$ is negligible and "exp" shows the observed values from OpenFHE [2] library with 2^{15} polynomial samples. As in Section 6, we set $||\nu||_\infty \leq D\sqrt{2V}$ with $D = 6$ and V variance of each coefficient of ν. Moreover, we chose $t = 65537$, $n = 2^{12}, \ldots, 2^{15}$ and q set by the library. We highlight the results in black in the tables when the security level is at least 128-bit, and in grey when it is below this threshold. We use Hybrid key switching and HPSPOVERQ multiplication and fixed $\chi_s = \chi_u = \mathcal{U}_3$, and $\chi_e = \mathcal{DG}(0, \sigma^2)$, with $\sigma = 3.19$.

	Addition					Multiplication				
	maximum value			mean value		maximum value			mean value	
n	can	our	exp	our	exp	can	our	exp	our	exp
2^{12}	25.5	31.0	31.7	34.1	34.4	57.0	64.5	65.9	67.6	68.2
2^{13}	24.5	30.5	31.6	33.6	33.9	55.0	63.1	64.1	66.1	66.7
2^{14}	23.5	30.0	31.0	33.1	33.4	53.0	61.6	62.7	64.7	65.2
2^{15}	22.5	29.5	30.5	32.6	32.9	51.0	60.0	61.3	63.1	63.7

Table 10: Addition and multiplication of a fresh ciphertext with itself.

Just as in the independent case, for dependent ciphertexts, our method provides more precise results, closely matching with experimental observations and offering substantial improvements over previous approaches. For example, for circuits with a depth of 3, our bounds are up to 12.4 bits tighter compared to the state-of-the-art, and for circuits with depth 5, they are up to 15.2 bits tighter, with a maximum deviation of 1.8 bits from the actual values.

	Multiplicative depth 2					Multiplicative depth 3				
	maximum value			mean value		maximum value			mean value	
n	can	our	exp	our	exp	can	our	exp	our	exp
2^{12}	21.5	30.8	32.2	33.9	34.7	49.0	59.9	61.2	63.0	63.7
2^{13}	18.5	28.4	29.8	31.4	32.2	45.0	56.5	57.9	59.6	60.2
2^{14}	15.5	25.9	27.1	29.0	29.6	41.0	53.0	54.2	56.1	56.7
2^{15}	12.5	23.3	24.6	26.4	27.1	37.0	49.4	50.7	52.5	53.2

Table 11: Comparison in the Base Model of depth 2 and 3 with $\alpha = 1$ and $\eta = 8$.

	Multiplicative depth 4					Multiplicative depth 5				
	maximum value			mean value		maximum value			mean value	
n	can	our	exp	our	exp	can	our	exp	our	exp
2^{12}	16.5	28.8	30.3	31.9	32.7	44.0	57.7	59.3	60.8	61.8
2^{13}	11.5	24.4	25.8	27.5	28.8	38.0	52.2	53.4	55.3	55.7
2^{14}	6.5	20.0	21.1	23.1	23.5	32.0	46.8	48.4	49.9	50.8
2^{15}	1.5	15.4	17.1	18.5	19.4	26.0	41.2	43.0	44.3	45.4

Table 12: Comparison in the Base Model of depth 4 and 5 with $\alpha = 1$ and $\eta = 8$.

It is worth noting that comparing the dependent (Tables 10 to 12) and the independent (Tables 7 to 9) case, reveals some differences. Specifically, we observe the following key points:

- The error is moderately larger when ciphertexts are dependent. In particular, the gap between the dependent and independent case grows as the number of multiplications increases, as reflected in the noise budget in the Tables 8, 9, 11 and 12. Indeed, after three multiplications, the difference can be up to 6 bits and after 5, the different is at most 10 bits (and this is also reflected in the experimental one where it is up to 6.3 bits and 10.4 bits, respectively).
- The difference in the noise budget is obviously reflected in the ciphertext modulus q. Specifically, the difference in q between the dependent (Figure 7) and independent (Figure 4) cases can be as large as 10 bits when the multiplicative depth is 5, leading to a 5.3% increase in the ciphertext modulus.

n	2^{12}	2^{13}	2^{14}	2^{15}
can	132.6	136.6	140.6	144.6
our	120.5	123.9	127.4	131.0

(a) Circuit of depth 3, $\eta = 8$ and $\alpha = 1$.

n	2^{12}	2^{13}	2^{14}	2^{15}
can	198.5	204.5	210.5	216.5
our	182.7	188.2	193.6	199.2

(b) Circuit of depth 5, $\eta = 8$ and $\alpha = 1$.

Fig. 7: Comparison of $\log_2(q)$ in the Base Model circuit (setting $D = 8$).

To conclude this section, we point out that choosing the correct parameters based on the specific case (whether dependent or independent) is crucial to ensure both correctness and, more importantly, security. Using parameters suited for one case in the other may compromise these guarantees. We believe that

with the correct parameter set (and setting $D = 8$), the recent attack [12] is not feasible.

8 Conclusion

Our average-case noise analysis significantly outperforms the state-of-the-art methods for the BFV scheme. Our approach provides very precise estimations for any multiplicative depth; in the examples, they deviate by no more than 2.5 bits from the values observed in experiments. This level of precision results in considerably smaller bounds on the ciphertext modulus, which translates into better performance.

In addition, the introduction of simple closed formulas for correctness simplifies the parameter selection. The development of the first automated parameter generation tool for BFV makes the scheme accessible to a wider range of users while still ensuring security, correctness, and high efficiency.

Moreover, we establish a baseline for understanding error bounds in dependently computed ciphertexts, highlighting key differences from the independent case through both theoretical and experimental results. This underlines the necessity of adjusting parameters according to the nature of the ciphertexts, as using those optimized for independent ciphertexts in the dependent case can open up the risk of recovery attacks.

Future work. It is worth noting that this approach is expected to be adaptable to BGV and CKKS schemes. In particular, we are currently in the process of developing and implementing this approach for the BGV scheme.

References

1. Acar, A., Aksu, H., Uluagac, A.S., Conti, M.: A survey on homomorphic encryption schemes: Theory and implementation. ACM Computing Surveys (CSUR) **51**(4), 1–35 (2018)
2. Al Badawi, A., Bates, J., Bergamaschi, F., Cousins, D.B., Erabelli, S., Genise, N., Halevi, S., Hunt, H., Kim, A., Lee, Y., et al.: OpenFHE: Open-Source Fully Homomorphic Encryption Library. In: Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography. pp. 53–63 (2022)
3. Albrecht, M.R., Chase, M., Chen, H., Ding, J., Goldwasser, S., Gorbunov, S., Halevi, S., Hoffstein, J., Laine, K., Lauter, K., Lokam, S., Micciancio, D., Moody, D., Morrison, T., Sahai, A., Vaikuntanathan, V.: Homomorphic encryption security standard. Tech. rep., HomomorphicEncryption.org, Toronto, Canada (2018)
4. Bajard, J.C., Eynard, J., Hasan, M.A., Zucca, V.: A full RNS variant of FV like somewhat homomorphic encryption schemes. In: International Conference on Selected Areas in Cryptography. pp. 423–442 (2016)
5. Bergerat, L., Boudi, A., Bourgerie, Q., Chillotti, I., Ligier, D., Orfila, J.B., Tap, S.: Parameter Optimization and Larger Precision for (T)FHE. Journal of Cryptology **36**(3), 28 (2023)

6. Bossuat, J.P., Cammarota, R., Cheon, J.H., Chillotti, I., Curtis, B.R., Dai, W., Gong, H., Hales, E., Kim, D., Kumara, B., et al.: Security guidelines for implementing homomorphic encryption. Cryptology ePrint Archive (2024)
7. Brakerski, Z.: Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. In: Advances in Cryptology – CRYPTO 2012. pp. 868–886 (2012)
8. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. ACM Transactions on Computation Theory (TOCT) **6**(3), 1–36 (2014)
9. Brakerski, Z., Vaikuntanathan, V.: Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In: Advances in Cryptology – CRYPTO 2011. pp. 505–524 (2011)
10. Checri, M., Sirdey, R., Boudguiga, A., Bultel, J.P., Choffrut, A.: On the practical cpad security of "exact" and threshold fhe schemes and libraries. Cryptology ePrint Archive (2024)
11. Chen, H., Han, K.: Homomorphic lower digits removal and improved FHE bootstrapping. In: Nielsen, J.B., Rijmen, V. (eds.) Advances in Cryptology – EUROCRYPT 2018. pp. 315–337. Springer International Publishing (2018)
12. Cheon, J.H., Choe, H., Passelègue, A., Stehlé, D., Suvanto, E.: Attacks against the indcpa-d security of exact fhe schemes. Cryptology ePrint Archive (2024)
13. Cheon, J.H., Costache, A., Moreno, R.C., Dai, W., Gama, N., Georgieva, M., Halevi, S., Kim, M., Kim, S., Laine, K., et al.: Introduction to homomorphic encryption and schemes. Protecting Privacy through Homomorphic Encryption pp. 3–28 (2021)
14. Cheon, J.H., Han, K., Kim, A., Kim, M., Song, Y.: A full RNS variant of approximate homomorphic encryption. In: International Conference on Selected Areas in Cryptography – SAC 2018. pp. 347–368. Springer (2018)
15. Cheon, J.H., Kim, A., Kim, M., Song, Y.: Homomorphic Encryption for Arithmetic of Approximate Numbers. In: Advances in Cryptology – ASIACRYPT 2017. pp. 409–437 (2017)
16. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds. In: Advances in Cryptology – ASIACRYPT 2016. pp. 3–33 (2016)
17. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology **33**(1), 34–91 (2020)
18. Costache, A., Smart, N.P.: Which ring based somewhat homomorphic encryption scheme is best? In: Topics in Cryptology – CT-RSA 2016. pp. 325–340 (2016)
19. Costache, A., Curtis, B.R., Hales, E., Murphy, S., Ogilvie, T., Player, R.: On the precision loss in approximate homomorphic encryption pp. 325–345 (2023)
20. Costache, A., Laine, K., Player, R.: Evaluating the effectiveness of heuristic worst-case noise analysis in FHE. In: Computer Security – ESORICS 2020. pp. 546–565 (2020)
21. Costache, A., Nürnberger, L., Player, R.: Optimisations and Tradeoffs for HElib. In: Topics in Cryptology – CT-RSA 2023. pp. 29–53 (2023)
22. Di Giusto, A., Marcolla, C.: Breaking the power-of-two barrier: noise estimation for BGV in NTT-friendly rings. ePrint Archive, Paper 2023/783 (2023)
23. Ducas, L., Micciancio, D.: FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology – EUROCRYPT 2015. pp. 617–640. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)

24. Fan, J., Vercauteren, F.: Somewhat practical fully homomorphic encryption. ePrint Archive (2012)
25. Geelen, R., Vercauteren, F.: Bootstrapping for bgv and bfv revisited. Journal of Cryptology **36**(2), 12 (2023)
26. Gentry, C.: A fully homomorphic encryption scheme. Stanford university (2009)
27. Gentry, C., Halevi, S., Smart, N.P.: Homomorphic Evaluation of the AES Circuit. In: Advances in Cryptology – CRYPTO 2012. pp. 850–867 (2012)
28. Halevi, S., Polyakov, Y., Shoup, V.: An improved RNS variant of the BFV homomorphic encryption scheme. In: Topics in Cryptology – CT-RSA 2019. pp. 83–105 (2019)
29. Iliashenko, I.: Optimisations of fully homomorphic encryption. PhD thesis (2019)
30. Kim, A., Polyakov, Y., Zucca, V.: Revisiting homomorphic encryption schemes for finite fields. In: Advances in Cryptology–ASIACRYPT 2021. pp. 608–639 (2021)
31. Kirshanova, E., Marcolla, C., Rovira, S.: Guidance for efficient selection of secure parameters for fully homomorphic encryption. In: International Conference on Cryptology in Africa. pp. 376–400. Springer (2024)
32. Liu, Z., Wang, Y.: Relaxed functional bootstrapping: A new perspective on bgv/bfv bootstrapping. Cryptology ePrint Archive (2024)
33. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: Advances in Cryptology – EUROCRYPT 2010. pp. 1–23 (2010)
34. Marcolla, C., Sucasas, V., Manzano, M., Bassoli, R., Fitzek, F.H., Aaraj, N.: Survey on fully homomorphic encryption, theory, and applications. Proceedings of the IEEE **110**(10), 1572–1609 (2022)
35. Martins, P., Sousa, L., Mariano, A.: A survey on fully homomorphic encryption: An engineering perspective. ACM Computing Surveys (CSUR) **50**(6), 1–33 (2017)
36. Mono, J., Marcolla, C., Land, G., Güneysu, T., Aaraj, N.: Finding and evaluating parameters for BGV. In: International Conference on Cryptology in Africa. pp. 370–394 (2023)
37. Murphy, S., Player, R.: A central limit approach for ring-LWE noise analysis. IACR Communications in Cryptology **1**(2) (2024)
38. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM) **56**(6), 1–40 (2009)
39. Microsoft SEAL (release 3.4). https://github.com/Microsoft/SEAL (Oct 2019)

A Proof of Lemma 1 and Lemma 2

Fresh ciphertext By Equation (5), we can write $\nu_{\mathsf{clean}} = a_0 + a_1 s$ with $a_0 = \frac{t}{q}(\varepsilon + eu + e_0)$ and $a_1 = \frac{t}{q}e_1$. Since the coefficients of all the polynomials in ν_{clean} are sampled independently from symmetric distributions (with expected value 0) we have $\mathbb{E}[a_0|_i] = \frac{t}{q}(\mathbb{E}[\varepsilon|_i] + \sum_{j=0}^{n-1} \xi(i,j)\mathbb{E}[e|_j]\mathbb{E}[u|_{i-j}] + \mathbb{E}[e_0|_i]) = 0$ and $\mathbb{E}[a_1|_i] = \frac{t}{q}\mathbb{E}[e_1|_i] = 0$, by (a) and (d) of Fact 1. Moreover, all the covariances are equal to zero. Indeed:

- $\mathsf{Cov}(a_0|_{i_1}, a_1|_{i_2}) = 0$ and $\mathsf{Cov}(a_1|_{i_1}, a_1|_{i_2}) = 0$, for $i_1 \neq i_2$, since the terms are independent, thus we apply property (d) of Fact 1.
- $\mathsf{Cov}(a_0|_{i_1}, a_0|_{i_2}) = 0$, by the bilinearity of the covariance (property (e) of Fact 1).

– $\mathsf{Cov}((eu)|_{i_1}, (eu)|_{i_2}) = 0$. Indeed, it is a sum of terms $\mathsf{Cov}(e|_{j_1}u|_{i_1-j_1}, e|_{j_2}u|_{i_2-j_2})$, where $j_1 \neq j_2$ or $i_1 - j_1 \neq i_2 - j_2$, hence we can apply property (g) of Fact 1.

Finally, by the properties of the variance (j), (k) and (l) of Fact 1, $\mathsf{Var}(a_1|_i) = \frac{t^2}{q^2}V_e$ and $V(a_0|_i) = \frac{t^2}{q^2}\big(\mathsf{Var}(\varepsilon|_i) + \sum_{j=0}^{n-1}\mathsf{Var}(e|_j)\mathsf{Var}(u|_{i-j}) + \mathsf{Var}(e_0|_i)\big) = \frac{t^2}{q^2}\big(\frac{1}{12} + nV_eV_u + V_e\big)$, where V_e, V_u, V_s denotes the variances of any element sampled from the distributions χ_e, χ_u, χ_s, respectively and $\mathsf{Var}(\varepsilon|_i) = \frac{1}{12}$ since $\varepsilon = -\frac{[qm]_t}{t}$ and $[qm]_t$ can be consider a random element from the uniform distribution $\mathcal{U}_t$.

Addition Let ν, ν' be the errors of two independently-computed ciphertexts, then $\nu = \sum_\iota a_\iota s^\iota$, $\nu' = \sum_{\iota'} a'_{\iota'} s^{\iota'}$ with a_ι, $a'_{\iota'}$ independent for any ι, ι'. By Equation (6), it follows that $\nu_{\mathsf{add}} = \nu + \nu' = \sum_\iota (a_\iota + a'_\iota)s^\iota$, where $\mathbb{E}[(a_\iota + a'_\iota)|_i] = \mathbb{E}[a_\iota|_i] + \mathbb{E}[a'_\iota|_i] = 0$ and $\mathsf{Cov}((a_{\iota_1} + a'_{\iota_1})|_{i_1}, (a_{\iota_2} + a'_{\iota_2})|_{i_2}) = 0$ if $\iota_1 \neq \iota_2$ or $i_1 \neq i_2$. Indeed, by the bilinearity of the covariance, it splits in $\mathsf{Cov}(a_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) + \mathsf{Cov}(a_{\iota_1}|_{i_1}, a'_{\iota_2}|_{i_2}) + \mathsf{Cov}(a'_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) + \mathsf{Cov}(a'_{\iota_1}|_{i_1}, a'_{\iota_2}|_{i_2})$, where the variables in each pairs are either uncorrelated by induction hypothesis or independent because they come from different ciphertexts. Finally, $\mathsf{Var}((a_\iota + a'_\iota)|_i) = \mathsf{Var}(a_\iota|_i) + \mathsf{Var}(a'_\iota|_i)$ does not depend on the coefficient i by inductive hypothesis.

Modulo switch & Key switch The proof is analogous to the addition case, we only need to do some observation.

In the modulo switch, as described in Equation (8), the noise ν increased by the quantity $\nu_{\mathsf{ms}}(q'_\ell) = \frac{t}{q'_\ell}(\varepsilon_0 + \varepsilon_1 s)$ with $\varepsilon_i = -\frac{[q'_\ell c_i]_{q_\ell}}{q_\ell}$. Hence, for $\iota = 0, 1$, $\frac{t}{q'_\ell}\varepsilon_\iota|_i$ is added to $a_\iota|_i$. The elements $\frac{t}{q'_\ell}\varepsilon_\iota|_i$ can be considered as sampled independently at random from $\mathcal{U}_{(-0.5,0.5)}$, since c_ι is indistinguishable from a random element in $\mathcal{R}_{q_\ell}$. Therefore, they have mean 0 and variance $\frac{t^2}{12q'^2_\ell}$.

For the key switch, we also have to make an observation about the first term. As described in (10)–(14), each key switching equation includes one of the terms $[d_2]_{r_i}$, d_2 or $[d_2]_{\tilde{r}_i}$. Similar to the modulo switch case, these terms can be considered as sampled uniformly at random from $\mathcal{U}_{r_i}, \mathcal{U}_{q_\ell}, \mathcal{U}_{\tilde{r}_i}$, respectively, resulting in a covariance 0 thanks to the property (g) of Fact 1. Finally, the added variances are

BV key switch. Since $r_i \approx \sqrt[k]{q}$,

$$\mathsf{Var}\Big(\big(\frac{t}{q_\ell}\sum_{i=1}^{k_\ell}[d_2]_{r_i}e_i\big)\big|_j\Big)1_{\iota=0} = \frac{t^2}{q_\ell^2}\sum_{i=1}^{k_\ell} n\frac{r_i^2}{12}V_e 1_{\iota=0} \approx \frac{t^2}{12q_\ell^2}k_\ell\sqrt[k]{q^2}nV_e 1_{\iota=0}.$$

GHS key switch. Since $P \approx q \geq q_\ell$, and $\mathsf{Var}(\frac{t}{q_\ell}\varepsilon_1|_i)1_{\iota=1} = \frac{t^2}{12q_\ell^2}1_{\iota=1}$,

$$\mathsf{Var}\Big(\frac{t}{q_\ell}\Big(\frac{d_2e'}{P}+\varepsilon_0\Big)\Big|_i\Big)1_{\iota=0} = \frac{t^2}{12q_\ell^2}\left(\frac{nq_\ell^2V_e}{P^2}+1\right)1_{\iota=0} \leq \frac{t^2}{12q_\ell^2}(nV_e+1)1_{\iota=0}.$$

GHS-RNS key switch. Analogously to GHS key switch,

$$\mathsf{Var}(\frac{t}{q_\ell}(\frac{(d_2+uq_\ell)e'}{P}+\varepsilon_0)|_i)\mathbf{1}_{\iota=0} \leq \frac{t^2}{12q_\ell^2}(n(k_\ell+2)V_e+1)\mathbf{1}_{\iota=0},$$

since $u = \lfloor\sum_{i=1}^{k_\ell}[[d_2]_{r_i}[(\frac{q_\ell}{r_i})^{-1}]_{r_i}]_{r_i}\frac{1}{r_i}\rceil$, hence $\mathsf{Var}(u|_i) = \frac{k_\ell+1}{12}$. Moreover, $\mathsf{Var}(\frac{t}{q_\ell}\varepsilon_1|_i)\mathbf{1}_{\iota=1} = \frac{t^2}{12q_\ell^2}\mathbf{1}_{\iota=1}$.

Hybrid key switch. Since $\tilde{r_i} \approx \sqrt[\omega]{q_\ell}$ and $P \approx \sqrt[\omega]{q}$, by Equation (13),

$$\begin{aligned}\mathsf{Var}\left(\frac{t}{q_\ell}(\frac{\sum_{i=1}^{\omega}[d_2]_{\tilde{r_i}}e_i}{P}+\varepsilon_0)|_i\right) &= \frac{t^2}{12q_\ell^2}\left(\frac{nV_e\sum_{i=1}^{\omega}\tilde{r_i}^2}{P^2}+1\right)\\ &= \frac{t^2}{12q_\ell^2}\Big(\frac{\omega nV_e\sqrt[\omega]{q_\ell^2}}{\sqrt[\omega]{q^2}}+1\Big) \leq \frac{t^2}{12q_\ell^2}\Big(\omega nV_e+1\Big),\end{aligned}$$

and $\mathsf{Var}\left(\frac{t}{q_\ell}\varepsilon_1|_i\right) = \frac{t^2}{12q_\ell^2}$.

Hybrid-RNS key switch. The only difference with the hybrid key switch is that $[d_2]_{\tilde{r_i}}$ becomes $([d_2]_{\tilde{r_i}}+u_i\tilde{r_i})$, with $u_i = \lfloor\sum_{r_j|\tilde{r_i}}[[d_2]_{r_j}[(\frac{\tilde{r_i}}{r_j})^{-1}]_{r_j}]_{r_j}\frac{1}{r_j}\rceil$. Thus, its variance is $\frac{\tilde{r_i}^2}{12}+\big(\sum_{r_j|\tilde{r_i}}\frac{r_j^2}{12}\frac{1}{r_j^2}+\frac{1}{12}\big)\tilde{r_i}^2 = \frac{\tilde{r_i}^2}{12}(\frac{k}{\omega}+2)$. Hence,

$$\mathsf{Var}\left(\frac{t}{q_\ell}\left(\frac{\sum_{i=1}^{\omega}([d_2]_{\tilde{r_i}}+u_i\tilde{r_i})e_i}{P}+\varepsilon_0\right)\Big|_i\right) = \frac{t^2}{12q_\ell^2}\Big(nV_e(k+2\omega)+1\Big).$$

Constant multiplication Let $\nu = \sum_\iota a_\iota s^\iota$ be any invariant noise and α a polynomial with coefficients sampled randomly from $\mathcal{U}_t$, then $\alpha\nu = \sum_\iota(\alpha a_\iota)s^\iota$, since α is constant in s, $\mathbb{E}[\alpha|_i] = 0$ and $\mathsf{Var}(\alpha|_i) \approx (t^2-1)/12$. Moreover, by the properties of the expected value and since the coefficients of α and a_ι are independent with mean 0, $\mathbb{E}\big[(\alpha a_\iota)|_i\big] = 0$. Finally, by the bilinearity of the covariance (property (e) of Fact 1) and by property (g) of Fact 1 applied to each summand $\mathsf{Cov}(\alpha|_{j_1}a_{\iota_1}|_{i_1-j_1},\ \alpha|_{j_2}a_{\iota_2}|_{i_2-j_2})$, we have $\mathsf{Cov}\big((\alpha a_{\iota_1})|_{i_1},(\alpha a_{\iota_2})|_{i_2}\big) = 0$. Analogously to the previous cases, $\mathsf{Var}((\alpha a_\iota)|_i) = \sum_{j=0}^{n-1}V(\alpha|_j)\cdot V(a_\iota|_{i-j}) = \frac{(t^2-1)n}{12}V(a_\iota|_{i-j})$ does not depend on the coefficient i by inductive hypothesis.

Multiplication Let $\nu = \sum_j a_j s^j$ and $\nu' = \sum_k a'_k s^k$ be the errors of two independently-computed ciphertexts, then $\nu\nu' = \sum_\iota\sum_{j+k=\iota}a_j a'_k s^\iota$. Note that the ι-th element of $\nu\nu'$, as a polynomial in s, is $\sum_{j+k=\iota}a_j a'_k$ where a_j, a'_k are independent for any j, k. It follows that $\mathbb{E}\big[\big(\sum_{j+k=\iota}a_j a'_k\big)|_i\big] = \sum_{j+k=\iota}\sum_{l=0}^{n-1}\xi(i,l)\mathbb{E}[a_j|_l]\mathbb{E}[a'_k|_{i-l}] = 0$. Furthermore, by bilinearity of the covariance, $\mathsf{Cov}((\sum_{j_1+k_1=\iota_1}a_{j_1}a'_{k_1})|_{i_1},(\sum_{j_2+k_2=\iota_2}a_{j_2}a'_{k_2})|_{i_2})$ is a linear combination of elements $\mathsf{Cov}(a_{j_1}|_{l_1}a'_{k_1}|_{i_1-l_1},a_{j_2}|_{l_2}a'_{k_2}|_{i_2-l_2})$. For $\iota_1\neq\iota_2$ or $i_1\neq i_2$, all these terms are null, hence the thesis, since we fall in the same case as in constant multiplication.

Analogously, this holds for $\nu\frac{t}{q'_\ell}(c'_0+c'_1s), \nu'\frac{t}{q_\ell}(c_0+c_1s)$. Finally, we have that the covariance of different summands in ν_{mul} is 0, hence the conditions hold also for $\nu_{\mathsf{mul}} = -\nu\nu'+\nu\frac{t}{q'_\ell}(c'_0+c'_1s)+\nu'\frac{t}{q_\ell}(c_0+c_1s)+\frac{t}{q}(\varepsilon_0+\varepsilon_1s+\varepsilon_2s^2)$. □

B Parameterized calculation of the function $f(\iota)$

In this section, we present the results obtained for the function $f(\iota)$ in Heuristic 1 across different values of n and χ_s.

Let us recall Heuristic 1: for $\iota \geq 2$, Equation (24) is well-approximated by the function $f(\iota)$, specifically:

$$f(\iota) = -e^{\alpha - \beta\iota - \gamma\iota^2} + \delta \approx \frac{\sum_{j=0}^{n-1} \mathbb{E}[s^{\iota}|_j^2]}{\sum_{j_1=0}^{n-1} \mathbb{E}[s^{\iota-1}|_{j_1}^2] \sum_{j_2=0}^{n-1} \mathbb{E}[s|_{j_2}^2]}, \tag{40}$$

where α, β, γ, and δ depend on the distribution χ_s and the ring dimension n (see Tables 13 to 15).

n	α	β	γ	δ
2^{12}	2.9163	0.0394	0.0036	18.8210
2^{13}	2.9069	0.0177	0.0051	19.5385
2^{14}	2.9568	0.0547	0.0025	18.8333
2^{15}	2.9525	0.0176	0.0043	20.5393

Table 13: $\chi_s = \mathcal{ZO}(1/2)$

n	α	β	γ	δ
2^{12}	2.9000	0.0157	0.0051	19.5356
2^{13}	2.9340	0.0042	0.0055	20.7063
2^{14}	2.9138	0.0290	0.0039	19.2973
2^{15}	2.9511	0.0129	0.0046	20.7263

Table 14: $\chi_s = \mathcal{DG}(0, \sigma^2)$

n	α	β	γ	δ
2^{12}	2.8367	0.0395	0.0032	17.6662
2^{13}	2.8331	0.0184	0.0047	18.3825
2^{14}	2.8964	0.0558	0.0023	17.8335
2^{15}	2.9036	0.0466	0.0023	18.4545

Table 15: $\chi_s = \mathcal{HWT}(64)$

We computed their values with Python function *curve_fit*[9].

The experiments were conducted for $\iota \leq 300$, considering 25000 samples (for $n = 2^{12}$ and $n = 2^{13}$) and 10000 samples (for $n = 2^{14}$ and $n = 2^{15}$). After $\iota \approx 100$, we observe an asymptote. Note that, in general, the BFV scheme utilizes a ternary distribution for the secret key. However, in setups expecting the bootstrapping, the $\mathcal{HWT}(h)$ distribution is preferred [11,25,32]. In light of this, we provide plots for both scenarios. In particular, we show $f(\iota)$ as n varies, with $\chi_s = \mathcal{U}_3$ (Figure 8) and $\chi_s = \mathcal{HWT}(64)$ (Figure 9). The grey dots represent the experimental values from the right-hand side of Equation (24), while the green line shows the approximation of $f(\iota)$ as in Equation (40). Note that for clarity, we have plotted only 50 data points.

[9] https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html

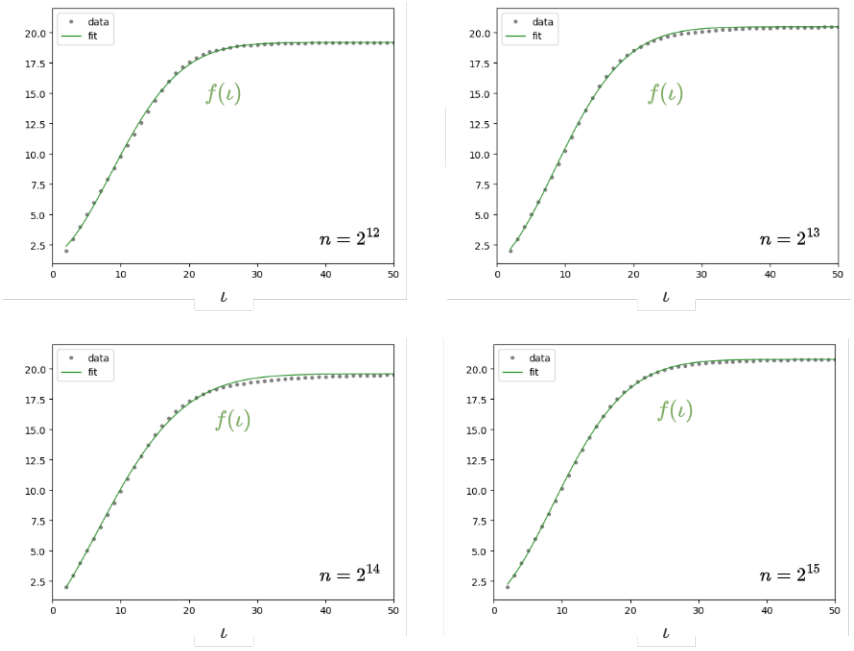


Fig. 8: Examples of $f(\iota)$ fitting the points for $\chi_s = \mathcal{U}_3$, where f is as in (40).

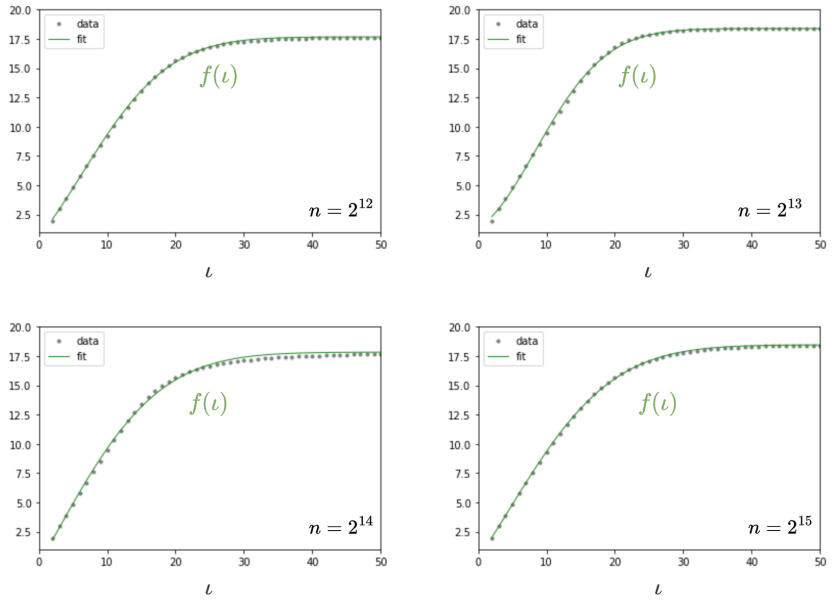


Fig. 9: Examples of $f(\iota)$ fitting the points for $\chi_s = \mathcal{HWT}(64)$, where f is as in Equation (40).

C Proof of Lemma 4

1. Let us fix ι_1 and consider ι_2. Since $f(\iota)$ is an increasing function and $\iota_2 \leq T_2$, we have $f(\iota_2 + i) \leq f(T_2 + i)$, then

$$\frac{g(\iota_1 + \iota_2)}{g(\iota_2)} = f(\iota_2 + 1) \cdots f(\iota_1 + \iota_2) \leq f(T_2 + 1) \cdots f(\iota_1 + T_2) = \frac{g(\iota_1 + T_2)}{g(T_2)}.$$

It follows, in particular, $\mathbb{F}(\iota_1, \iota_2) \leq F(\iota_1, T_2)$.
We get the thesis analogously on ι_1.

2. We divide the proof in two parts:

- proving that, defined $T = T_1 + T_2$,

$$\frac{F(T_1, T_2)}{f(T_1 + 1)f(T_2 + 1)} = \frac{g(T_1 + T_2)}{g(T_1 + 1)g(T_2 + 1)} \leq \frac{g(T)}{g(\lfloor T/2 \rfloor + 1)g(\lceil T/2 \rceil + 1)};$$

- proving the existence of the limit

$$K_n = \lim_{T \to +\infty} \frac{g(T)}{g(\lfloor T/2 \rfloor + 1)g(\lceil T/2 \rceil + 1)}$$

  and giving an good approximation for different values of n and χ_s.

Firstly, we prove $\frac{g(T_1+T_2)}{g(T_1+1)g(T_2+1)} \leq \frac{g(T)}{g(\lfloor T/2 \rfloor+1)g(\lceil T/2 \rceil+1)}$. For example, for $T = 4$, we have

$$\frac{g(4)}{g(1)g(5)} = \frac{1}{f(5)} < \frac{g(4)}{g(2)g(4)} = \frac{1}{f(2)} < \frac{g(4)}{g(3)g(3)} = \frac{f(4)}{f(2)f(3)}.$$

To simplify the notation, we assume wlog $T_1 \leq T_2$ and define $T = T_1 + T_2$, $\tau = \lfloor T/2 \rfloor$. It follows that $T - \tau = \lceil T/2 \rceil$ and $T_1 \leq \tau \leq T - \tau \leq T_2$. Then, we can write T_1, T_2 as $T_1 = \tau - k$, $T_2 = T - \tau + k$ for some $k \in \mathbb{N}$. Now,

$$\begin{aligned}
\frac{g(T_1 + T_2)}{g(T_1 + 1)g(T_2 + 1)} &= \frac{g(T)}{g(\tau - k + 1)g(T - \tau + k + 1)} \\
&= \frac{g(T)}{g(\tau + 1)g(T - \tau + 1)} \frac{f(\tau - k + 2) \cdots f(\tau + 1)}{f(T - \tau + 2) \cdots f(T - \tau + k + 1)} \leq \\
&\leq \frac{g(T)}{g(\tau + 1)g(T - \tau + 1)} = \frac{g(T)}{g(\lfloor T/2 \rfloor + 1)g(\lceil T/2 \rceil + 1)},
\end{aligned}$$

indeed $\frac{f(\tau-k+2)\cdots f(\tau+1)}{f(T-\tau+2)\cdots f(T-\tau+k+1)} = \prod_{j=0}^{k-1} \frac{f(\tau-k+2+j)}{f(T-\tau+2+j)} \leq 1$, as f is an increasing function and $\tau - k \leq T - \tau$.

Now, we prove the existence of the limit

$$\lim_{T \to +\infty} \frac{g(T)}{g(\lfloor T/2 \rfloor + 1)g(\lceil T/2 \rceil + 1)}.$$

Let

$$G(T) = \frac{g(T)}{g(\lfloor T/2 \rfloor + 1)g(\lceil T/2 \rceil + 1)} = \frac{1}{f(\tau+1)} \prod_{\iota=2}^{\tau} \frac{f(T-\tau+\iota)}{f(\iota)},$$

where $\tau = \lceil T/2 \rceil$. Let us define $s = T - \tau$, $c_\iota = f(\iota) = \delta - e^{\alpha - \beta\iota - \gamma\iota^2}$ and $\varepsilon_\iota = (1 - e^{-\beta s - \gamma(s^2+2s\iota)})e^{\alpha-\beta\iota-\gamma\iota^2}$. We have

$$G(T) = \frac{1}{c_{\tau+1}} \prod_{\iota=2}^{\tau} \frac{c_\iota + \varepsilon_\iota}{c_\iota}$$

Since $(c_\iota + \varepsilon_\iota)/c_\iota \geq 1$, we have $\prod_{\iota=2}^{\tau} (c_\iota + \varepsilon_\iota)/c_\iota \leq \exp\left(\sum_{\iota=2}^{\tau} \varepsilon_\iota/c_\iota\right)$.

Let $\bar{\varepsilon}_\iota = (1 - e^{-\beta s - \gamma(s^2+2s\iota)})e^{-\gamma\iota^2}$. The derivative of $\bar{\varepsilon}_\iota$ with respect ι is given by

$$\frac{d\bar{\varepsilon}_\iota}{d\iota} = 2\gamma e^{-\gamma\iota^2}(-\iota + (\iota + s)e^{-\beta s - \gamma(s^2+2s\iota)}).$$

So the sign of the derivative is given by the term $(-\iota + (\iota + s)e^{-\beta s - \gamma(s^2+2s\iota)})$ and for ι sufficiently large this is negative (if s is large, which is the case if T is large, then it is negative for any ι). Therefore, from a certain point, we have $\bar{\varepsilon}_{\iota+1} < \bar{\varepsilon}_\iota$. Noting that $\varepsilon_\iota = \bar{\varepsilon}_\iota e^{\alpha-\beta\iota}$ we have

$$\varepsilon_{\iota+1} < e^{-\beta}\varepsilon_\iota,$$

and so

$$\varepsilon_{\iota+1}/c_{\iota+1} < e^{-\beta}\varepsilon_\iota/c_\iota.$$

Since $e^{-\beta} < 1$, we get that the sum $\sum_{\iota=2}^{\tau} \varepsilon_\iota/c_\iota$ converges and thus the limit $\lim_{T\to+\infty} G(T)$ is finite.

We can show computationally that the limit $K_n = \lim_{T\to+\infty} G(T) < 40\,n$, for any $n = 2^\kappa$, where $\kappa \in \{12, \ldots, 15\}$.

□