# Finding and Evaluating Parameters for FV using the 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.** Fully homomorphic encryption is a revolutionary technology that allows arbitrary computations on encrypted data, providing privacy and security. State-of-the-art schemes such as the Fan-Vercauteren (FV) scheme are based on the Learning with Errors assumption and its variants. Thus, each ciphertext has an error that increases with each homomorphic operation. To maintain correctness, the error must be kept below a certain threshold, which requires a balance between security and computational efficiency. Therefore, choosing optimal, secure, and efficient parameters can be a challenging task, even for experts in a particular scheme.

In this paper, we present two major contributions to improve the parameter selection in the FV scheme. We perform the first average case analysis to estimate the error growth. Our method significantly improves on previous work in terms of accuracy and tightness of bounds. For a circuit with a multiplicative depth of only 3, our bounds are within 1.2 bits of the experimentally observed values while being up to 19 bits tighter than previous analyses.

In addition, we take advantage of our theoretical advances and propose the first parameter generation tool for the FV 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, FV, Parameter Generation, average-case noise analysis, PALISADE, OpenFHE

## 1 Introduction

Data privacy concerns are increasing significantly in the context of Internet of Things, cloud services, edge computing, artificial intelligence applications, and other applications enabled by the next generation networks. Homomorphic encryption addresses privacy challenges by enabling multiple operations on encrypted messages without decryption. Namely, a set of operations can be performed over ciphertexts such that these operations are reflected as additions and multiplications on the corresponding plaintexts. Thus, homomorphic encryption

allows data manipulation in the encrypted domain, or, in other words, privacy-preserving data processing [23].

The first Fully Homomorphic Encryption (FHE) scheme was introduced in 2009 by Gentry in [18]. In his Ph.D. 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 [7], FV [6, 17], TFHE [11], and CKKS [9] some of the most representative.

The security of most of the FHE schemes is based on the presumed intractability of the decision Learning with Errors (LWE) problem and its ring variant (RLWE), [2]. Informally, they consist of distinguishing equations perturbed by small noise from random tuples. The problem arising from this construction is noise growth. Indeed, in order to guarantee a correct decryption, the error added has to be small. However, it increases as long as operations are carried on. In particular, it grows exponentially when homomorphic multiplications are computed. To increase the number of supported operations, we could increase the ciphertext modulus $q$. However, a higher modulus also decreases the security level of the underlying scheme. On the other hand, to increase the security level, we can adopt a higher polynomial degree at the cost of efficiency. This required trade-off between security (small ciphertext modulus), and error margin (big ciphertext modulus) illustrates the difficulty of finding an optimal set of parameters for a specific FHE scheme.

*Related works.* One of the greatest challenges facing the FHE community is to find a set of parameters that must strike a balance between security and efficiency. Several efforts have been made in this direction. For instance, Bergerat *et al.* proposed a framework for efficiently selecting parameters in TFHE-like schemes [5], while Mono *et al.* [24] developed an interactive parameter generator for the leveled BGV scheme that supports an arbitrary circuit model. The Homomorphic Encryption Standard [1] uses the LWE Estimator[4] [2] and provides upper limits on the size of the ciphertext modulus for certain security levels $\lambda$ and polynomial degrees $d$ in the form of lookup tables. While the standard is crucial for fully homomorphic encryption (FHE) and provides a solid foundation for selecting Fan-Vercauteren (FV) parameters, it may not include all the necessary information required for implementation. For researchers familiar with FHE, this flexibility is valuable. For other users, however, this burden of choice increases the difficulty of using libraries securely.

Regarding the selection of the FV parameters, the state-of-the-art approach for establishing the theoretical bounds for the error growth is based on using

[4] The LWE Estimator is a software tool to determine the security level of LWE instances since it shows the timeline and fundamentals of the main lattice attacks proposed until the present time. Its successor is the Lattice Estimator (`https://github.com/malb/lattice-estimator`).

either the infinity norm [22] or the canonical norm [12, 14, 21]. The canonical norm is known to result in better parameters. However, both of these methods often yield overly conservative bounds. An alternative approach is an average-case analysis, which estimates the growth of the error on average and only sets a bound on its maximal value at the end of the computation. This method provides a predicted error closer to the actual errors observed in experimental results. The average-case approach was first proposed in [10] for computing bounds in the TFHE scheme. Recent works have introduced similar techniques for CKKS [13] and BGV [15, 25].

*Our contribution.* This paper aims to improve the current state of FV parameters selection by providing, for the first time, 1) an estimation of the noise in average-case scenarios and 2) a tool to automate the parameters generation in the FV scheme based on our theoretical findings.

More in detail, in this paper, we present a novel approach for the FV scheme based on average-case noise analysis. Our method differs from those proposed for the BGV and CKKS schemes. Specifically, the error coefficients are not independent, making it impossible to apply the Central Limit Theorem, as suggested in previous works [13, 25]. As a result, our analysis is more intricate, particularly for homomorphic multiplication.
To demonstrate the effectiveness of our method, we compare our bounds with prior heuristic noise analyses based on the canonical norm. For a circuit with a multiplicative depth of only 3, our approach provides bounds at least 19 bits tighter than previous analyses and only up to 1.2 bits lower than the practical computation.

Finally, we develop an interactive parameters generator for the FV scheme, which utilizes our theoretical results and the security formula proposed in [24]. 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.

The structure of the paper is the following:

- To facilitate 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 FV scheme.
- The core of the paper is Section 4, where we introduce our average-case approach.
- In Section 5, we investigate the error behavior in four different circuits (proposed in [24]) and two distinct circumstances: with and without modulus switching. As expected, the modulus switching technique increases the bounds. However, in Fact 1, we propose a set of parameters that achieve a ciphertext modulus similar to the one obtained without modulus switching while improving efficiency.

– Finally, in Section 6, we compare our average-case approach with prior bounds of FV noise growth. Additionally, we introduce our parameter generator to facilitate the selection of optimal parameters for the FV scheme.

## 2 Preliminaries

### 2.1 Notation

We start with the general notations we will use in the remainder of this work.

Let $f(x)$ be a monic irreducible polynomial of degree $n$, we denote by $\mathcal{R} = \mathbb{Z}[x]/\langle f(x)\rangle$ and with $\mathcal{K} = \mathbb{Q}[x]/\langle f(x)\rangle$. In particular, we take $f(x) = x^n + 1$ with $n$ a power of 2.

For a positive integer $p$, we denote by $\mathbb{Z}_p$ the set of integers $\{-\lfloor \frac{p-1}{2} \rfloor, \ldots, \lceil \frac{p-1}{2} \rceil\}$ 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 to every coefficient. Let $x \in \mathbb{Q}$, $\lfloor x \rceil$ be the rounding to the nearest integer. The same holds coefficient-wise for polynomials in $\mathcal{K}$.

We denote by the integer $t > 1$ 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 and space by $q$ and $\mathcal{R}_q$, respectively. Note that $q$ is the product of $k$ pair-wise co-prime integers $r_i > 1$ of approximately the same size. Moreover, we set $r_i$ to be co-prime with $t$ and satisfy $r_i \equiv 1 \mod 2n$.

Finally, if we want to apply the modulo switch in a circuit as BGV (see Section 5.2), we need $q_{\mathsf{ms}} = \prod_{j=1}^{L} p_j$ with the $p_j$ defined analogously to $q$. For any $\ell$, we denote by $q_\ell = \prod_{j=1}^{\ell} p_j$. The multiplicative depth $M$ of the circuit determines the number of primes $L = M + 1$.

Let $\chi_s$ and $\chi_u$ be secret key distributions and $\chi_e$ an error distribution from the Learning with Errors over Rings (RLWE) problem. Tipically, we have $\chi_s = \chi_u = \mathcal{U}_3$, uniform distribution on $\mathbb{Z}_3$, and $\chi_e = \mathcal{DG}(0, \sigma^2)$, discrete Gaussian centered in 0 with standard deviation $\sigma = 3.2$ [1]. Note that in this article, we assume that the distributions are symmetric. In general, if $\chi$ is a probabilistic distribution and $a \in \mathcal{R}$ is a random polynomial, we write $a \leftarrow \chi$ when sampling each coefficient independently from $\chi$.

### 2.2 Mathematical Background

*Coverage probability for Gaussian-distributed variables.* Let $X$ be a random variable from a Gaussian distribution centered in 0 of variance $V$, then

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

*Coverage probability for vectors.* Suppose we want to study the infinity norm of a vector $\mathbf{X}$ of random variables distributed as $X$. 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:

$$\mathbb{P}\Big(||\mathbf{X}||_\infty > x\Big) \leq n\mathbb{P}\Big(|X| > x\Big). \tag{2}$$

In particular, if $X$ follows a Gaussian distribution as above, we have

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

*Canonical embedding and norm.* We recall the results of [12, 14, 21]. 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 [12].

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 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 [16], using the Lyapunov Central Limit Theorem, it is possible to prove that for any root of unity $\zeta = \cos(\alpha) + i\sin(\alpha)$, the random variable $a(\zeta)$ is a complex random variable which can be approximated by a complex Gaussian random variable. 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 $(X, Y)^t = C(Z, Z')^t$ with $Z$ and $Z'$ i.i.d. standard Gaussian random variables. Therefore,

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

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 random variable, $||(Z,Z')||_2^2$ is Chi-squared distributed and

$$\mathbb{P}\left(||(Z,Z')||_2^2 < \frac{x^2}{M}\right) = 1 - e^{-\frac{x^2}{2M}} \geq 1 - e^{-\frac{x^2}{nV_a}} \Rightarrow \mathbb{P}\left(|a(\zeta_m)| > x\right) \leq e^{-\frac{x^2}{nV_a}}.$$

Therefore,

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

*Probability operators.* Let $X, Y, Z$ be real random variables and $c$ a constant. The expected value enjoys the following properties:

- it is linear: $\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$ and $\mathbb{E}[cX] = c\mathbb{E}[X]$;
- if $X$ is sampled from a symmetric distribution, i.e. $\mathbb{P}(X = x) = \mathbb{P}(X = -x)$ for any $x \in \mathbb{R}$, then $\mathbb{E}[X] = 0$;
- if $X$ and $Y$ are independent, then $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$;
- in general, $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] + \mathsf{Cov}(X,Y)$.

The covariance is consequently defined as $\mathsf{Cov}(X,Y) = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$ and is such that

- if $X$ and $Y$ are independent, then $\mathsf{Cov}(X,Y) = 0$;
- it is bilinear.

Some characteristics of the variance are

- $\mathsf{Var}(X) \geq 0$;
- $\mathsf{Var}(X+Y) = \mathsf{Var}(X) + \mathsf{Var}(Y) + 2\mathsf{Cov}(X,Y)$ and, more in general, $V(\sum_i X_i) = \sum_i V(X_i) + \sum_{i_1 \neq i_2} \mathsf{Cov}(X_{i_1}, X_{i_2})$;
- if $X$ and $Y$ are independent, then $\mathsf{Var}(X+Y) = \mathsf{Var}(X) + \mathsf{Var}(Y)$;
- $\mathsf{Var}(cX) = c^2\mathsf{Var}(X)$;
- if $X$ and $Y$ are independent and $\mathbb{E}[X] = \mathbb{E}[Y] = 0$, then $\mathsf{Var}(XY) = \mathsf{Var}(X)\mathsf{Var}(Y)$.

We list the variances of the variable we will use in the rest of the work:

$$\begin{array}{lll} \text{If } X \leftarrow \mathcal{DG}(0,\sigma^2) & \text{then} & \mathsf{Var}(X) = \sigma^2 \\ \text{If } X \leftarrow \mathcal{U}_q & \text{then} & \mathsf{Var}(X) = \frac{q^2-1}{12} \approx \frac{q^2}{12} \\ \text{If } X \leftarrow \mathcal{U}_t & \text{then} & \mathsf{Var}(X) = \frac{t^2-1}{12} \\ \text{If } X \leftarrow \mathcal{U}_3 & \text{then} & \mathsf{Var}(X) = \frac{3^2-1}{12} = \frac{2}{3} \end{array} \tag{5}$$

## 3 The FV Scheme

The FV scheme [17] is a cutting-edge FHE scheme whose security relies on the hardness of the ring learning with errors (RLWE) problem. This section presents

the scheme, considering the latest enhancements proposed in [22]. 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($\lambda, L$)**

Define parameters and distributions accordingly to $\lambda$ 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, $q_\ell$ denotes the ciphertext modulus and $\nu$ the *invariant noise*. The invariant noise [21] is the minimal $\nu$ 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 $\nu$ is small enough. In particular, it is correct when the coefficients of $\nu$ belong to the interval $(-\frac{1}{2}, \frac{1}{2}]$. After the encryption operation, the invariant noise is

$$\nu_{\mathsf{clean}} = \frac{t}{q} (\varepsilon + eu + e_0 + e_1 s) \tag{6}$$

where $\varepsilon = \lfloor \frac{q}{t} m \rceil - \frac{q}{t} m = -\frac{[qm]_t}{t}$, [22].

*Proof.*

$$\frac{t}{q} [c_0 + c_1 s]_q = \frac{t}{q} \left[ \left\lfloor \frac{q}{t} m \right\rceil + ub + e_0 + (ua + e_1) s \right]_q =$$
$$= \frac{t}{q} \left( \frac{q}{t} m + \varepsilon + ue + e_0 + e_1 s \right) + kt = m + \nu_{\mathsf{clean}} + kt.$$

*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 $\mathbf{c}_{\mathsf{const}} = ([\alpha c_0]_{q_\ell}, [\alpha c_1]_{q_\ell})$.

Let $u, k \in \mathcal{R}$. The invariant noise is

$$\begin{aligned} \frac{t}{q_\ell}[c_0 + c_1 s + c_0' + c_1' s]_{q_\ell} &= \frac{t}{q_\ell}([c_0 + c_1 s]_{q_\ell} + [c_0' + c_1' s]_{q_\ell} - u q_\ell) \\ &= [m + m']_t + \nu + \nu' + kt \implies \nu_{\mathsf{add}} = \nu + \nu \end{aligned} \tag{7}$$

$$\begin{aligned} \frac{t}{q_\ell}[\alpha c_0 + \alpha c_1 s)]_{q_\ell} &= \frac{t}{q_\ell}\left(\alpha[c_0 + c_1 s]_{q_\ell} - u q_\ell\right) = [\alpha m]_t + \alpha\nu + kt \\ &\implies \nu_{\mathsf{const}} = \alpha\nu, \end{aligned} \tag{8}$$

*Multiplication & Modulus switching.* In this section, we are going to see the multiplication algorithm presented in [22], which applies the modulus switching to one of the ciphertexts before multiplying them, in order to make the RNS representation more efficient. The modulus switch technique was first introduced for the BGV scheme in [8] to reduce the error associated with a ciphertext. In the FV 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 $q_\ell'$. 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

$$\nu_{\mathsf{ms}}(q_\ell') = \frac{t}{q_\ell'}(\varepsilon_0 + \varepsilon_1 s), \text{ with } \varepsilon_i = -\frac{[q_\ell' c_i]_{q_\ell}}{q_\ell}. \tag{9}$$

Indeed, since $\frac{t}{q_\ell'}[c_0' + c_1' s]_{q_\ell'} = \frac{t}{q_\ell'}\left[\lfloor \frac{q_\ell'}{q_\ell} c_0 \rceil + \lfloor \frac{q_\ell'}{q_\ell} c_1 \rceil s\right]_{q_\ell'}$, we have

$$\begin{aligned} \frac{t}{q_\ell'}[c_0' + c_1' s]_{q_\ell'} &= \frac{t}{q_\ell'}\left[\frac{q_\ell'}{q_\ell} c_0 + \varepsilon_0 + \frac{q_\ell'}{q_\ell} c_1 s + \varepsilon_1 s\right]_{q_\ell'} = \\ &= \frac{t}{q_\ell}[c_0 + c_1 s]_{q_\ell} + \frac{t}{q_\ell'}(\varepsilon_0 + \varepsilon_1 s) + ht = m + \nu + \frac{t}{q_\ell'}(\varepsilon_0 + \varepsilon_1 s) + k't. \end{aligned}$$

In the case of multiplication, the 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 $q_\ell'$. The new modulus $q_\ell'$ is required to be of approximately the same size of $q_\ell$, 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 $\frac{t}{q'_\ell}(c'_0 + c'_1 s) = m' + \nu' + h't$. Thus,

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

where the noise after the multiplication is

$$\nu_{\mathsf{mul}}(q_\ell) = -\nu\nu' + \nu \frac{t}{q'_\ell} (c'_0 + c'_1 s) + \nu' \frac{t}{q_\ell} (c_0 + c_1 s) + \frac{t}{q_\ell} (\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2). \quad (10)$$

Note that the multiplication output needs to be transformed back to a ciphertext in $\mathcal{R}_q^2$ (re-linearization); this is done by encrypting its last term $d_2$ via key switching (see Section 3.1).

### 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. For a multiplication, we convert the ciphertext term $d_2 \cdot s^2$ to a polynomial $c_0^{\mathsf{ks}} + c_1^{\mathsf{ks}} \cdot s$ and for a rotation, we convert the ciphertext term $c_1 \cdot \mathrm{rot}(s)$ to a polynomial $c_0^{\mathsf{ks}} + c_1^{\mathsf{ks}} \cdot s$. In the following, we will only analyze multiplication, and more specifically, we will output $\mathbf{c}' = (d_0 + c_0^{\mathsf{ks}}, d_1 + c_1^{\mathsf{ks}})$ and denote the ciphertext term we want to remove by $d_2$. This concept encompasses rotations, as we can identify the term we wish to eliminate as $d_1$, resulting in an output of $(d_0 + c_0^{\mathsf{ks}}, c_1^{\mathsf{ks}})$.

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 variants: Brakerski

Vaikuntanathan (BV), Gentry Halevi Smart (GHS), and Hybrid. In particular, for the BV re-linearization, we consider the latest improvements proposed in [4, 20] to make this operation more compatible with the RNS representation.

*Brakerski-Vaikuntanathan* The strategy is to decompose $d_2$ exploiting the Chinese Remainder Theorem (CRT). Let $q_\ell = r_1 \cdots r_{k_\ell}$ with $r_i$ pairwise co-prime of approximately the same size ($r_i \approx \sqrt[k_\ell]{q_\ell}$).

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

Sample $a_i \leftarrow \mathcal{U}_{q_\ell}$, $e_i \leftarrow \chi_e$ and set $(b_i, a_i) = \left( \left[ \left[ \left( \frac{q_\ell}{r_i} \right)^{-1} \right]_{r_i} \frac{q_\ell}{r_i} s^2 - a_i s + e_i \right]_{q_\ell}, a_i \right)$ for $i = 1, \ldots, k_\ell$. 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 $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu + \nu_{\mathsf{ks}}^{\mathsf{BV}})$ where $\mathbf{c} = \left( \left[ d_0 + \sum_{i=1}^{k_\ell} [d_2]_{r_i} b_i \right]_{q_\ell}, \left[ d_1 + \sum_{i=1}^{k_\ell} [d_2]_{r_i} a_i \right]_{q_\ell} \right)$.

Observing that $\left[ \sum_{i=1}^{k_\ell} [d_2]_{r_i} (b_i + a_i s) \right]_{q_\ell}$ is equal to

$$\left[ \sum_{i=1}^{k_\ell} [d_2]_{r_i} \left( \left[ \left( \frac{q_\ell}{r_i} \right)^{-1} \right]_{r_i} \frac{q_\ell}{r_i} s^2 + e_i \right) \right]_{q_\ell} = \left[ d_2 s^2 + \sum_{i=1}^{k} [d_2]_{r_i} e_i \right]_{q_\ell},$$

we have

$$\begin{aligned} \frac{t}{q_\ell} [c_0 + c_1 s]_{q_\ell} &= \frac{t}{q_\ell} \left[ d_0 + d_1 s + d_2 s^2 + \sum_{i=1}^{k_\ell} [d_2]_{r_i} e_i \right]_{q_\ell} \\ &= m + \nu + \frac{t}{q_\ell} \sum_{i=1}^{k_\ell} [d_2]_{r_i} e_i + kt. \end{aligned}$$

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

$$\nu_{\mathsf{ks}}^{\mathsf{BV}}(q_\ell) = \frac{t}{q_\ell} \sum_{i=1}^{k_\ell} [d_2]_{r_i} e_i. \tag{11}$$

*Gentry-Halevi-Smart* An alternative is encrypting $q'_\ell s^2$ instead of $s^2$ with $q'_\ell$ a big number, usually of approximately the same size of $q_\ell$. In this way, the error quantity added is divided by $q'_\ell$.

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

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

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

**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 $\mathfrak{c} = (\mathbf{c}, q_\ell, \nu + \nu_{\mathsf{ks}}^{\mathsf{GHS}})$ with $\mathbf{c} = \left(\left[d_0 + \left\lfloor \frac{d_2 b'}{q'_\ell} \right\rceil\right]_{q_\ell}, \left[d_1 + \left\lfloor \frac{d_2 a'}{q'_\ell} \right\rceil\right]_{q_\ell}\right)$.

To compute the invariant noise, we have to perform the following operation

$$\begin{aligned}
\frac{t}{q_\ell}[c_0 + c_1 s]_{q_\ell} &= \frac{t}{q_\ell}\left[d_0 + d_1 s + \left\lfloor \frac{d_2 b'}{q'_\ell} \right\rceil + \left\lfloor \frac{d_2 a'}{q'_\ell} \right\rceil s\right]_{q_\ell} \\
&= \frac{t}{q_\ell}\left[d_0 + d_1 s + \frac{d_2(q'_\ell s^2 + e')}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right]_{q_\ell} \\
&= m + \nu + \frac{t}{q_\ell}\left(\frac{d_2 e'}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right) + kt.
\end{aligned}$$

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

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

*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 higher 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 $q_\ell q'_\ell$ instead of $q_\ell$. This larger factor means that to achieve the same level of security, the modulus $q_\ell$ 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 $k$, and the division is done considering $q'_\ell \approx \sqrt[k]{q_\ell}$. For further information see [19, 22].

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

Sample $a_i \leftarrow \mathcal{U}_{q_\ell q'_\ell}$, $e_i \leftarrow \chi_e$ and output $\mathsf{ks}^{\mathsf{Hybrid}} = \{(b_i, a_i)\}_{i=1,\dots,k}$ with

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

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

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

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

Since $[b_i + a_i s]_{q_\ell q'_\ell} = \left[q'_\ell \left[\left(\frac{q_\ell}{r_i}\right)^{-1}\right]_{r_i} \frac{q_\ell}{r_i} s^2 + e_i\right]_{q_\ell q'_\ell}$, we have

$$\begin{aligned}
\frac{t}{q_\ell}[c_0 + c_1 s]_{q_\ell} &= \frac{t}{q_\ell}\left[d_0 + d_1 s + \frac{\sum_{i=1}^{k}[d_2]_{r_i}(b_i + a_i s)}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right]_{q_\ell} \\
&= \frac{t}{q_\ell}\left[d_0 + d_1 s + d_2 s^2 + \frac{\sum_{i=1}^{k}[d_2]_{r_i} e_i}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right]_{q_\ell} \\
&= m + \nu + \frac{t}{q_\ell}\left(\frac{\sum_{i=1}^{k}[d_2]_{r_i} e_i}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right) + kt.
\end{aligned}$$

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

$$\nu_{\mathsf{ks}}^{\mathsf{Hybrid}}(q_\ell) = \frac{t}{q_\ell}\left(\frac{\sum_{i=1}^{k}[d_2]_{r_i} e_i}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right). \tag{13}$$

## 4 Analyzing the Error with the average-case approach

The purpose of this section is to investigate how errors behave during homomorphic operations, with the goal of ensuring correct decryption. Specifically, we aim to establish that the error coefficients lie in the interval $(-\frac{1}{2}, \frac{1}{2}]$ with overwhelming probability.

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

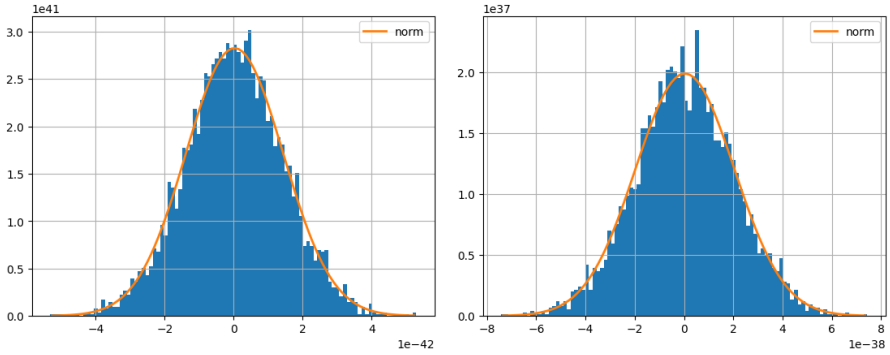
$$\mathbb{P}\Big(||\nu||_\infty > \frac{1}{2}\Big) \leq n\Big(1 - \mathrm{erf}\big(\frac{1}{2\sqrt{2V}}\big)\Big) \leq n(1 - \mathrm{erf}(D)),$$

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

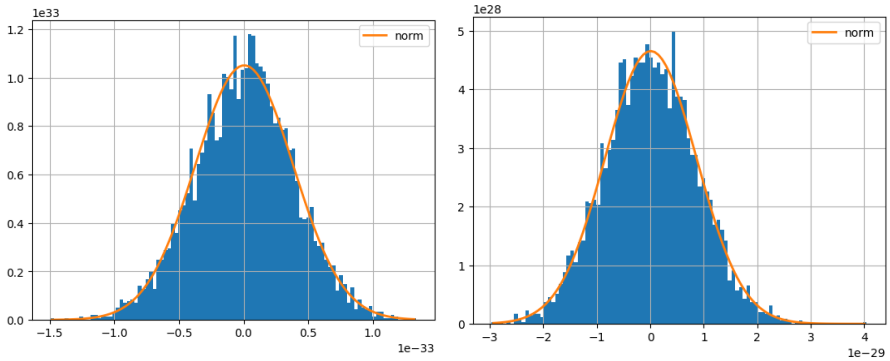
*Distribution.* We have studied the distribution of the coefficients of the error vector computationally[5] with Python **fitter** package[6], obtaining that they can be well-approximated by i.d. Gaussians. We show the results for the first coefficient applied to a Base Model circuit of multiplicative depth 0, 1, 2, and 3 (see Figure 2).

[5] The code used in this paper is written by us, we will update these and all the following results using an open library

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




SSE: 1.1044e+82, $\text{ks}_{\text{stat}}$: 0.00628, $\text{ks}_{\text{pval}}$: 0.988437 SSE: 8.4321e+73, $\text{ks}_{\text{stat}}$: 0.008881, $\text{ks}_{\text{pval}}$: 0.821915




SSE: 2.6367e+65, $\text{ks}_{\text{stat}}$: 0.012477, $\text{ks}_{\text{pval}}$: 0.41443 SSE: 3.7681e+56, $\text{ks}_{\text{stat}}$: 0.007756, $\text{ks}_{\text{pval}}$: 0.922085

Table 1: The analysis pictured considers 5000 samples computed with the GHS re-linearization and parameters $t = 3$, $n = 2^{13}$, $q = 2^{149} + 1$, $\chi_s = \chi_u = \mathcal{U}_3$, $\chi_e = \mathcal{DG}(0, \sigma^2)$ with $\sigma = 3.2$ and $\eta = 2$. The level of security guaranteed is $\lambda = 128$.

*Characterization of the error.* In the next paragraphs, we prove that the error coefficients have always mean 0, and we show how to compute the variance as the different operations are performed. To do so, we give a general characterization of the error as

$$\nu = \sum_{\iota} a_{\iota} s^{\iota}, \tag{14}$$

where the following conditions hold:

1. $\mathbb{E}[a_{\iota}|_i] = 0$ for any $\iota$,
2. $\mathsf{Cov}(a_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) = 0$ if either $\iota_1 \neq \iota_2$ or $i_1 \neq i_2$.

See Appendix A for the proof.

*Expected value.* As a consequence of Condition 1, we obtain that the error coefficients have mean 0 all along the circuit, i.e.

$$\mathbb{E}[\nu|_i] = 0. \tag{15}$$

*Proof.* Since the error $\nu$ can be written as in (14), we have

$$\nu|_i = \sum_\iota (a_\iota s^\iota)|_i = \sum_\iota \sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|_{i-j \bmod n}$$

where $\xi(i,j) = \begin{cases} 1 & \text{if } i-j \in [0,n) \\ -1 & \text{otherwise} \end{cases}$.
Hence, by the linearity of the expected value (see Section 2.2),

$$\mathbb{E}[\nu|_i] = \sum_\iota \sum_{j=0}^{n-1} \xi(i,j) \mathbb{E}[a_\iota|_j] s^\iota|_{i-j \bmod n} = 0.$$

Note that the secret key $s$ is seen as a fixed vector.

*Variance.* From Condition 2, we have

$$\mathsf{Var}(\nu|_i) = \sum_\iota \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j) s^\iota|^2_{i-j \bmod n}. \tag{16}$$

*Proof.* Analogously to the previous proof, we have

$$\begin{aligned} \mathsf{Var}(\nu|_i) &= \mathsf{Var}\Big(\sum_\iota \sum_{j=0}^{n-1} \xi(i,j) a_\iota|_j s^\iota|^2_{i-j \bmod n}\Big) \\ &\overset{\text{Sec. 2.2}}{=} \sum_\iota \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j) s^\iota|^2_{i-j \bmod n} + \\ &\quad + \sum_{\iota_1 \neq \iota_2 \text{ or } j_1 \neq j_2} \xi(i,j_1)\xi(i,j_2) \mathsf{Cov}(a_{\iota_1}|_{j_1}, a_{\iota_2}|_{j_2}) s^{\iota_1}|^2_{i-j_1 \bmod n} s^{\iota_2}|^2_{i-j_2 \bmod n} \end{aligned}$$

Thus, by Equation (14), $\mathsf{Var}(\nu|_i) = \sum_\iota \sum_{j=0}^{n-1} \mathsf{Var}(a_\iota|_j) s^\iota|^2_{i-j \bmod n}$.

While the variance of $\nu|_i$ could be computed by calculating $\mathsf{Var}(a_\iota|_i)$ and $s^\iota|_i$ for any $\iota$, this approach can be challenging to generalize due to the potential complexity of $a_\iota$. Instead, we examined how the variance changes with each operation applied. Although this method is slightly less precise, it still provides good bounds and is easier to manage.

*Encryption.* The variance of the error coefficients of a fresh ciphertext is

$$\mathsf{Var}(\nu_{\mathsf{clean}}|_i) \approx \frac{B_{\mathsf{clean}}}{q^2} \quad \text{where} \quad B_{\mathsf{clean}} = t^2\Big(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s\Big). \tag{17}$$

*Proof.* By Equation (6), the fresh error $\nu_{\mathsf{clean}}$ can be written as $\nu_{\mathsf{clean}} = a_0 + a_1 s$ with $a_0 = \frac{t}{q}(\varepsilon + eu + e_0)$, $a_1 = \frac{t}{q}e_1$. Defined $V_e, V_s, V_u$ the variances of coefficients from the distributions $\chi_e, \chi_s, \chi_u$, respectively, $\mathsf{Var}(a_0|_i) = \frac{t^2}{q^2}\left(\frac{1}{12} + nV_eV_u + V_e\right)$ and $\mathsf{Var}(a_1|_i) = \frac{t^2}{q^2}V_e$, while $\mathbb{E}[\sum_i s|_i^2] = nV_s$. It follows that

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{clean}}|_i) \overset{16}{=} \mathsf{Var}(a_0|_i) + \sum_{j=0}^{n-1} \mathsf{Var}(a_1|_j)s|^2_{i-j \bmod n} \approx \\ \approx \frac{t^2}{q^2}\Big(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s\Big),\end{aligned} \tag{18}$$

where $\mathsf{Var}(\varepsilon|_i) = \frac{1}{12}$ comes from the fact that $\varepsilon = \left\lfloor \frac{q}{t}m \right\rceil - \frac{q}{t}m = -\frac{[qm]_t}{t}$ and $[qm]_t$ can be consider a random element from the uniform distribution $\mathcal{U}_t$.

*Addition.* Let $\nu$, $\nu'$ the errors of two ciphertexts computed independently, so independent themselves. Then,

$$\mathsf{Var}((\nu + \nu')|_i) \overset{\text{Sec. } 2.2}{=} \mathsf{Var}(\nu|_i) + \mathsf{Var}(\nu'|_i). \tag{19}$$

The same argument can be applied to modulo-switch and key-switch operations.

*Modulo switching.* Since the modulus switch from $q_\ell$ to $q'_\ell$ adds an error $\nu_{\mathsf{ms}}(q'_\ell) = \frac{t}{q'_\ell}(\varepsilon_0 + \varepsilon_1 s)$ independent on $\nu$ by Equation (9), the variance becomes

$$\mathsf{Var}((\nu + \nu_{\mathsf{ms}}(q'_\ell))|_i) = \mathsf{Var}(\nu|_i) + \frac{B_{\mathsf{ms}}}{q'^2_\ell} \quad \text{where} \quad B_{\mathsf{ms}} = \frac{t^2}{12}(1 + nV_s). \tag{20}$$

*Key switching.* Analogously, after the key switch, the variance becomes

$$\mathsf{Var}(\nu|_i) + V_{\mathsf{ks}}(q_\ell), \tag{21}$$

where $V_{\mathsf{ks}}(q_\ell)$ depends on the chosen key-switching variants. Specifically,

- *BV key switching.* Since $\nu^{\mathsf{BV}}_{\mathsf{ks}}(q_\ell) = \frac{t}{q_\ell}\sum_{i=1}^{k_\ell}[d_2]_{r_i}e_i$ as in Equation (11) with $r_i \approx \sqrt[k_\ell]{q_\ell}$ and $[d_2]_{r_i}$ behaves as if selected uniformly at random from $\mathcal{U}_{r_i}$, we have that

$$V^{\mathsf{BV}}_{\mathsf{ks}}(q_\ell) = \frac{t^2}{q_\ell^2}nV_e\sum_{i=1}^{k_\ell}\frac{r_i^2}{12} \approx k_\ell\frac{t^2n}{12q_\ell^2}\sqrt[k_\ell]{q_\ell^2}V_e. \tag{22}$$

- *GHS key switching.* From Equation (12) we have $\nu^{\mathsf{GHS}}_{\mathsf{ks}}(q_\ell) = \frac{t}{q_\ell}\left(\frac{d_2e'}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right)$ where $q'_\ell \approx q_\ell$, hence

$$V^{\mathsf{GHS}}_{\mathsf{ks}}(q_\ell) \approx \frac{t^2}{12q_\ell^2}\left(nV_e + 1 + nV_s\right) \tag{23}$$

and, in particular, we can write $V_{\mathsf{ks}}^{\mathsf{GHS}}(q_\ell) = B_{\mathsf{ks}}^{\mathsf{GHS}}/q_\ell^2$, where

$$B_{\mathsf{ks}}^{\mathsf{GHS}} \approx \frac{t^2}{12}\left(nV_e + 1 + nV_s\right). \tag{24}$$

– *Hybrid key switching.* Since $\nu_{\mathsf{ks}}^{\mathsf{Hybrid}}(q_\ell) = \frac{t}{q_\ell}\left(\frac{\sum_{i=1}^{k}[d_2]_{r_i}e_i}{q'_\ell} + \varepsilon_0 + \varepsilon_1 s\right)$ as in Equation (13) and $r_i \approx q'_\ell \approx \sqrt[k]{q_\ell}$, we have

$$V_{\mathsf{ks}}^{\mathsf{Hybrid}}(q_\ell) \approx \frac{t^2}{q_\ell^2}\left(\frac{knVe}{12} + \frac{1}{12} + \frac{nV_s}{12}\right)$$

then we can set $V_{\mathsf{ks}}^{\mathsf{Hybrid}}(q_\ell) = B_{\mathsf{ks}}^{\mathsf{Hybrid}}/q_\ell^2$, where

$$B_{\mathsf{ks}}^{\mathsf{Hybrid}} \approx \frac{t^2}{12}\left(knVe + 1 + nV_s\right).$$

Note that, in this work, we focus on the GHS and Hybrid variants.

*Constant multiplication.* The variance of the error coefficients after a constant multiplication is

$$\mathsf{Var}((\alpha\nu)|_i) \approx \frac{(t^2-1)n}{12}\mathsf{Var}(\nu|_i), \tag{25}$$

*Proof.* Since the coefficients of $\alpha$ behave as sampled independently at random from a uniform distribution over $\mathcal{U}_t$, then $\mathbb{E}[\alpha|_i] = 0$ and $\mathsf{Var}(\alpha|_i) \approx \frac{t^2-1}{12}$. It follows

$$\begin{aligned}\mathsf{Var}((\alpha\nu)|_i) &\overset{(14)}{=} \sum_j \mathsf{Var}(\alpha|_j \nu|_{i-j \bmod n}) \overset{\text{indep.}}{=} \\ &= \sum_j \mathsf{Var}(\alpha|_j)\mathsf{Var}(\nu|_{i-j \bmod n}) = \frac{(t^2-1)n}{12}\mathsf{Var}(\nu|_{i-j \bmod n}).\end{aligned}$$

*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 two independently-computed ciphertexts, then by Equation (10),

$$\begin{aligned}\nu_{\mathsf{mul}}(q_\ell) &= -\nu\nu' + \nu\frac{t}{q'_\ell}(c'_0 + c'_1 s) + \nu'\frac{t}{q_\ell}(c_0 + c_1 s) + \frac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2) = \\ &= -\sum_{\iota_1}\sum_{\iota_2} a_{\iota_1} a'_{\iota_2} s^{\iota_1+\iota_2} + \sum_{\iota_1} a_{\iota_1}\Big(\frac{t}{q'_\ell}c'_0 s^{\iota_1} + \frac{t}{q'_\ell}c'_1 s^{\iota_1+1}\Big) + \\ &\quad + \sum_{\iota_2} a_{\iota_2}\Big(\frac{t}{q_\ell}c_0 s^{\iota_2} + \frac{t}{q_\ell}c_1 s^{\iota_2+1}\Big) + \frac{t}{q_\ell}(\varepsilon_0 + \varepsilon_1 s + \varepsilon_2 s^2)\end{aligned}$$

and the variance of its coefficients is

$$\begin{aligned}\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i) = n\sum_{\iota_1}\sum_{\iota_2}\mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)\sum_{j=0}^{n-1} s^{\iota_1+\iota_2}|^2_{i-j \bmod n} + \\ + n\sum_{\iota_1}\mathsf{Var}(a_{\iota_1}|_i)\frac{t^2}{12}\sum_{j=0}^{n-1}\big(s^{\iota_1}|^2_{i-j \bmod n} + s^{\iota_1+1}|^2_{i-j \bmod n}\big) + \\ + n\sum_{\iota_2}\mathsf{Var}(a'_{\iota_2}|_i)\frac{t^2}{12}\sum_{j=0}^{n-1}\big(s^{\iota_2}|^2_{i-j \bmod n} + s^{\iota_2+1}|^2_{i-j \bmod n}\big) + \\ + \frac{t^2}{12q_\ell^2}\Big(1 + \sum_{j=0}^{n-1} s|^2_{i-j \bmod n} + \sum_{j=0}^{n-1} s^2|^2_{i-j \bmod n}\Big).\end{aligned}$$

Assuming the independence of the coefficients of a noise vector among each other, we could compute the quantity above as

$$\begin{aligned}n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i) + n\mathsf{Var}(\nu|_i)\frac{t^2}{12}(1+nV_s) + \\ + n\mathsf{Var}(\nu'|_i)\frac{t^2}{12}(1+nV_s) + \mathsf{Var}\Big(\frac{t}{q_\ell}(\varepsilon_0+\varepsilon_1 s+\varepsilon_2 s^2)|_i\Big).\end{aligned}$$

However, what we would obtain is

$$\begin{aligned}n\sum_{\iota_1}\sum_{\iota_2}\mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)\sum_{j_1=0}^{n-1} s^{\iota_1}|^2_{i-j_1 \bmod n}\sum_{j_2=0}^{n-1} s^{\iota_2}|^2_{i-j_2 \bmod n} + \\ + n\sum_{\iota_1}\mathsf{Var}(a_{\iota_1}|_i)\frac{t^2}{12}\sum_{j=0}^{n-1}\Big(s^{\iota_1}|^2_{i-j \bmod n} + s^{\iota_1}|^2_{i-j \bmod n}\sum_{j_1=0}^{n-1} s|^2_{i-j_1 \bmod n}\Big) + \\ + n\sum_{\iota_2}\mathsf{Var}(a'_{\iota_2}|_i)\frac{t^2}{12}\sum_{j=0}^{n-1}\Big(s^{\iota_2}|^2_{i-j \bmod n} + s^{\iota_2}|^2_{i-j \bmod n}\sum_{j_1=0}^{n-1} s|^2_{i-j_1 \bmod n}\Big) + \\ + \mathsf{Var}\Big(\frac{t}{q_\ell}(\varepsilon_0+\varepsilon_1 s+\varepsilon_2 s^2)|_i\Big),\end{aligned}$$

which differs from $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ in the powers of $s$.

To address this issue, we examined the expected value of the ratio between the target and obtained terms. Specifically, we computed the mean of

$$\frac{\sum_{i=0}^{n-1} s^{\iota}|_i^2}{\sum_{i_1=0}^{n-1} s|_{i_1}^2 \sum_{i_2=0}^{n-1} s^{\iota-1}|_{i_2}^2}, \tag{26}$$

for $\iota \geq 2$. Our analysis revealed that this mean can be approximated by

$$f(\iota) = -\frac{1}{e^{a\iota-b}} + c, \tag{27}$$

| $n$ | $a$ | $b$ | $c$ |
|---|---|---|---|
| $2^{12}$ | 0.2417 | 2.3399 | 8.1603 |
| $2^{13}$ | 0.2240 | 2.4181 | 8.8510 |
| $2^{14}$ | 0.2058 | 2.4844 | 9.5691 |
| $2^{15}$ | 0.1906 | 2.5489 | 10.2903 |

Table 2: Value for $a, b, c$ setting $\chi_s = \mathcal{U}_3$.

where $a, b, c$ are dependent on $n$ and are listed in Table 2.

As an illustrative example, we consider the case where $n = 2^{13}$. The values of $f(\iota)$ are listed in the table below, and Figure 1 pictures the graph of $f(\iota)$. It can be observed that $f$ approximates the mean of Equation (26) accurately.

| $\iota$ | 2 | 3 | 4 | 5 | ... | 27 | 28 | 29 | 30 |
|---|---|---|---|---|---|---|---|---|---|
| $f(\iota)$ | 1.9997 | 2.9996 | 3.9847 | 4.9405 | ... | 8.7651 | 8.7847 | 8.7968 | 8.7950, |

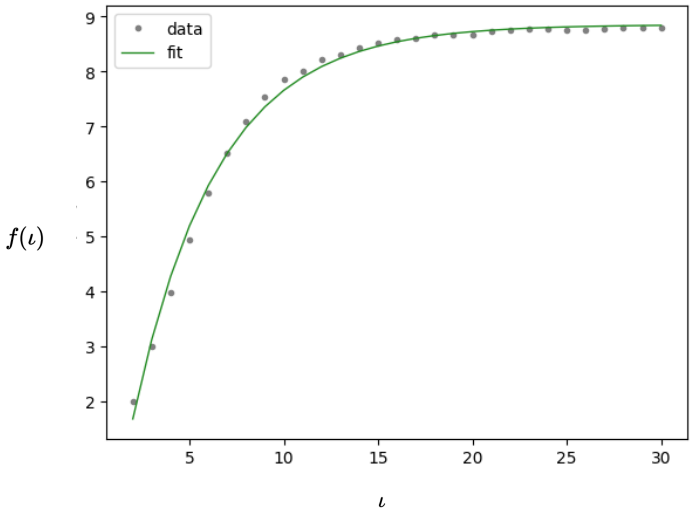



Fig. 1: The function $f(\iota)$ fitting the points from Equation (26) for $n = 2^{13}$.

It follows by induction from Equation (26) that

$$\sum_{i=0}^{n-1} s^{\iota}|_i^2 \approx (nV_s)^{\iota} g(\iota)$$

where $g(\iota) = f(2) \cdots f(\iota)$.

*Proof.* Recall that $\sum_{i=0}^{n-1} s|_i^2 \approx nV_s$, then

– $\sum_{i=0}^{n-1} s^2|_i^2 \overset{(26)}{\approx} \Big(\sum_{i=0}^{n-1} s|_i^2\Big) f(2) \approx (nV_s)^2 f(2) = (nV_s)^2 g(2)$,
– $\sum_{i=0}^{n-1} s^{\iota-1}|_i^2 \approx (nV_s)^{\iota-1} g(\iota-1) \implies$
$$\implies \sum_{i=0}^{n-1} s^{\iota}|_i^2 \overset{(26)}{\approx} nV_s(nV_s)^{\iota-1} g(\iota-1) f(\iota) = (nV_s)^{\iota} g(\iota).$$

It follows that, in order to convert the first term of $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$

$$n \sum_{\iota_1} \sum_{\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i) \sum_{j_1=0}^{n-1} s^{\iota_1}|^2_{i-j_1 \bmod n} \sum_{j_2=0}^{n-1} s^{\iota_2}|^2_{i-j_2 \bmod n} \approx$$
$$\approx n \sum_{\iota_1} \sum_{\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)(nV_s)^{\iota_1} g(\iota_1)(nV_s)^{\iota_2} g(\iota_2)$$

into

$$n \sum_{\iota_1} \sum_{\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i) \sum_{j=0}^{n-1} s^{\iota_1+\iota_2}|^2_{i-j \bmod n} \approx$$
$$\approx n \sum_{\iota_1} \sum_{\iota_2} \mathsf{Var}(a_{\iota_1}|_i)\mathsf{Var}(a'_{\iota_2}|_i)(nV_s)^{\iota_1+\iota_2} g(\iota_1+\iota_2),$$

we can multiply it by $\frac{g(\iota_1+\iota_2)}{g(\iota_1)g(\iota_2)} \leq \frac{g(T_1+T_2)}{g(T_1)g(T_2)}$ by Proposition 2. Analogously, we get

$$n \sum_{\iota_k} \mathsf{Var}(a_{\iota_k}|_i) \frac{t^2}{12} \sum_{j=0}^{n-1} s^{\iota_k+1}|^2_{i-j \bmod n} \approx n \sum_{\iota_k} \mathsf{Var}(a_{\iota_k}|_i) \frac{t^2}{12} (nV_s)^{\iota_k+1} g(\iota_k+1)$$

multiplying by $f(\iota_k+1) \leq f(T_k+1)$

$$n \sum_{\iota_k} \mathsf{Var}(a_{\iota_k}|_i) \frac{t^2}{12} \sum_{j=0}^{n-1} s^{\iota_1}|^2_{i-j \bmod n} \sum_{j_1=0}^{n-1} s|^2_{i-j_1 \bmod n} \approx n \sum_{\iota_k} \mathsf{Var}(a_{\iota_k}|_i) \frac{t^2}{12} (nV_s)^{\iota_k} g(\iota_k)(nV_s).$$

Therefore, we can bound $\mathsf{Var}(\nu_{\mathsf{mul}}(q_\ell)|_i)$ by

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

Now we prove that the first and last terms are negligible compared to the others. Hence we can set

$$\begin{aligned} \mathsf{Var}(\nu_{\mathsf{mul}}|_i) &\approx \tfrac{t^2 n}{12} \Big(\mathsf{Var}(\nu|_i)\Big(1 + nV_s f(T_1+1)\Big) + \mathsf{Var}(\nu'|_i)\Big(1 + nV_s f(T_2+1)\Big)\Big) \\ &\approx \tfrac{t^2 n^2 V_s}{12} \big(\mathsf{Var}(\nu|_i) f(T_1+1) + \mathsf{Var}(\nu'|_i) f(T_2+1)\big). \end{aligned} \tag{28}$$

*Proof.* For correct decryption, we require $\mathsf{Var}(\nu_{\mathsf{mul}}|_i) \leq \frac{1}{8D^2}$. Thus, since all the addends of Equation (28) are positive quantities, we have

$$\mathsf{Var}(\nu|_i) \frac{t^2 n^2 V_s}{12} f(T_1+1) \leq \mathsf{Var}(\nu|_i) \frac{t^2 n}{12} \big(1 + nV_s f(T_1+1)\big) \leq \frac{1}{8D^2},$$

That is,

$$\mathsf{Var}(\nu|_i) \leq \frac{3}{2D^2t^2n^2V_sf(T_1+1)}.$$

Thanks to Proposition 3,

$$n\mathsf{Var}(\nu|_i)\mathsf{Var}(\nu'|_i)\frac{g(T_1+T_2)}{g(T_1)g(T_2)} \leq n\frac{3}{2D^2t^2n^2V_sf(T_1+1)}\mathsf{Var}(\nu'|_i)\frac{g(T_1+T_2)}{g(T_1)g(T_2)} \leq$$
$$\leq \frac{18K_n}{D^2t^4n^3V_s^2}\frac{t^2n^2V_s}{12}f(T_2+1)\mathsf{Var}(\nu'|_i) \ll \frac{t^2n^2V_s}{12}f(T_2+1)\mathsf{Var}(\nu'|_i).$$

Note that all the homomorphic operations performed increase the variance of the error coefficients, hence either $\mathsf{Var}(\nu|_i) \geq B_{\mathsf{clean}}/q^2$ (thanks to Equation (17)) or $\mathsf{Var}(\nu|_i) \geq B_{\mathsf{clean}}/q^2 + B_{\mathsf{ms}}/q_\ell^2$ if we switched to a modulus $q_\ell \neq q$ (see Equation (20)). In the first case

$$\frac{t^2n}{12}\mathsf{Var}(\nu|_i)\big(1+nV_sf(T_1+1)\big) \geq \frac{t^2n}{12}\frac{B_{\mathsf{clean}}}{q^2}\big(1+nV_sf(T_1+1)\big) \overset{17}{\geq}$$
$$\geq \frac{t^2}{12q^2}t^2n^2V_eV_s\big(1+nV_sf(T_1+1)\big) \gg \frac{t^2}{12q^2}\big(1+nV_s+n^2V_s^2f(2)\big).$$

In the second one,

$$\frac{t^2n}{12}\mathsf{Var}(\nu|_i)\big(1+nV_sf(T_1+1)\big) \geq \frac{t^2n}{12}\frac{B_{\mathsf{ms}}}{q_\ell^2}\big(1+nV_sf(T_1+1)\big) \geq$$
$$\geq \frac{t^2}{12q_\ell^2}\frac{t^2n^2V_s}{12}\big(1+nV_sf(T_1+1)\big) \gg \frac{t^2}{12q_\ell^2}\big(1+nV_s+n^2V_s^2f(2)\big).$$

## 5 Modeling the Homomorphic Circuit

In this section, we exploit our theoretical work (Section 4) to improve the parameter generation for the FV scheme, providing closed formulas to compute the ciphertext modulus $q$ and, eventually, its sub-moduli $p_j$. These formulas are employed in our tool, which provides automated parameter selection for non-FHE experts (Section 6.2). In our analysis, we extend the previous work on BGV [24] considering the circuit models newly proposed by Mono *et al.* [24].

Each circuit performs a list of operations on $\eta$ ciphertexts $c_i$ in parallel, as illustrated in Figure 2. The resulting ciphertexts are homomorphically multiplied with another ciphertext 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 afterward before the homomorphic multiplication. It is mainly used to make the analysis easier, and it is equal to Model 1 and 2 with $\tau = 0$.

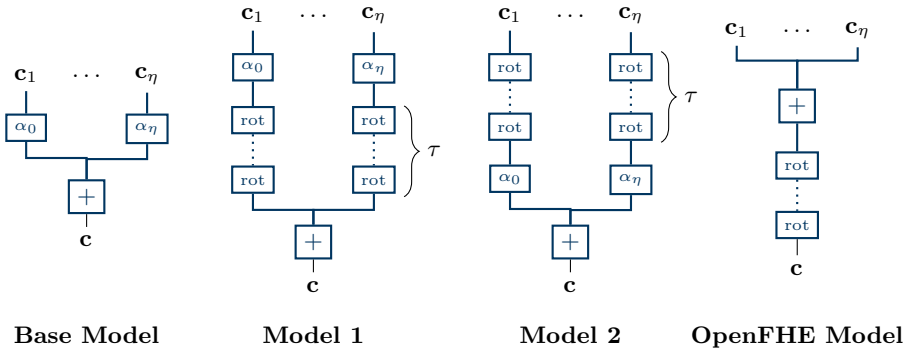



Fig. 2: Sequences of operations in the different models.

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

**OpenFHE Model** For comparison with previous work, we also define the model as used in the OpenFHE library [3, 22]. Here the first operation to be performed is homomorphic multiplication, then $\eta$ additions and $\tau$ rotations are carried out. The sequence is repeated $M$ times.

In the following, we consider the input ciphertexts in a circuit to encrypt different messages, therefore independent of each other. Moreover, we focus our analysis on Model 2, as it has the worst possible error growth. The same techniques can, however, be applied to simpler models as well, and we provide the results of our study in Table 3. In Section 5.2, we will study the case where a modulus switch to a smaller modulus is applied before every round of operations, as in BGV circuits. This divides the circuit into levels, and the number of moduli $L$ is determined by the multiplicative depth $M$, namely, $L = M + 1$.

### 5.1 Computing the error growth and the parameters setting

Consider a circuit with multiplicative depth $M = L - 1$. Let $V_\ell$ denote the variance of the error coefficients after the $\ell$-th level. In particular, $V_0 = \mathsf{Var}(\nu_{\mathsf{clean}}|i)$ is the variance just after the encryption, and $V_\ell$ is the variance after the $\ell$-th multiplication. Since the variance increases with each operation, we only need to ensure that the final error coefficients (with variance $V_{L-1}$) satisfy the condition in Section 4 for correct decryption throughout the circuit. That is, we require

$$V_{L-1} \leq \frac{1}{8D^2}.$$

We now examine the $\ell$-th level of Model 2, in order to compute $V_{L-1}$ recursively. Given the variance $V_{\ell-1}$ of each ciphertext in the circuit input, the evolution of the model can be described as follows:

– We first apply $\tau$ rotations, obtaining

$$V_{\ell-1} + \tau V_{\mathsf{ks}}.$$

Note that when the modulus is not explicitly specified in the formulas, it is assumed to be $q$.

– Secondly, we have a constant multiplication. Thus the variance is multiplied by $B_{\mathsf{const}} = \frac{(t^2-1)n}{12}$ (25), becoming

$$(V_{\ell-1} + \tau V_{\mathsf{ks}})B_{\mathsf{const}}.$$

If constant multiplications are not required, we set $B_{\mathsf{const}} = 1$.

– We add $\eta$ ciphertexts, getting

$$\eta(V_{\ell-1} + \tau V_{\mathsf{ks}})B_{\mathsf{const}}.$$

– During homomorphic multiplication, a modulo switch is applied from $q$ to $q' \approx q$ on one of the ciphertexts. This operation, adds to the variance a quantity $V_{\mathsf{ms}}(q') \approx V_{\mathsf{ms}}(q)$, leading to a total variance of

$$\eta(V_{\ell-1} + \tau V_{\mathsf{ks}})B_{\mathsf{const}} + V_{\mathsf{ms}}.$$

Finally, after performing multiplication (with re-linearization) of two ciphertexts, we have, thanks to Equations (21) and (28),

$$\begin{aligned} V_\ell &\approx \frac{t^2}{12}n^2 V_s \Big(2\eta(V_{\ell-1} + \tau V_{\mathsf{ks}})B_{\mathsf{const}} + V_{\mathsf{ms}}\Big) f(\ell+1) + V_{\mathsf{ks}} \\ &\approx \frac{t^2}{12}n^2 V_s \Big(2\eta(V_{\ell-1} + \tau V_{\mathsf{ks}})B_{\mathsf{const}} + V_{\mathsf{ms}}\Big) f(\ell+1). \end{aligned} \tag{29}$$

since $V_{\mathsf{ks}}$ is negligible.
Note that in the function $f$ (see Equation (27)), $T_1 = T_2 = \ell$, indeed the initial error has degree 1 in $s$ and this degree increases by 1 after each multiplication.

Since $V_\ell = B_\ell/q^2$ with $B_\ell$ is independent of $q$, we can rewrite Equation (29) as

$$V_\ell = \frac{B_\ell}{q^2} \approx \frac{(AB_{\ell-1} + C)f(\ell+1)}{q^2} \tag{30}$$

where $A = \frac{\eta t^2 n^2 V_s}{6} B_{\mathsf{const}}$ and $C = \frac{t^2 n^2 V_s}{12}(2\eta\tau B_{\mathsf{ks}} B_{\mathsf{const}} + B_{\mathsf{ms}})$. From Equation (30), we can recursively compute the final variance

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

and use it to determine a bound on the ciphertext modulus. Indeed, since $V_{L-1} \leq \frac{1}{8D^2}$, we have

$$q^2 \geq 8D^2 A^{L-2}(AB_{\mathsf{clean}} + C)g(L). \tag{31}$$

Note that the bound on the modulus $q$ is computed in the same way for all the models, except for the OpenFHE one, where the multiplication is done at the beginning of the circuit. In this case, we approximate $V_\ell = \frac{AB_{\ell-1}f(\ell+1)+C}{q^2}$, hence

$$q^2 \geq 8D^2 A^{L-2}(AB_{\mathsf{clean}} + C/f(2))g(L). \tag{32}$$

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: The table shows the constants $A$ and $C$ for each model depicted in Figure 2. These constants are required to calculate the ciphertext modulus using Equation (31) for all models, with the exception of the OpenFHE model where $q$ is computed using Equation (32).

### 5.2 Error growth and parameter setting with modulo switching

This section analyzes the homomorphic circuit (Model 2 in Figure 2) in which the modulus is switched to a smaller value as the first step in the circuit levels. It is important to note that, unlike BGV, the modulo-switching procedure does not decrease the error. However, it can still be useful because it enables computations to be performed in smaller moduli.

Using the same argument as in the previous section, we begin with noise after fresh encryption, with coefficient variance $V_0^{\mathsf{ms}} = \mathsf{Var}(\nu_{\mathsf{clean}}|_i)$, and need only ensure that the final variance is bounded. To analyze the $\ell$-th level, we follow a similar approach to that in Section 5.1, with the main difference being the use of different moduli. To compute $V_\ell^{\mathsf{ms}}$ from $V_{\ell-1}^{\mathsf{ms}}$, a modulo switch is performed from the current modulus $q_{L-\ell+1}$ to the smaller modulus $q_{L-\ell}$, obtaining

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

where $B_{\mathsf{ms}}$ as in Equation (20). Similarly to before (see Equation (29)), 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

$$V_\ell^{\mathsf{ms}} \approx \left( A_{\mathsf{ms}} V_{\ell-1} + \frac{C_{\mathsf{ms}}}{q_{L-\ell}^2} \right) f(\ell+1), \tag{33}$$

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 Table 3 considering Model 2.

Thanks to Equation (33), we can recursively compute the variance $V_{L-1}^{\mathsf{ms}}$ as

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

therefore,

$$\frac{A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L) + \sum_{i=1}^{L-1} \frac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \leq \frac{1}{8D^2}. \tag{34}$$

Observe that $V_{L-1}^{\mathsf{ms}} > V_{L-1}$, since $C_{\mathsf{ms}} > C$ and $q_\ell \leq q_L$. This implies that the ciphertext modulus obtained with the modulus switch technique is bigger than the one without this method. However, we can select specific moduli to achieve a ciphertext modulus close to the one obtained in Equation (31), improving efficiency.

**Fact 1.** *An optimal choice of the $p_j$'s that maximizes the efficiency of keeping the ciphertext modulus close to the one obtained without modulus-switching is the following:*

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

*then $q_{\mathsf{ms}}^2 \approx 8D^2 L A^{L-1} B_{\mathsf{clean}} g(L)$. It is worth noting that $q_{\mathsf{ms}}$ is approximately $\sqrt{L}$ times the ciphertext modulus obtained without modulus-switching.*

*Proof.* We begin our proof by contradiction, assuming that there exists at least one index $i$ in Equation (34) such that

$$\frac{A^{i-1} C_{\mathsf{ms}}}{q_i^2} \frac{g(L)}{g(L-i)} \gg \frac{A^{L-1} B_{\mathsf{clean}}}{q_L^2} g(L), \tag{35}$$

Then, we can estimate the variance of the $(L-1)$-th level to be

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

where $N \geq 1$ is the number of indices $i$ that satisfy Equation (35). It follows that $q_i^2 \geq 8D^2NA^{i-1}C_{\mathsf{ms}}g(L)/g(L-i)$, and, from Equation (35), we also have $q_L^2/q_i^2 \gg \frac{A^{L-i}B_{\mathsf{clean}}}{C_{\mathsf{ms}}}g(L-i)$. Therefore, we would obtain the bound

$$q_{\mathsf{ms}}^2 \gg 8D^2NA^{L-1}B_{\mathsf{clean}}g(L),$$

which is much larger than the bound for $q$ given by (31).

Thus, we can now suppose that for any $i$ we have[7]

$$\frac{A^{i-1}C_{\mathsf{ms}}}{q_i^2}\frac{g(L)}{g(L-i)} \leq \frac{A^{L-1}B_{\mathsf{clean}}}{q_L^2}g(L). \tag{36}$$

So we have that

$$V_{L-1}^{\mathsf{ms}} \leq \frac{LA^{L-1}B_{\mathsf{clean}}}{q_L^2}g(L) \implies q_{\mathsf{ms}}^2 \geq 8D^2LA^{L-1}B_{\mathsf{clean}}g(L), \tag{37}$$

namely, $q_{\mathsf{ms}}$ is $\sqrt{L}$ times the previous bound over $q$ (see Equation (31)). From Equation (36), for $1 \leq i \leq L-1$, we have

$$\begin{aligned} p_L^2 &\leq \frac{AB_{\mathsf{clean}}}{C_{\mathsf{ms}}} \\ p_{L-1}^2 p_L^2 &\leq \frac{A^2B_{\mathsf{clean}}}{C_{\mathsf{ms}}}g(2) \\ &\vdots \\ p_2^2 \cdots p_L^2 &\leq \frac{A^{L-1}B_{\mathsf{clean}}}{C_{\mathsf{ms}}}g(L-1) \end{aligned}$$

Moreover, from Equation (37), we have $p_1^2 \cdots p_L^2 \geq 8D^2LA^{L-1}B_{\mathsf{clean}}g(L)$. For optimal efficiency, we can choose $p_1$ to be as small as possible so we set $p_2^2 \cdots p_L^2$ to be the largest value satisfying $p_2^2 \cdots p_L^2 \approx \frac{A^{L-1}B_{\mathsf{clean}}}{C_{\mathsf{ms}}}g(L-1)$. This yields

$$p_1^2 \approx 8D^2LC_{\mathsf{ms}}f(L).$$

We can recursively apply the same argument to estimate the values of $p_2^2, \ldots, p_L^2$. Specifically, for any $2 \leq \ell \leq L$, we have

$$p_\ell^2 \approx Af(L-\ell+1), \quad \text{and} \quad p_L^2 \approx \frac{AB_{\mathsf{clean}}}{C_{\mathsf{ms}}}.$$

[7] Note that in this context, when we use the symbol $\leq$, we mean that the two members of the inequality can be approximately the same, with the left-hand side possibly being slightly larger.

# 6 Results

In this section, we show the effectiveness of our average-case approach by comparing it to the prior heuristic noise analyses of FV [12, 14, 21]. These works present a worst-case study employing the canonical norm (to facilitate the comparison, we recall their bounds in Section 6.1). In contrast, our approach, outlined in Sections 4 and 5, analyses the average behavior of error coefficients along a circuit. In Tables 5 and 6, we compare the error analysis for the single functions and for basic circuits using the bounds computed with the canonical norm and our approach. As can be seen, our method provides more accurate results, especially as the multiplicative depth of the circuit grows. Notably, with just three multiplications (see Table 6), we improve the bounds by at least 17 bits, and, more importantly, our bounds are very close to experimentally observed values (it differs at most of 3 bits). Furthermore, in Table 7, we show how this reflects in smaller bounds for the ciphertext modulus, which has a big impact on the efficiency and the security of the scheme. This suggests that our approach is a promising method for analyzing noise in the FV scheme, providing reliable estimates very close to actual error values.

Finally, in Section 6.2, using our theoretical formulas (see Section 4) and the security formula by Mono *et al.* [24], we provide an interactive parameter generator for the FV scheme. The generator outputs easy-to-use code snippets for the computed parameters for multiple state-of-the-art libraries.

## 6.1 Comparison with previous works

In order to facilitate the comparison of the results, we use the noise budget rather than the invariant noise itself. Roughly speaking, it measures in bits the distance between the input and the optimal bound for correct decryption, namely $\frac{1}{2}$.
Let $\nu$ be the invariant noise associated with a ciphertext $\mathbf{c}$, the *noise budget*, [26], of $||\nu||$ is

$$-\log_2(2 \cdot ||\nu||) = \log_2\left(\tfrac{1}{2}\right) - \log_2(||\nu||).$$

In the tables below, we compare our average-case approach with the current state-of-the-art method [14], as well as with experimental results. We conduct the computations with our MAGMA code[8] considering over 1000 trials.

In the following, we recall some important results about the canonical embedding norm and our method to explain how the bounds in the tables are obtained.

*Canonical norm bounds.* The bounds with the canonical norm are computed following the latest work by Costache *et al.* [14], and Iliashenko [21]. To ensure clarity, we summarize the relevant bound, taking into account the modifications

[8] Please find the link to our MAGMA code here: `https://github.com/Crypto-TII/FV_error_computation.git`. Note that the computations will soon be updated using the PALISADE library.

we made to the encryption and multiplication algorithms based on the work of Kim *et al.* [22].

Recall that in [12], the authors used the bound $||a||^{can} \leq D\sqrt{nV_a}$ for polynomials $a \in R$, assuming independence among the coefficients. With the same hypothesis, we can bound the canonical norm of the invariant noise $\nu$ with

$$||\nu||^{can} \leq D\sqrt{nV} \tag{38}$$

with probability greater or equal to

$$1 - ne^{\frac{-(D\sqrt{nV})^2}{nV}} = 1 - ne^{-D^2}$$

by Section 2. Aligning with previous works [12, 14, 21], we set $D = 6$, which guarantee that Equation (38) holds 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 $2^{-76}$.

Let us list the main results.

**Fresh ciphertext** By Equation (17), we have

$$||\nu_{\mathsf{clean}}||^{\text{can}} \leq D\frac{t}{q}\sqrt{n\Big(\frac{1}{12} + nV_eV_u + V_e + nV_eV_s\Big)} = D\frac{t}{q}\sqrt{n\left(\frac{1}{12} + \sigma^2\Big(\frac{4}{3}n + 1\Big)\right)}$$

**Addition & Modulo switch & Key switch** By the properties of norm, we have that

$$||\nu + \nu'||^{\text{can}} \leq ||\nu||^{\text{can}} + ||\nu'||^{\text{can}}$$

$$||\nu + \nu_{\mathsf{ms}}(q')||^{\text{can}} \leq ||\nu||^{\text{can}} + ||\nu_{\mathsf{ms}}(q')||^{\text{can}} \overset{20}{\leq} ||\nu||^{\text{can}} + \frac{D\sqrt{nB_{\mathsf{ms}}}}{q'}$$

$$||\nu + \nu_{\mathsf{ks}}||^{\text{can}} \leq ||\nu||^{\text{can}} + ||\nu_{\mathsf{ks}}||^{\text{can}} \overset{21}{\leq} ||\nu||^{\text{can}} + D\sqrt{nV_{\mathsf{ks}}}$$

**Constant multiplication**

$$||\alpha\nu||^{\text{can}} \leq ||\alpha||^{\text{can}}||\nu||^{\text{can}} \leq D\sqrt{n\frac{(t^2-1)}{12}}||\nu||^{\text{can}}$$

**Multiplication**

$$\begin{aligned}||\nu_{\text{mul}}||^{\text{can}} &\overset{10}{\leq} ||\nu||^{\text{can}}||\frac{t}{q'}(c_0' + c_1's)||^{\text{can}} + (||\nu||^{\text{can}} + \\ &\quad + ||\nu_{\mathsf{ms}}(q)||^{\text{can}})||\frac{t}{q}(c_0 + c_1s)||^{\text{can}} = \\ &\leq (2||\nu||^{\text{can}} + D\sqrt{nV_{\mathsf{ms}}(q)})Dt\sqrt{\frac{n}{12}(1 + nV_s)}\end{aligned}$$

Applying the same argument of Section 5.2 on the Base Model, we obtain

$$||\nu_\ell||^{\text{can}} \leq (2\eta||\nu_{\ell-1}||^{\text{can}} + \frac{D\sqrt{nB_{\mathsf{ms}}}}{q})Dt\sqrt{\frac{n}{12}(1+nV_s)} = A'||\nu_{\ell-1}||^{\text{can}} + \frac{C'}{q},$$

hence $||\nu_{L-1}||^{\text{can}} \leq A'^{L-2}\left(A'D\sqrt{nB_{\mathsf{clean}}} + C'\right)/q$, where $A' = D\eta t\sqrt{\frac{n}{3}(1+nV_s)}$ and $C' = \frac{D^2t^2n}{12}(1+nV_s)$. Finally, as we require the norm to satisfy $||\nu_{L-1}||^{\text{can}} \leq 1/2$, the ciphertext modulus is computed as

$$q \geq 2A'^{L-2}(A'D\sqrt{nB_{\mathsf{clean}}} + C'). \tag{39}$$

*Average-case bounds.* In the average-case approach, we set

$$||\nu||_\infty \leq D\sqrt{2V}, \tag{40}$$

with the $V$ is the variance of each coefficient of $\nu$ and it depends of the homomorphic operations (see Table 4). Thanks to Equation (3), the bound holds with probability at least

$$1 - n\Big(1 - \text{erf}(D)\Big), \tag{41}$$

which for $D = 6$ is at least $1 - 2^{-40}$.

Summarizing the results of Section 4, let $\nu, \nu'$ 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. Then $V$ depends on the homomorphic operations, and its values are listed in Table 4.

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

Table 4: Variance depending on the homomorphic operations. $V_{\mathsf{ks}}(q_\ell)$ depends on the key switching variants (see Equations (22) to (24)) and $f$ is as Equation (27).

In Table 5 and 6, we show the estimations obtained with the worst-case error analysis employing the canonical norm [14, 21] (tagged with "can" ) and with our average-case approach ("our"). Moreover, we compare them with the experimental results ("exp"). In particular, in the columns labeled by "maximum value", we list the bounds computed as explained in Section 6.1 for the maximum coefficient of $\nu$ in absolute value. While in "mean value",

we compare our estimation of the mean value of the coefficients, computed as $\sqrt{\mathsf{Var}(\nu|_i)}$, with the experimental result. The chosen set of parameters is the following: the plaintext modulus is $t = 3$, the degree is $n = 2^\kappa$ where $12 \leq \kappa \leq 15$. Moreover, we take $D = 6$, $\chi_s = \chi_u = \mathcal{U}_3$ the ternary distribution, $\chi_e = \mathcal{DG}(0, \sigma^2)$ the discrete Gaussian distribution with standard deviation $\sigma = 3.2$ centred in 0 and GHS key switch.

In Table 5, we examine the error $\nu$ after encryption ("Encryption"), the addition of two fresh ciphertexts ("Addition"), and the multiplication of two fresh ciphertexts ("Multiplication").

| | | Encryption | | | | | Addition | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | maximum value | | mean value | | | maximum value | | mean value | |
| $n$ | $\log_2(q)$ | can | our | exp | our | exp | can | our | exp | our | exp |
| $2^{12}$ | 74 | 54.9 | 60.4 | 61.1 | 63.5 | 63.9 | 53.9 | 59.9 | 60.7 | 63.0 | 63.4 |
| $2^{13}$ | 149 | 128.9 | 134.9 | 135.7 | 138.0 | 138.4 | 127.9 | 134.4 | 135.1 | 137.5 | 137.86 |
| $2^{14}$ | 298 | 276.9 | 283.4 | 284.0 | 286.5 | 286.9 | 275.9 | 282.9 | 283.6 | 286.0 | 286.4 |
| $2^{15}$ | 597 | 574.9 | 581.9 | 582.5 | 585.0 | 585.4 | 573.9 | 581.4 | 582.0 | 584.5 | 584.9 |

| | | Multiplication | | | | |
|---|---|---|---|---|---|---|
| | | | maximum value | | mean value | |
| $n$ | $\log_2(q)$ | can | our | exp | our | exp |
| $2^{12}$ | 74 | 38.0 | 48.0 | 48.8 | 51.1 | 51.6 |
| $2^{13}$ | 149 | 111.0 | 121.6 | 122.3 | 124.7 | 125.1 |
| $2^{14}$ | 298 | 258.0 | 269.1 | 269.8 | 272.2 | 272.6 |
| $2^{15}$ | 597 | 555.0 | 566.6 | 567.3 | 569.7 | 570.1 |

Table 5: Comparison between the estimation of the noise budget using heuristic bounds obtained the canonical norm approach (can), our average-case analysis (our). The last column (exp) denoted the results from the experimental computations using our MAGMA code over 1000 trials.

In Table 6, the error is obtained from a Base Model circuit(see Figure 2) with multiplicative depth of either 2 or 3, with no constant multiplication and $\eta = 8$.

We want to emphasize the result of the first row in the circuit of multiplicative depth 3 of Table 6. According to the state-of-the-art estimation, we would not be able to perform any further operations since there would be no noise budget left. However, this analysis overestimates the error, as the experimental computation shows a remaining noise budget of 21 bits, while our estimation suggests a noise budget of at least 17.7 bits.

In Table 7, we present the bounds on the ciphertext modulus $q$ obtained using both approaches (Equations (31) and (39)). It is important to emphasize that

| | | 2 multiplications | | | | | 3 multiplications | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | maximum value | | mean value | | | maximum value | | mean value | |
| $n$ | $\log_2(q)$ | can | our | exp | our | exp | can | our | exp | our | exp |
| $2^{12}$ | 74 | 18.8 | 32.2 | 33.5 | 35.3 | 35.9 | 0.7 | 17.7 | 19.1 | 20.8 | 21.4 |
| $2^{13}$ | 149 | 90.8 | 104.7 | 106.0 | 107.8 | 108.4 | 71.7 | 89.2 | 90.3 | 92.3 | 92.8 |
| $2^{14}$ | 298 | 236.8 | 251.3 | 252.3 | 254.4 | 254.8 | 216.7 | 234.7 | 235.9 | 237.8 | 238.4 |
| $2^{15}$ | 597 | 532.8 | 547.8 | 548.9 | 550.9 | 551.3 | 511.7 | 530.3 | 531.5 | 533.3 | 533.9 |

Table 6: Comparison between the estimation of the noise budget in the Base Model (Figure 2) with $\alpha = 1$ and $\eta = 8$, using heuristic bounds obtained the canonical norm approach (can), our average-case analysis (our). The last column (exp) denoted the results from the experimental computations using our MAGMA code over 1000 trials.

having a smaller value of $q$ is crucial for both the efficiency and security of the scheme. As seen from Table 7, our approach yields significantly smaller values of $q$ compared to the norm trace approach, highlighting the importance of our new approach for improving the efficiency and security of the FV scheme. Note that in Table 7, we set $D = 8$ to have a failure probability smaller than $2^{-80}$ (see Equation (41)), which is usually required in a practical scenario. Moreover, we use a Base Model circuit (Figure 2) of depth $M = 3$ (i.e., $L = 4$) with $\alpha = 1$ and $\eta = 8$.

| | $\log_2(q)$ | | | $\log_2(q)$ | |
|---|---|---|---|---|---|
| $n$ | can | our | $n$ | can | our |
| $2^{12}$ | 75.0 | 56.7 | $2^{14}$ | 83.0 | 63.7 |
| $2^{13}$ | 79.0 | 60.2 | $2^{15}$ | 87.0 | 67.2 |

Table 7: Comparison between the ciphertext modulus $q$, using the heuristic bounds obtained by the worst-case approach (can) [14, 21] and our average-case analysis (our) in the Base Model (Figure 2) with $\alpha = 1$ and $\eta = 8$, $M = 3$ and $D = 8$.

### 6.2 A Parameter Generator for FV

To make our work more valuable and approachable for practical purposes, we provide automated parameter generation implemented in Python and publicly available on GitHub [9]. The interactive mode of the parameter generator prompts the user with a number of questions. We list required inputs in the first part and optional inputs in the second part of Table 8.

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

| | |
|---|---|
| Model | `'Base'`, `'Model1'`, `'Model2'`, `'OpenFHE'` |
| $t$ or $\log t$ | any integer $\geq 2$ |
| $\lambda$ or $m$ | any integer $\geq 40$ or $\geq 4$, respectively |
| $M$, $\eta$ | any integer $> 0$ |
| $\tau$ | any integer $\geq 0$ |
| Library | `'None'`, `'OpenFHE'`, `'PALISADE'`, `'SEAL'` |
| Full Batching | full batching with $t$, `'True'` or `'False'` |
| Secret Distribution | `'Ternary'`, `'Error'` |
| Key Switching | `'Hybrid'`, `'BV'`, `'GHS'` |
| $\beta$ | any integer $\geq 2$ |
| $\omega$ | any integer $\geq 1$ |

Table 8: Required and optional inputs to the parameter generator

We use the approach by Mono *et al.* [24] to estimate security. Additionally, to support arbitrary circuit models, we adopt their approach to estimate the key switching noise: we use fix values for $\beta$ and $\omega$, per default $\beta = 2^{10}$ and $\omega = 3$. If applicable, we set the extension modulus $P$ to be roughly equal to the noise produced by the ciphertext modulus depending on the key switching variant and scale it by a constant $K$, per default $K = 100$. Now, we can use this estimate for the key switching modulus to compute the noise bound programmatically. Note that we slightly overestimate the error this way and that the error growth from key switching is rather small compared to other operations, thus using this estimate results in valid parameter sets.

## 7 Conclusion

Selecting optimal, secure, and efficient parameters for a specific Fully Homomorphic Encryption scheme can be challenging.

This work presents several significant contributions to parameter selection in the Fan-Vercauteren scheme. We propose a new approach to average-case noise estimation, which significantly improves the accuracy and tightness of the bounds compared to previous works. In addition, we combine our theoretical analysis with the security formula proposed in [24] to develop the first flexible and user-friendly parameter generator for the FV scheme.

Overall, our work advances the state-of-the-art parameter selection for the FV scheme and provides a powerful tool that can assist in selecting efficient and reliable parameters for the FV scheme, making the task significantly more efficient and accessible for researchers and practitioners.

# Bibliography


[1] Martin R Albrecht, Melissa Chase, Hao Chen, Jintai Ding, Shafi Goldwasser, Sergey Gorbunov, Shai Halevi, Jeffrey Hoffstein, Kim Laine, Kristin Lauter, Satya Lokam, Daniele Micciancio, Dustin Moody, Travis Morrison, Amit Sahai, and Vinod Vaikuntanathan. Homomorphic encryption security standard. Technical report, `HomomorphicEncryption.org`, Toronto, Canada, November 2018.

[2] Martin R Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.

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

[4] Jean-Claude Bajard, Julien Eynard, M Anwar Hasan, and Vincent Zucca. A full RNS variant of FV like somewhat homomorphic encryption schemes. In *International Conference on Selected Areas in Cryptography*, pages 423–442. Springer, 2016.

[5] Loris Bergerat, Anas Boudi, Quentin Bourgerie, Ilaria Chillotti, Damien Ligier, Jean-Baptiste Orfila, and Samuel Tap. Parameter Optimization & Larger Precision for (T) FHE. *Cryptology ePrint Archive*, 2022.

[6] Zvika Brakerski. Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, pages 868–886, Berlin, Heidelberg, 2012. Springer.

[7] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. *ACM Transactions on Computation Theory (TOCT)*, 6(3):1–36, 2014.

[8] Zvika Brakerski and Vinod Vaikuntanathan. Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, pages 505–524, Berlin, Heidelberg, 2011. Springer.

[9] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. Homomorphic encryption for arithmetic of approximate numbers. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017*, pages 409–437, Cham, 2017. Springer International Publishing.

[10] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds. In *international conference on the theory and application of cryptology and information security*, pages 3–33. Springer, 2016.

[11] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. TFHE: fast fully homomorphic encryption over the torus. *Journal of Cryptology*, pages 1–58, 2019.

[12] Ana Costache and Nigel P Smart. Which ring based somewhat homomorphic encryption scheme is best? In *Cryptographers' Track at the RSA Conference*, pages 325–340. Springer, 2016.

[13] Anamaria Costache, Benjamin R Curtis, Erin Hales, Sean Murphy, Tabitha Ogilvie, and Rachel Player. On the precision loss in approximate homomorphic encryption. *Cryptology ePrint Archive*, 2022.

[14] Anamaria Costache, Kim Laine, and Rachel Player. Evaluating the effectiveness of heuristic worst-case noise analysis in FHE. In *European Symposium on Research in Computer Security*, pages 546–565. Springer, 2020.

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

[16] Andrea di Giusto. A study of the BGV scheme for non-power-of-two cyclotomic polynomials. In *To Appear*, 2023.

[17] Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. *IACR Cryptology ePrint Archive*, 2012.

[18] Craig Gentry. *A fully homomorphic encryption scheme*, volume 20. Stanford university Stanford, 2009.

[19] Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic Evaluation of the AES Circuit. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, pages 850–867, Berlin, Heidelberg, 2012. Springer.

[20] Shai Halevi, Yuriy Polyakov, and Victor Shoup. An improved RNS variant of the BFV homomorphic encryption scheme. In *Cryptographers' Track at the RSA Conference*, pages 83–105. Springer, 2019.

[21] Ilia Iliashenko. Optimisations of fully homomorphic encryption. PhD thesis, 2019.

[22] Andrey Kim, Yuriy Polyakov, and Vincent Zucca. Revisiting homomorphic encryption schemes for finite fields, 2021.

[23] Chiara Marcolla, Victor Sucasas, Marc Manzano, Riccardo Bassoli, Frank H. P. Fitzek, and Najwa Aaraj. Survey on Fully Homomorphic Encryption, Theory, and Applications. *Proceedings of the IEEE*, pages 1–38, 2022.

[24] Johannes Mono, Chiara Marcolla, Georg Land, Tim Güneysu, and Najwa Aaraj. Finding and Evaluating Parameters for BGV. *Cryptology ePrint Archive*, 2022.

[25] Sean Murphy and Rachel Player. A central limit approach for ring-lwe noise analysis.

[26] Microsoft SEAL (release 3.4). `https://github.com/Microsoft/SEAL`, October 2019. Microsoft Research, Redmond, WA.

# A Characterization of the error

**Proposition 1.** *The noise invariant can always be written as*

$$\nu = \sum_{\iota} a_\iota s^\iota$$

*and enjoys the following properties*

1. $\mathbb{E}[a_\iota|_i] = 0$ *for any* $\iota$,
2. $\mathsf{Cov}(a_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) = 0$ *if either* $\iota_1 \neq \iota_2$ *or* $i_1 \neq i_2$.

*Proof.* **Fresh ciphertext** Recall that the noise after the encryption is

$$\nu_{\mathsf{clean}} \overset{(6)}{=} \frac{t}{q}(\varepsilon + eu + e_0 + e_1 s).$$

We can rewrite it as $a_0 + a_1 s$ with $a_0 = \frac{t}{q}(\varepsilon + eu + e_0)$ and $a_1 = \frac{t}{q}e_1$.

1. Since $e_1|_i \leftarrow \chi_e$ and $\chi_e$ is symmetric, we have $\mathbb{E}[e_1|_i] = 0$, then

$$\mathbb{E}[a_1|_i] = \frac{t}{q}\mathbb{E}[e_1|_i] = 0.$$

Analogously, $\mathbb{E}[\varepsilon|_i] = \mathbb{E}[e|_i] = \mathbb{E}[u|_i] = \mathbb{E}[e_0|_i] = 0$ and all the coefficients of these polynomial are independent among each other, so

$$\mathbb{E}[a_0|_i] \overset{\text{Sec. 2.2}}{=} \frac{t}{q}(\mathbb{E}[\varepsilon|_i] + \sum_{j=0}^{n-1} \xi(i,j)\mathbb{E}[e|_j]\mathbb{E}[u|_{i-j \bmod n}] + \mathbb{E}[e_0|_i]) = 0.$$

See Section 4 for the definition of $\xi(i,j)$.

2. By the independence of the coefficients of $e_1$ among each other and with the coefficients of the other polynomial and by the covariance bilinearity (see Section 2.2), we get $\mathsf{Cov}(a_1|_{i_1}, a_1|_{i_2}) = 0$ for $i_1 \neq i_2$ and $\mathsf{Cov}(a_0|_{i_1}, a_1|_{i_2}) = 0$ for any $i_1, i_2$. Again for independence of the coefficients and the bilinearity of the covariance, we have

$$\mathsf{Cov}(a_0|_{i_1}, a_0|_{i_2}) = \frac{t^2}{q^2}\mathsf{Cov}((eu)|_{i_1}, (eu)|_{i_2}).$$

Since $(eu)|_i = \sum_{j=0}^{n-1} \xi(i,j) e|_j u|_{i-j \bmod n}$ (we will avoid to write the mod $n$ from now on),

$$\mathsf{Cov}((eu)|_{i_1}, (eu)|_{i_2}) = \sum_{j_1,j_2=0}^{n-1} \xi(i_1,j_1)\xi(i_2,j_2)\mathsf{Cov}(e|_{j_1}u|_{i_1-j_1}, e|_{j_2}u|_{i_2-j_2}) =$$

$$= \sum_{j_1,j_2=0}^{n-1} \xi(i_1,j_1)\xi(i_2,j_2)\Big(\mathbb{E}[e|_{j_1}e|_{j_2}]\mathbb{E}[u|_{i_1-j_1}u|_{i_2-j_2}] + $$

$$- \mathbb{E}[e|_{j_1}]\mathbb{E}[e|_{j_2}]\mathbb{E}[u|_{i_1-j_1}]\mathbb{E}[u|_{i_2-j_2}]\Big) = 0,$$

as either $j_1 \neq j_2$, hence $\mathbb{E}[e|_{j_1} e|_{j_2}] = \mathbb{E}[e|_{j_1}]\mathbb{E}[e|_{j_2}]$, or $i_1 - j_1 \bmod n \neq i_2 - j_2 \bmod n$, so $\mathbb{E}[u|_{i_1-j_1} u|_{i_2-j_2}] = \mathbb{E}[u|_{i_1-j_1}]\mathbb{E}[u|_{i_2-j_2}]$.

**Addition** Let $\nu = \sum_\iota a_\iota s^\iota$, $\nu' = \sum_{\iota'} a'_{\iota'} s^{\iota'}$ be two errors as claimed in the proposition, then $\nu_{\mathsf{add}} = \sum_\iota (a_\iota + a'_\iota) s^\iota$ with

1. $\mathbb{E}[(a_\iota + a'_\iota)|_i] = \mathbb{E}[a_\iota|_i] + \mathbb{E}[a'_\iota|_i] = 0$,
2. If $\iota_1 \neq \iota_2$ or $i_1 \neq i_2$, $\mathsf{Cov}((a_{\iota_1} + a'_{\iota_1})|_{i_1}, (a_{\iota_2} + a'_{\iota_2})|_{i_2})$ is equal to

   $$\mathsf{Cov}(a_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) + \mathsf{Cov}(a_{\iota_1}|_{i_1}, a'_{\iota_2}|_{i_2}) + \mathsf{Cov}(a'_{\iota_1}|_{i_1}, a_{\iota_2}|_{i_2}) + \mathsf{Cov}(a'_{\iota_1}|_{i_1}, a'_{\iota_2}|_{i_2}) = 0,$$

   because $a_{\iota_1}$ and $a'_{\iota_2}$ are independent and the other pairs are uncorrelated.

**Modulo switch & Key switch** The proof is analogous to the addition one by independence of the added quantity with the error $\nu$.

**Constant multiplication** Let $\nu = \sum_\iota a_\iota s^\iota$ be an error satisfying the properties above and $\alpha$ be a random element from $\mathcal{U}_t$, then $\alpha\nu = \sum_\iota \alpha a_\iota s^\iota$ and

1. Since the $\alpha$ and $a_\iota$ are independent with null expected value,

   $$\mathbb{E}\big[(\alpha a_\iota)|_i\big] \overset{\text{Sec. } 2.2}{=} \sum_{j=0}^{n-1} \xi(i,j)\mathbb{E}[\alpha|_j]\mathbb{E}[a_\iota|_{i-j}] = 0,$$

2. By bilinearity of the covariance, we have

   $$\mathsf{Cov}\Big((\alpha a_{\iota_1})|_{i_1}, (\alpha a_{\iota_2})|_{i_2}\Big) = \sum_{j_1,j_2=0}^{n-1} \xi(i_1,j_1)\xi(i_2,j_2)\mathsf{Cov}\big(\alpha|_{j_1} a_{\iota_1}|_{i_1-j_1}, \alpha|_{j_2} a_{\iota_2}|_{i_2-j_2}\big),$$

   with $\mathsf{Cov}\big(\alpha|_{j_1} a_{\iota_1}|_{i_1-j_1}, \alpha|_{j_2} a_{\iota_2}|_{i_2-j_2}\big)$ equal to

   $$\mathbb{E}[\alpha|_{j_1}\alpha|_{j_2}]\mathbb{E}[a_{\iota_1}|_{i_1-j_1} a_{\iota_2}|_{i_2-j_2}] - \mathbb{E}[\alpha|_{j_1}]\mathbb{E}[\alpha|_{j_2}]\mathbb{E}[a_{\iota_1}|_{i_1-j_1}]\mathbb{E}[a_{\iota_2}|_{i_2-j_2}].$$

   If $\iota_1 \neq \iota_2$ or $i_1 - j_1 \bmod n \neq i_2 - j_2 \bmod n$, then

   $$\mathbb{E}[a_{\iota_1}|_{i_1-j_1} a_{\iota_2}|_{i_2-j_2}] = \mathbb{E}[a_{\iota_1}|_{i_1-j_1}]\mathbb{E}[a_{\iota_2}|_{i_2-j_2}]$$

   by the second property of the error $\nu$.
   Otherwise, $j_1 \neq j_2$, by independence of the coefficients of $\alpha$,

   $$\mathbb{E}[\alpha|_{j_1}\alpha|_{j_2}] = \mathbb{E}[\alpha|_{j_1}]\mathbb{E}[\alpha|_{j_2}].$$

   It follows

   $$\mathsf{Cov}\Big((\alpha a_{\iota_1})|_{i_1}, (\alpha a_{\iota_2})|_{i_2}\Big) = 0.$$

**Multiplication** Let $\nu, \nu'$ as before, then $\nu\nu' = \sum_\iota \sum_{j+k=\iota} a_j a'_k s^\iota$.

1. Since $a_j$ and $a'_k$ are independent,
$$\mathbb{E}\big[\big(\sum_{j+k=\iota} a_j a'_k\big)|_i\big] = \sum_{j+k=\iota} \mathbb{E}[a_j]\mathbb{E}[a'_k] = 0.$$
2. For $\iota_1 \neq \iota_2$ or $i_1 \neq i_2$,
$$\mathsf{Cov}\Big(\big(\sum_{j_1+k_1=\iota_1} a_{j_1}a'_{k_1}\big)|_{i_1}, \big(\sum_{j_2+k_2=\iota_2} a_{j_2}a'_{k_2}\big)|_{i_2}\Big) =$$
$$= \sum_{j_1+k_1=\iota_1}\sum_{j_2+k_2=\iota_2}\sum_{l_1,l_2=0}^{n-1} \xi(i_1,l_1)\xi(i_2,l_2)\mathsf{Cov}(a_{j_1}|_{l_1}a'_{k_1}|_{i_1-l_1}, a_{j_2}|_{l_2}a'_{k_2}|_{i_2-l_2}),$$
where
$$\mathsf{Cov}(a_{j_1}|_{l_1}a'_{k_1}|_{i_1-l_1}, a_{j_2}|_{l_2}a'_{k_2}|_{i_2-l_2}) = \mathbb{E}[a_{j_1}|_{l_1}a_{j_2}|_{l_2}]\mathbb{E}[a'_{k_1}|_{i_1-l_1}a'_{k_2}|_{i_2-l_2}] + \\ - \mathbb{E}[a_{j_1}|_{l_1}]\mathbb{E}[a_{j_2}|_{l_2}]\mathbb{E}[a'_{k_1}|_{i_1-l_1}\mathbb{E}[a'_{k_2}|_{i_2-l_2}] = 0,$$
indeed, if $\iota_1 = \iota_2$ then $j_1 \neq j_2$ or $k_1 \neq k_2$, while if $i_1 \neq i_2$ then $i_1 - l_1 \bmod n \neq i_2 - l_2 \bmod n$ or $l_1 \neq l_2$.

Analogously, this holds for $\nu\frac{t}{q'_\ell}(c'_0+c'_1 s), \nu'\frac{t}{q_\ell}(c_0+c_1 s)$. Finally, we have that the covariance of different summands is 0, hence the conditions hold also for $\nu_{\mathsf{mul}} = -\nu\nu' + \nu\frac{t}{q'_\ell}(c'_0+c'_1 s) + \nu'\frac{t}{q_\ell}(c_0+c_1 s) + \frac{t}{q}(\varepsilon_0+\varepsilon_1 s+\varepsilon_2 s^2)$.

## B Properties of the function $g$

Let us define $f(0) = f(1) = 1$ and $g(\iota) = \prod_{i=0}^{\iota} f(i)$ for more generality.

**Proposition 2.** *Let $\iota_1 = 0,\dots,T_1$ and $\iota_2 = 0,\dots,T_2$, then*
$$\frac{g(\iota_1+\iota_2)}{g(\iota_1)g(\iota_2)} \leq \frac{g(T_1+T_2)}{g(T_1)g(T_2)}.$$

*Proof.* Let us fix $\iota_1$ and consider $\iota_2, \iota'_2$ with $\iota_2 \leq \iota'_2$. Since $f$ is an increasing function, we have $f(\iota_2+i) \leq f(\iota'_2+i)$, then
$$\frac{g(\iota_1+\iota_2)}{g(\iota_2)} = f(\iota_2+1)\cdots f(\iota_1+\iota_2) \leq f(\iota'_2+1)\cdots f(\iota_1+\iota'_2) = \frac{g(\iota_1+\iota'_2)}{g(\iota'_2)}.$$
It follows, in particular,
$$\frac{g(\iota_1+\iota_2)}{g(\iota_1)g(\iota_2)} \leq \frac{g(\iota_1+T_2)}{g(\iota_1)g(T_2)}.$$
Analogously, we get
$$\frac{g(\iota_1+T_2)}{g(\iota_1)g(T_2)} \leq \frac{g(T_1+T_2)}{g(T_1)g(T_2)}.$$

**Proposition 3.** *Let $T_1, T_2 \in \mathbb{N}$, then*

$$\frac{g(T_1+T_2)}{g(T_1+1)g(T_2+1)} \leq K_n < +\infty.$$

*Proof.* Let us assume $T_1 \leq T_2$ and observe that the values that $\frac{g(T_1+T_2)}{g(T_1+1)g(T_2+1)}$ can take are

$$\begin{aligned}
&\frac{g(T_1+T_2)}{g(1)g(T_1+T_2+1)} = \frac{1}{f(T_1+T_2+1)} \quad \text{if } T_1 = 0, T_2 = T_1+T_2,\\
&\frac{g(T_1+T_2)}{g(2)g(T_1+T_2)} = \frac{1}{f(2)} \quad \text{if } T_1 = 1,\ T_2 = T_1+T_2-1,\\
&\frac{g(T_1+T_2)}{g(3)g(T_1+T_2-1)} = \frac{f(T_1+T_2)}{f(2)f(3)} \quad \text{if } T_1 = 2,\ T_2 = T_1+T_2-2,\\
&\frac{g(T_1+T_2)}{g(4)g(T_1+T_2-2)} = \frac{f(T_1+T_2)f(T_1+T_2-1)}{f(2)f(3)f(4)} \quad \text{if } T_1 = 3,\ T_2 = T_1+T_2-3,\\
&\qquad\vdots\\
&\frac{g(T_1+T_2)}{g(\lfloor\frac{T_1+T_2}{2}\rfloor+1)g(\lceil\frac{T_1+T_2}{2}\rceil+1)} = \frac{f(\lceil\frac{T_1+T_2}{2}\rceil+2)\cdots f(T_1+T_2)}{f(2)\cdots f(\lfloor\frac{T_1+T_2}{2}\rfloor+1)} \quad \text{if } T_1 = \lfloor\frac{T_1+T_2}{2}\rfloor,\ T_2 = \lceil\frac{T_1+T_2}{2}\rceil,
\end{aligned}$$

which are in increasing order. Hence

$$\begin{aligned}
\frac{g(T_1+T_2)}{g(T_1+1)g(T_2+1)} &\leq \frac{f(\lceil\frac{T_1+T_2}{2}\rceil+2)\cdots f(T_1+T_2)}{f(2)\cdots f(\lfloor\frac{T_1+T_2}{2}\rfloor+1)} =\\
&= \frac{1}{f(\lfloor\frac{T_1+T_2}{2}\rfloor+1)} \prod_{\iota=2}^{\lfloor\frac{T_1+T_2}{2}\rfloor} \frac{f(\lceil\frac{T_1+T_2}{2}\rceil+\iota)}{f(\iota)}
\end{aligned}$$

We set $\tau = \lfloor\frac{T_1+T_2}{2}\rfloor$, $c_\iota = c - \frac{1}{e^{a\iota-b}} = f(\iota)$ and $\varepsilon_\iota = \left(1 - \frac{1}{e^{a\tau-b}}\right)\frac{1}{e^{a\iota-b}}$, then

$$\frac{g(T_1+T_2)}{g(T_1+1)g(T_2+1)} \leq \frac{1}{c_{(\tau+1)}} \prod_{\iota=2}^{\tau} \frac{c_\iota+\varepsilon_\iota}{c_\iota}.$$

Since $\frac{c_\iota+\varepsilon_\iota}{c_\iota} \geq 1$, we have that

$$\prod_{\iota=2}^{\tau} \frac{c_\iota+\varepsilon_\iota}{c_\iota} \leq \exp\left(\sum_{\iota=2}^{\tau} \frac{\varepsilon_\iota}{c_\iota}\right).$$

Now, comparing $\frac{\varepsilon_i}{c_i}$ and $\frac{\varepsilon_{i+1}}{c_{i+1}}$ we get

$$\begin{aligned}
\sum_{i=2}^{\tau} \frac{\varepsilon_i}{c_i} &\leq \frac{\varepsilon_2}{c_2} \sum_{i=2}^{\tau} \left(\frac{1}{e^a}\right)^i \leq \frac{\varepsilon_2}{c_2}\left(\frac{e^a}{e^a-1}\cdot\frac{e^{\tau a}-1}{e^{\tau a}} - \frac{e^a+1}{e^a}\right)\\
&\leq \frac{\varepsilon_2}{c_2}\left(\frac{1}{e^{2a}-e^a}\right) \leq \frac{e^b}{(e^{2a}c-e^b)(e^{2a}-e^a)}.
\end{aligned}$$

We computed the value of the fraction

$$\frac{f(\lceil \frac{T_1+T_2}{2} \rceil + 2) \cdots f(T_1+T_2)}{f(2) \cdots f(\lfloor \frac{T_1+T_2}{2} \rfloor + 1)},$$

for $T_1 + T_2 = 2^{20}$, obtaining the following upper bounds (varying $n$):

| $n$ | $2^{12}$ | $2^{13}$ | $2^{14}$ | $2^{15}$ |
|---|---|---|---|---|
| $K_n$ | 22 | 38 | 70 | 133 |

Table 9: Values of $K_n$