On the Concrete Security of Approximate FHE with Noise-Flooding Countermeasures

Flavio Bergamaschi[1], Anamaria Costache[2⋆], Dana Dachman-Soled[3⋆⋆], Hunter Kippen[3⋆⋆⋆], Lucas LaBuff[3], and Rui Tang[3]

[1] Intel Labs
flavio@intel.com
[2] Norwegian University of Science and Technology
anamaria.costache@ntnu.no
[3] University of Maryland
{danadach@, hkippen@, llabuff@terpmail., ruitang@}umd.edu

Abstract. Approximate fully homomorphic encryption (FHE) schemes such as the CKKS scheme (Asiacrypt '17) are popular in practice due to their efficiency and utility for machine learning applications. Unfortunately, Li and Micciancio (Eurocrypt, '21) showed that, while achieving standard semantic (or $\mathsf{IND\text{-}CPA}$ security), the CKKS scheme is broken under a variant security notion known as $\mathsf{IND\text{-}CPA}^D$. Subsequently, Li, Micciancio, Schultz, and Sorrell (Crypto '22) proved the security of the CKKS scheme with a noise-flooding countermeasure, which adds Gaussian noise of sufficiently high variance before outputting the decrypted value. However, the variance required for provable security is very high, inducing a large loss in message precision.

In this work, we ask whether there is an intermediate noise-flooding level, which may not be provably secure, but allows to maintain the performance of the scheme, while resisting known attacks. We analyze the security with respect to different adversarial models and various types of attacks.

We investigate the effectiveness of lattice reduction attacks, guessing attacks and hybrid attacks with noise-flooding with variance ρ^2_{circ}, the variance of the noise already present in the ciphertext as estimated by an average-case analysis, $100 \cdot \rho^2_{\mathsf{circ}}$, and $t \cdot \rho^2_{\mathsf{circ}}$, where t is the number of decryption queries. For noise levels of ρ^2_{circ} and $100 \cdot \rho^2_{\mathsf{circ}}$, we find that a full guessing attack is feasible for all parameter sets and circuit types. We find that a lattice reduction attack is the most effective attack for noise-flooding level $t \cdot \rho^2_{\mathsf{circ}}$, but it only induces at most a several bit reduction in the security level.

Due to the large dimension and modulus in typical FHE parameter sets, previous techniques even for *estimating* the concrete security of these attacks – such as those in (Dachman-Soled, Ducas, Gong, Rossi, Crypto '20) – become computationally infeasible, since they involve high dimensional and high precision matrix multiplication and inversion. We therefore develop new techniques that allow us to perform fast security estimation, even for FHE-size parameter sets.

⋆ Supported in part by Intel through the Intel Labs Soteria Research Collaboration

⋆⋆ Supported in part by NSF grant #CNS-2154705 and by Intel through the Intel Labs Crypto Frontiers Research Center.

⋆⋆⋆ Supported in part by the Clark Doctoral Fellowship from the Clark School of Engineering, University of Maryland, College Park.

1 Introduction

The notion of "approximate FHE" – fully homomorphic encryption schemes that guarantee only approximate correctness of decryption – was put forth by Cheon, Kim, Kim, and Song [13]. Their proposed scheme, henceforth referred to as CKKS, is one of the leading schemes in terms of efficiency, in particular in terms of suitability for Machine Learning (ML) tasks, as well as its paralellisation capabilities. Unfortunately, in [26], it was pointed out that approximate FHE schemes come with added risks: While they indeed achieve the standard notion of CPA-security, they can fail against a variant, $\mathsf{IND\text{-}CPA}^D$ introduced by Li and Micciancio [26], in which the adversary is given limited access to the decryption oracle. In the same work [26], the authors show that for exact schemes (such as BGV, BFV and TFHE), the notions of $\mathsf{IND\text{-}CPA}^D$ and $\mathsf{IND\text{-}CPA}$ are equivalent.[4]

Noise-flooding techniques have been suggested as a practical countermeasure against $\mathsf{IND\text{-}CPA}^D$ attacks [1]. These techniques add noise (from a Gaussian distribution) to the message obtained by decrypting a ciphertext, before it is returned to the adversary. Such countermeasures were formally analyzed in the work of Li, Micciancio, Schultz, and Sorrell [27], and it was shown that when the noise-flooding level is sufficiently high, they are indeed provably secure.

Nevertheless, the amount of noise required for provable security remains very high, and as a result, CKKS may lose some of the efficiency that originally made it attractive in comparison to exact FHE schemes. The main exact FHE schemes today are BGV [11], BFV [10,19], and TFHE [14]. A precise comparison between the schemes is hard to provide, since these schemes all have different trade-offs in terms of latency, amortized latency and the type of circuits that they can support. CKKS typically performs quite well in terms of amortized latency – see for example [9]. Therefore, while quantifying exactly how CKKS relates to these schemes with a much smaller message precision is not easily done, it certainly loses at least part of its edge over the other schemes in terms of amortized latency when noise-flooding is introduced. To bridge this gap, we ask whether there is an intermediate noise-flooding level, which may fault short of provable security, but which withstands a rigorous security analysis and which affords the efficiency needed in practice.

Our goal in this work is to investigate the concrete security degradation of the CKKS scheme when an adversary observes some number of decryptions, t, with noise flooding variance of some variance, ρ^2. The optimal setting of ρ^2 in terms of message precision is to set ρ^2 equal to the variance of the noise already present in an honestly generated ciphertext, since this means that only 1 additional bit of message precision is lost. On the other side of the spectrum is setting ρ^2 as large as the variance needed for provable, statistical security. We also investigate settings of ρ^2 that fall between these two extremes. Our aim is to present tradeoffs among (1) the number of allowed decryptions before the secret/public key must be refreshed, (2) the variance of the noise-flooding added to the decryption (which determines the loss of precision), and (3) the concrete security of the scheme after a number of decryptions have been observed by the adversary (e.g. a drop of 10 or 15 bits in security for a 256-bit parameter set may still be acceptable).

In the next section, we discuss in more detail the adversarial model we consider and our methodology for determining the concrete hardness of key-recovery after some number of decryptions have been observed. We emphasize that prior methods for providing concrete hardness estimates such as [17], require performing matrix operations on the covariance matrix representing the conditional distribution of the LWE secret/error. For FHE-scale parameter sets, the covariance matrix can have dimension as high as $256K \times 256K$ and thus several hundred terrabytes are required to naively store the values (this is assuming 64-bit precision, whereas in our experimental results in Section 9.2, we find that up to 2,000 bit precision is required for meaningful results). Therefore, one of our main contributions is developing new tools to provide fast and accurate estimates that do not require these high-dimensional matrix operations.

[4] In a recent work [12], this is called into question, as the authors point out that that the proof of equivalence between $\mathsf{IND\text{-}CPA}^D$ and $\mathsf{IND\text{-}CPA}$ does not take into account the decryption failure probability of an exact scheme. The authors of [12] exploit the fact that this decryption failure probability is rather high in implementations of exact schemes to run an $\mathsf{IND\text{-}CPA}^D$ attack on the BFV scheme, and remark that their attack also applies to BGV and TFHE.

1.1 Our Methodology

In our work, we consider several types of adversarial models and attacks. The choice of adversarial model governs how the adversary is allowed to interact with the encryption/eval/decryption functionalities, while the choice of attack corresponds to the way the information obtained from the decryption oracle is used to perform key recovery. Importantly, we note that all our adversarial models are *semi-honest* in the sense that the adversary obtains fresh ciphertexts sampled from the correct distribution, and only requests computations on circuits whose inputs are fresh, independent, ciphertexts. Thus, the estimates of the noise present in the ciphertext (which determine the variance for the noise-flooding) and the actual noise present in every ciphertext submitted for decryption are consistent. Such a constraint could be enforced in practice by requiring zero knowledge proofs of well-formedness of fresh ciphertexts and/or signatures of designated parties to be checked before decryption is performed, or Verifiable Computation (VC) techninques [8,20,21]. This is different from attacks such as those of Guo, Nabokov, Suvanto, and Johansson [22], which work by constructing adversarial ciphertexts with noise distribution that is far from the noise distribution estimated during the noise flooding step.

The first adversarial model we consider allows the adversary to run the encryption algorithm honestly and to request decryptions of "fresh" ciphertexts, i.e. the adversary queries the decryption oracle on the identity circuit. The second adversarial model we consider allows the adversary black-box access to the encryption algorithm, and to request decryptions of ciphertexts resulting from the evaluation of circuits from one of two circuit classes on the encryptions. The attacks we consider are lattice reduction attacks, guessing attacks, and hybrid attacks. We elaborate below on the adversarial models, attacks, and our methodology for analyzing the concrete hardness of each attack.

Decryption Queries on Identity Circuits. We start by considering an attacker who submits a number t of fresh ciphertexts for decryption. This can also be viewed as an attacker who requests a decryption of a ciphertext corresponding to the evaluation of the identity circuit on a fresh encryption. In the original paper of Li and Micciancio [26], these simple $\mathsf{IND\text{-}CPA}^D$ attacks were already shown to allow full key-recovery against CKKS. We, however, consider a strengthening of their adversarial model. They allowed the adversary only black-box access to the encryption oracle. We assume that the adversary obtains "white-box" access to the encryption oracle, namely the internal randomness of the encryption is returned to the adversary, along with the ciphertext. Once the ciphertexts have been created, our attacker observes decryptions with "noise-flooding" added before the message is returned, where the noise is a centered Gaussian of variance ρ^2. After observing this information, we consider the concrete security of a key recovery attack under three types of attacks: (1) Lattice Reduction attacks, (2) Guessing attacks, (3) Hybrid attacks.

Lattice Reduction Attacks. Here we assume that the adversary embeds the original LWE instance and the additional information that it obtains (which we refer to as "hints") into a DBDD instance (introduced by [17]), which is then transformed into a u-SVP instance. We note that the "hints" consist of noisy linear equations on the LWE secret/error, where the noise is sampled form a Gaussian distribution. Therefore, the conditional distribution on the LWE secret/error, given the hints, remains a Gaussian distribution and a closed-form formula for the new distribution can be obtained from known techniques. Thus, the steps to integrate the hints and transform the DBDD instance to a u-SVP instance follow those given in [17] for the case of conditional, full-dimensional, approximate hints. Upon obtaining the resulting u-SVP instance, the adversary then uses the BKZ algorithm to recover the shortest vector which corresponds to the LWE secret/error. We provide concrete security guarantees in terms of bikz (i.e. BKZ-β) required to solve the final u-SVP instance, as well as the bit-security.

Importantly, although the attack template proceeds as the one outlined in [17], our *analysis* of the attack differs. To obtain concrete security estimates as in [17], one would need to compute the determinant of a $2n \times 2n$ dimensional matrix that depends on the t ciphertexts submitted for decryption and the outputs observed by the adversary. For $n = 256$ and $t = 16$, our experiments showed that this computation takes roughly a week on a supercomputer (See Section 9.2). In contrast, typical FHE parameters sets can have dimension up to $\log_2(n) = 17$. Thus, to provide fast estimates, we analyze the *distribution* of the resulting $2n \times 2n$ dimensional matrix arising from the outlined attack. We provide a closed-form expression for the *expected* determinant of a matrix drawn from this distribution (See Section 5 and Lemma 5.1). We verify experimentally (See Section 9.2) that the predicted and actual expected determinant match closely, even though our prediction approximates the original secret and error distributions as Gaussian while they are in fact a uniform ternary distribution and a discrete Gaussian distribution, respectively. We believe this type of analysis is a crucial component for allowing concrete hardness estimates for FHE-size parameters.

Guessing Attacks. Here the attacker keeps track of the conditional multivariate Gaussian distribution on the LWE secret/error after integrating the t hints. When the variance of individual secret/error coordinates becomes small enough, the adversary rounds the coordinate of the mean of the multivariate Gaussian distribution to the nearest integer. At some point, the adversary can guess n out of $2n$ coordinates correctly with high probability, in which case it can solve the original LWE system to obtain the remaining n coordinates. Similarly to the lattice reduction case, actually keeping track of the covariance matrix of the multivariate Gaussian distribution requires a $2n \times 2n$ matrix inversion and is highly computationally intensive for FHE-scale parameters. Since we know the distribution of the matrix, we are able to derive bounds that hold with high probability on the trace and eigenvalues of the matrix, which in turn can be used to bound the success probability of the guessing attack, using the Gaussian correlation inequality [25] (See Section 6 and Lemma 6.1).

Hybrid Attacks. Here the attacker guesses $g < n$ number of coordinates as above, but cannot guess n of them w.h.p. The attacker integrates these g guesses as "perfect hints" into the DBDD instance and finally obtains a new u-SVP instance, which it then solves using lattice reduction. After integrating the guesses, the information known to the adversary corresponds to a principal submatrix of the covariance matrix, whose determinant we need to compute in order to estimate hardness. As before, we do not compute the actual $2n \times 2n$ covariance matrix for the instance, which is highly computationally intensive, but rather use the fact that the distribution of the covariance matrix is known. We use the Eigenvalue Interlacing Theorem (see e.g. [23]) and bounds on the eigenvalues that hold w.h.p. in order to bound the determinant of the principal submatrix, given the determinant of the entire matrix (See Section 7 and Lemma 7.1).

Broader Classes of Circuits. We extend our analysis to broader classes of circuits (see Section 8 for formal definitions of these classes). Briefly, Class 1 circuits are circuits that consist of ℓ independent subcircuits $C_1, \ldots, C_\ell$. These circuits can be completely arbitrary as long as they all have the same multiplicative depth $d \geq 1$ and they each end in a *multiplication with rescale* operation. The final circuit consists of the addition of the outputs of these subcircuits. Intuitively, we require addition of ℓ ciphertexts so that the noise coefficients, which are individually uniformly random between $[-0.5, 0.5]$, can be well-approximated by a Gaussian distribution. Class 2 circuits are circuits whose output corresponds to the *multiplication without rescale* of the outputs of two independent Class 1 circuits. Our motivation for considering Class 2 circuits is that in practice, a rescale is typically not performed in the final multiplication gate of the circuit, in order to reduce the size of the top-level modulus.

For circuits in Class 1 and 2, our adversarial model is captured by the $\mathsf{IND\text{-}CPA}^D$-definition. I.e. the adversary *does not need to know the internal randomness used by the encryption process*, and can launch the attack with only *black-box* access to the encryption algorithm. The analysis in this case is facilitated by the fact that it was shown in prior work [16,7] that after a **rescale** step, the *rounding* noise (which can be publicly computed) dominates the noise present in the ciphertext. Upon decryption, the information obtained by the adversary corresponds to an approximate linear equation on the secret, which induces a conditional Gaussian distribution on the secret. Thus, the information obtained is in fact a special case of the information obtained by decryptions of the identity circuit, which correspond to noisy linear equations on both the LWE secret and error. As before, we consider three types of attacks for each of the two classes of circuits: (1) Lattice Reduction attacks, (2) Guessing attacks, (3) Hybrid attacks.

1.2 Summary of Experimental Results

We performed extensive experimentation for a wide range of parameter sets proposed by the homomorphicencryption.org standards [2], as well as a larger parameter set with a ring dimension of $\log_2 n = 17$. In Section 9, we provide an experimental validation of Lemma 5.1, as well as tables detailing the effectiveness of each of the three attack types on fresh ciphertexts (identity circuits) at various noise-flooding levels: ρ^2_{circ}—the noise variance already present in a ciphertext—$100 \cdot \rho^2_{\mathsf{circ}}$, and $t \cdot \rho^2_{\mathsf{circ}}$, where t is the number of decryptions the attacker may observe. For additional tabular data on Class 1 and 2 circuits, readers may consult with our supplementary material, section A.

In Section 10, we provide a graphical representation of our results and highlight our key findings. Most notably, we find that with noise-flooding levels of ρ^2_{circ} and $100 \cdot \rho^2_{\mathsf{circ}}$, full guessing attacks are feasible after observing a sufficient number of decryption queries (at most $\sim$ 100K needed), for all parameter sets and types of circuits considered. On the other hand, for noise level of $t \cdot \rho^2_{\mathsf{circ}}$, lattice reduction attacks are the only effective attacks. Given the above, a noise-flooding magnitude of $\alpha \cdot t \cdot \rho^2_{\mathsf{circ}}$, where ρ^2_{circ} is the average-case noise present in a ciphertext output by a circuit circ, appears sufficient to preserve security when t decryptions are made available to the adversary. Tuning $0 < \alpha \leq 1$ establishes a way to enforce security-precision tradeoffs in concrete applications. Finally, we note that all attacks become less effective as $\log_2(n)$ increases. Establishing the "appropriate" noise-flooding values will therefore depend on the application itself, on the number of decryption queries t that may be available to an adversary, as well as the FHE parameters, in particular the ring dimension $\log_2(n)$.

1.3 Related Work

After the advent of the CKKS scheme, Li and Micciancio [26] demonstrated it is insecure under a variant of the $\mathsf{IND\text{-}CPA}$ security notion, which they called $\mathsf{IND\text{-}CPA}^D$. Their work left the door open as to whether noise-flooding countermeasures, in which additional noise is added to the decrypted message before it is returned, can patch the vulnerability. Li, Micciancio, Schultz, and Sorrell [27] proved the security of the CKKS scheme with the noise-flooding countermeasure for Gaussian noise with sufficiently high variance. However, the variance required for provable security is very high, inducing a large loss in message precision. Later, the inherent noise already present in a CKKS ciphertext was analyzed closely in [15]. This analysis allows for a better understanding of how much message precision is lost via the noise-flooding countermeasures.

The tools of incorporating side information on the LWE secret/error into a lattice reduction attack were developed in [17] via an introduction of an intermediate problem known as Distorted Bounded Distance Decoding (DBDD). Their framework allows the incorporation of "hints" into DBDD instances, which are finally converted to uSVP instances via homogenization/isotropization, and can be applied to analyze the concrete security of the CKKS scheme with noise-flooding countermeasures. However, in practice, keeping track of the intermediate DBDD instance is not feasible for FHE-scale parameters. The security estimation for the LWE problem was revisited in [18], but those techniques similarly do not scale to FHE-size parameter sets.

The work of Kim, Lee, Seo, and Song [24] considered the provable security of the Hint-LWE problem, and it can be observed that the information obtained from noisy decryptions of fresh ciphertexts can be viewed as an instance of Hint-LWE. Theorem 1 in [24] provides a security reduction from a spherical LWE instance to Hint-LWE. However, because the conditional Gaussian distribution arising from the Hint-LWE problem is ellipsoidal (not spherical), the reduction is not tight (additional noise is added to convert from the spherical to ellipsoidal distribution). This is in contrast to our approach, which provides an attack that first converts the Hint-LWE instance to a DBDD instance. Importantly, a DBDD instance with an ellipsoidal distribution is *equivalent* to another DBDD instance with a spherical distribution, and there is no loss in this reduction. Thus, our concrete security estimates are tighter, but only apply to certain classes of attack strategies. We also note that reduction in Theorem 1 of [24] is for decisional LWE, whereas our attacks are for the search LWE problem, which makes the two results somewhat incomparable.

Two recent works [22,12] present a key-recovery attack on the schemes CKKS and the exact FHE schemes, respectively. Both attacks rely on the following observation: an average-case noise analysis models all noise terms as independent Gaussians. When that assumption fails, the noise predicted by an average-case noise analysis will underestimate the actual noise observed. Indeed both works successfully run a key-recovery attack by using correlated inputs. We note that, while that research direction is interesting, this does not affect our setting. In particular, in all circuits we consider (the identity circuit, and the classes C1 and C2), the noise terms remain independent. We note that a recent work [5] argues that those attacks amount to incorrect estimation of the underlying ciphertext noise, as the heuristics specifically assume that inputs are independent, but [22,12] heavily rely on correlated inputs. The authors of [5] therefore define the notion of *application-aware* homomorphic encryption, that can precisely counter these types of attacks. Our work therefore fits well within their model.

2 Preliminaries and Notation

Notation. We use bold lower case letters to denote vectors, and bold upper case letters to denote matrices. We use row notation for vectors, and denote by $\mathbf{I}_d$ the identity matrix of dimension d. We denote by $\{\mathbf{e}_i\}_{i\in[n]}$ the standard basis vectors in dimension n.

We use the notation R_q to denote the ring $\mathbb{Z}[x]/(\Phi_m(x), q)$, where $\Phi_m(x) = x^n + 1$, and $n = \phi(m)$ is a power of two. We denote ring elements by lowercase, non-bolded letters. When we employ a particular vector representation of a ring element in the coefficient or canonical embedding, we use vector notation. $[\cdot]_q$ denotes modular reduction $\pmod q$ (usually centered around 0).

We will make use of the canonical embedding and the subspace $H \subseteq \mathbb{C}^{\mathbb{Z}_m^*}$ defined as follows:

$$H = \{\mathbf{x} = (x_i)_{i\in\mathbb{Z}_m^*} \in \mathbb{C}^n : x_i = \overline{x_{-i}}, \forall i \in \mathbb{Z}_m^*\}.$$

H is isomorphic to $\mathbb{R}^n$ as an inner product space via the unitary transformation

$$B = \begin{pmatrix} \frac{1}{\sqrt{2}}\mathbf{I} & \frac{i}{\sqrt{2}}\mathbf{J} \\ \frac{1}{\sqrt{2}}\mathbf{J} & \frac{-i}{\sqrt{2}}\mathbf{I} \end{pmatrix}$$

where $\mathbf{I}$ is the identity matrix of size $n/2$ and $\mathbf{J}$ is its reversal matrix.

The canonical embedding of $a \in \mathbb{Q}[x]/\Phi_m(x)$ into $\mathbb{C}^n$ is the vector of evaluations of a at the roots of $\Phi_m(x)$. Specifically $\sigma(a) = [a(\zeta^j)_{j\in\mathbb{Z}_m^*}]$, where ζ is a primitive m-th root of unity. Due to the conjugate pairs, σ maps into the subspace H. When a is represented as a vector of coefficients $\mathbf{a}$, we can express the canonical embedding transformation as a linear transformation $\mathbf{aV}$.

We denote by $\mathcal{N}(\mu, \mathbf{\Sigma})$ the multivariate Gaussian with mean μ and covariance $\mathbf{\Sigma}$. We note that a multivariate Gaussian is fully determined by its mean and covariance. Thus, when the covariance of a dim dimensional multivariate Gaussian is a multiple of $\mathbf{I}_{\mathsf{dim}}$, the dim variables are all independent.

DBDD and concrete hardness estimates. A DBDD instance (defined in [17]) consists of a tuple $(\Lambda, \mu, \mathbf{\Sigma})$, where Λ is a lattice, and $(\mu, \mathbf{\Sigma})$ are viewed as the mean and covariance of a Gaussian distribution. Informally, the DBDD problem asks to find the unique vector in the lattice Λ that is contained in the ellipsoid defined by (μ, Σ) (for the formal definition see [17]). The prior work of [17] showed how to transform a DBDD instance into a uSVP instance with lattice Λ' using the homogenization and isotropization steps, and further showed that the secret vector of this uSVP instance has expected squared norm $||\mathsf{s}||^2 = \mathsf{dim}(\Lambda')$. Thus, standard techniques can be used to estimate the hardness of the resulting uSVP instance, where hardness is measured in terms of the "bikz" or BKZ-β required to find the unique solution. In particular, following [3,6,17], β can be estimated as the minimum integer that satisfies

$$\sqrt{\beta} \leq \delta_\beta^{2\beta-\mathsf{dim}(\Lambda')-1}\mathsf{Vol}(\Lambda')^{1/\mathsf{dim}(\Lambda')} \tag{1}$$

for a lattice Λ' where δ is the root-Hermite-Factor of BKZ-β.

The CKKS scheme. See Appendix 3 for a detailed description of the CKKS encryption scheme as well as a derivation of the error terms present in the message when decrypting a fresh ciphertext, and when decrypting after one or more multiplication steps (with or without a rescale operation). Following [15], we also present the noise variance in a fresh CKKS ciphertext, and in a ciphertext resulting from a multiplication and rescale operation (See Appendix 3.5).

3 The CKKS scheme [13]

Let χ be a discrete Gaussian of standard deviation $\sigma = 3.2$. We denote by $\mathcal{ZO}(\rho)$ the distribution where 0 is sampled with probability ρ, and ± 1 are sampled with probability $\rho/2$. We denote the secret key distribution by S. This is the uniform ternary distribution.

We assume a sequence of moduli $q_L, \ldots, q_0$. After ℓ levels of multiplication, we obtain level ℓ ciphertexts with moduli $q_{\overline{\ell}}$, where $\overline{\ell} = L - \ell$. We note that, although we present encryption as being performed at the "top" level L, it can be performed at any level $\overline{\ell}$.

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

`PublicKeyGen`(`sk`): For $\mathtt{sk} = (1, s)$, sample $a \leftarrow R_q$ uniformly at random and $e \leftarrow \chi$. Output $\mathtt{pk} = ([-as + e]_{q_L}, a)$.

`EvaluationKeyGen`(`sk`, w): Set $s = \mathtt{sk}$. Sample $a' \leftarrow R_Q$, $(Q = Pq_L)$ uniformly at random and $e' \leftarrow \chi$. Output $\mathtt{evk} = \left([-a's + e' + Ps^2]_Q, a'\right)$.

`Encrypt`(`pk`, m): For the message $m \in R$. Let $\mathtt{pk} = (p_0, p_1)$, sample $v \leftarrow S$ and $e_0, e_1 \leftarrow \chi$. Output $\mathtt{ct} = ([m + p_0 v + e_0]_q, [p_1 v + e_1]_q)$.

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

`Add`($\mathtt{ct}_0, \mathtt{ct}_1$): Given two level ℓ ciphertexts, output $\mathtt{ct} = ([\mathtt{ct}_0[0] + \mathtt{ct}_1[0]]_{q_{\overline{\ell}}}, [\mathtt{ct}_0[1] + \mathtt{ct}_1[1]]_{q_{\overline{\ell}}})$.

`Pre-Multiply`($\mathtt{ct}_0, \mathtt{ct}_1$): Given two level ℓ ciphertexts, set

$$\begin{aligned}
d_0 &= [\mathtt{ct}_0[0]\mathtt{ct}_1[0]]_{q_{\overline{\ell}}} \\
d_1 &= [\mathtt{ct}_0[0]\mathtt{ct}_1[1] + \mathtt{ct}_0[1]\mathtt{ct}_1[0]]_{q_{\overline{\ell}}} \\
d_2 &= [\mathtt{ct}_0[1]\mathtt{ct}_1[1]]_{q_{\overline{\ell}}}
\end{aligned}$$

Output $\mathtt{ct} = (d_0, d_1, d_2)$.

`Relinearize`(`ct`, `evk`, P): Given level a level ℓ ciphertext as input, let $\mathtt{ct}[0] = d_0$, $\mathtt{ct}[1] = d_1$ and $\mathtt{ct}[2] = d_2$. Let $\mathtt{evk}[0] = -a's + e' + Ps^2$ and $\mathtt{evk}[1] = a'$. Set

$$\begin{aligned}
c_0' &= [d_0 + \lfloor P^{-1} \cdot d_2 \cdot (-a's + e' + Ps^2) \rceil]_{q_{\overline{\ell}}} \\
c_1' &= [d_1 + \lfloor P^{-1} \cdot d_2 \cdot a' \rceil]_{q_{\overline{\ell}}}
\end{aligned}$$

Output $\mathtt{ct}' = (c_0', c_1')$.

`Rescale`(`ct`, Δ) : Given level a level ℓ ciphertext as input, let $\mathtt{ct} = (c_0, c_1)$. Set $c_0' = \left[\lfloor \frac{c_0}{\Delta} \rceil\right]_{q_{\overline{\ell}-1}}$ and $c_1' = \left[\lfloor \frac{c_1}{\Delta} \rceil\right]_{q_{\overline{\ell}-1}}$. Output $\mathtt{ct} = (c_0', c_1')$.

3.1 Decrypting a fresh ciphertext

Let `ct` be a fresh ciphertext encrypted under the public key `pk`, where we have $\mathtt{pk} = ([-as + e]_{q_L}, a)$. Then, decrypting `ct` yields

$$\begin{aligned}
\mathtt{Decrypt}(\mathtt{ct}, \mathtt{sk}) &= c_0 + sc_1 \pmod{q_L} \\
&= m + p_0 v + e_0 + svp_1 + se_1 \\
&= m + ve + e_0 + se_1.
\end{aligned}$$

Recall that $e, e_0, e_1 \leftarrow \chi$. The ephemeral key v here is drawn from the same distribution as the secret key S, but sometimes it can be sampled from a slightly different distribution. This can for example be the distribution $\mathcal{ZO}(\rho)$.

3.2 Decrypting a multiplication, no rescale

Let $\texttt{ct} = (c_0, c_1)$ and $\texttt{ct}' = (c'_0, c'_1)$ be two level ℓ ciphertexts that decrypt as follows.

$$
\begin{aligned}
c_0 + sc_1 &= \frac{m^{2^\ell}}{\Delta^{2^\ell - 1}} + E \\
c'_0 + sc'_1 &= \frac{m^{2^\ell}}{\Delta^{2^\ell - 1}} + E'.
\end{aligned}
$$

Then, the output of $\texttt{Pre-Mult}$ is

$$
(d_0, d_1, d_2) = (c_0 c'_0, c_0 c'_1 + c'_0 c_1, c_1 c'_1).
$$

Note that this decrypts as

$$
\begin{aligned}
d_0 + sd_1 + s^2 d_2 &= (c_0 + c_1 s)(c'_0 + sc'_1) \\
&= \frac{m^{2^{\ell+1}}}{\Delta^{2^{\ell+1} - 2}} + \tilde{E},
\end{aligned}
$$

for some error $\tilde{E}$. Recall that the evaluation key is $\texttt{evk} = \left([-a's + e' + Ps^2]_Q, a'\right)$. Then, the output of $\texttt{Relinearize}$ is

$$
\begin{aligned}
C_0 &= d_0 + \lfloor P^{-1} \cdot d_2 \cdot (-a's + e' + Ps^2) \rceil \\
&= d_0 + P^{-1} \cdot d_2 \cdot (-a's + e' + Ps^2) + \epsilon_0 \\
C_1 &= d_1 + \lfloor P^{-1} \cdot d_2 \cdot a' \rceil \\
&= d_1 + P^{-1} \cdot d_2 \cdot a' + \epsilon_1,
\end{aligned}
$$

where ϵ_i are rounding errors. Decrypting this yields

$$
\begin{aligned}
C_0 + sC_1 &= d_0 + P^{-1} d_2(-a's + e' + Ps^2) + \epsilon_0 + sd_1 + sP^{-1} d_2 a' + s\epsilon_1 \\
&= d_0 + sd_1 + s^2 d_2 + (\epsilon_0 + \epsilon_2 s) + P^{-1} d_2 e' \\
&= \frac{m^{2^{\ell+1}}}{\Delta^{2^{\ell+1} - 2}} + \tilde{E} + (\epsilon_0 + \epsilon_1 s) + P^{-1} d_2 e'.
\end{aligned}
$$

It has been shown that for all the FHE parameter sets we consider, the error above is dominated by $\tilde{E} = E \cdot \frac{m^{2^\ell}}{\Delta^{2^\ell - 1}} + E' \cdot \frac{m^{2^\ell}}{\Delta^{2^\ell - 1}}$ [16,7].

3.3 Decrypting a multiplication, with rescale

From the previous subsection, we have that the noise after a $\texttt{Pre-Mult}$ and a $\texttt{Relin}$ is

$$
C_0 + sC_1 = \frac{m^{2^{\ell+1}}}{\Delta^{2^{\ell+1} - 2}} + E + (\epsilon_0 + \epsilon_1 s) + P^{-1} d_2 e'.
$$

We are going from level ℓ to level $\ell + 1$ and from modulus $q_{\bar{\ell}}$ to modulus $q_{\bar{\ell}-1}$. Following the notation of the previous subsection, we have the ciphertext

$$
(C_0, C_1) = \texttt{Relin}(\texttt{Pre-Mult}(\texttt{ct}, \texttt{ct}')).
$$

Let $(C'_0, C'_1) = \texttt{Rescale}(C_0, C_1) = ([\lfloor \frac{C_0}{\Delta} \rceil]_{q_{\bar{\ell}-1}}, \lfloor \frac{C_1}{\Delta} \rceil]_{q_{\bar{\ell}-1}})$. Then

$$\begin{aligned}
\mathtt{Dec}((C_0', C_1'), \mathtt{sk}) &= C_0' + sC_1' \\
&= \left\lfloor \frac{C_0}{\Delta} \right\rceil + s \left\lfloor \frac{C_1}{\Delta} \right\rceil \\
&= \frac{C_0}{\Delta} + s\frac{C_1}{\Delta} + \delta_0 + s\delta_1 \\
&= \frac{1}{\Delta}(C_0 + sC_1) + \delta_0 + s\delta_1 \\
&= \frac{1}{\Delta}(\frac{m^{2^{\ell+1}}}{\Delta^{2^{\ell+1}-2}} + E + (\epsilon_0 + \epsilon_1 s) + P^{-1}d_2 e') + \delta_0 + s\delta_1 \\
&= \frac{m^{2^{\ell+1}}}{\Delta^{2^{\ell+1}-1}} + \frac{1}{\Delta}(E + (\epsilon_0 + \epsilon_1 s) + P^{-1}d_2 e') + \delta_0 + s\delta_1,
\end{aligned}$$

where δ_i are rounding errors, and we omit a reduction modulo $q_{\bar{\ell}-1}$ throughout.

It was observed in [16,7] that the error above is typically dominated by $\delta_0 + s\delta_1$ for most parameter sets. When the resulting ciphertext (C_0', C_1') is a level ℓ' ciphertext, we denote the error as $E_{\ell'}$. Note that if an adversary knows the evaluation key $\mathtt{evk}$, then the adversary can compute δ_0 and δ_1 on its own. Further, each element of δ_0 and δ_1 can be assumed to be independently and uniformly distributed between $[-0.5, 0.5]$.

If the adversary does not know the evaluation key $\mathtt{evk}$, then it will be unable to gain information about the values of δ_0 and δ_1 as $\mathtt{evk}[1] = a'$ is sampled uniformly at random. If the adversary knows $\mathtt{evk}[0] = [-a's + e' + Ps^2]_Q$ but not $\mathtt{evk}[1]$, then it is able to calculate δ_0 exactly by computing C_0. However, by the LWE assumption, it is unable to learn a' and thus cannot determine δ_1. Similarly, knowing only $\mathtt{evk}[1]$ allows the adversary to compute δ_1 exactly but learn nothing about δ_0. In our attack model, as is standard, we will assume the adversary knows $\mathtt{evk}$.

3.4 Two or more multiplications, with no final rescale

Recall that our chain of ciphertext moduli are formed as follows. Let $q_0, \ldots, q_L$ be primes of roughly equal size. We recall that the size of the scaling parameter Δ is also roughly equal to each q_i. Then, for any level i, the ciphertext modulus Q_i is $Q_i = \prod_{j=0}^{i} q_j$. We encrypt "at the top" level Q_L, and go "down" one level after each multiplication.

Let $\mathtt{ct}_0$ and $\mathtt{ct}_1$ be two ciphertexts encrypting the same message m^{2^ℓ} at level ℓ, where $\ell > 0$. Note that this implies that the re-scale operation has been performed on $\mathtt{ct}_0$ and $\mathtt{ct}_1$ and so the error in each of these ciphertexts is $E_{\ell,0}, E_{\ell,1}$. From the previous subsections, we know that these errors are dominated by $\delta_{0,0} + s\delta_{1,0}$, and $\delta_{0,1} + s\delta_{1,1}$ respectively, for most parameter sets. Further, each element of $\delta_{0,0}, \delta_{1,0}, \delta_{0,1}$ and $\delta_{1,1}$ can be assumed to be independently and uniformly distributed between $[-0.5, 0.5]$.

$$\begin{aligned}
\mathsf{Dec}(\mathtt{ct}_0, \mathtt{sk}) &= \frac{m^{2^\ell}}{\Delta^{2^\ell-1}} + E_{\ell,0} \pmod{q_{\bar{\ell}}} \\
\mathsf{Dec}(\mathtt{ct}_1, \mathtt{sk}) &= \frac{m^{2^\ell}}{\Delta^{2^\ell-1}} + E_{\ell,1} \pmod{q_{\bar{\ell}}}.
\end{aligned}$$

Re-using the analysis from Section 3.2, we have that the error after multiplication without rescale is dominated by:

$$B = E_{\ell,0}\frac{m^{2^\ell}}{\Delta^{2^\ell-1}} + E_{\ell,1}\frac{m^{2^\ell}}{\Delta^{2^\ell-1}} + E_{\ell,0}E_{\ell,1}.$$

3.5 CKKS Error Estimation

The following formulas are taken from [15] and will be useful in our work.

3.5.1 Fresh ciphertext

The variance of the error of a fresh ciphertext with error distribution $\mathcal{N}(0, \sigma_e^2 I_n)$ and ternary secret distribution (over domain $\{-1, 0, 1\}$) of variance $2/3$ is approximated as

$$\rho^2_{fresh} = (\frac{4}{3}n + 1)\sigma^2.$$

3.5.2 Multiplication with rescale

Multiplication of two ciphertexts with rescale results in a ciphertext with error of the following form

$$B_{final\ error} = \Delta^{-1}(B_{mult} + B_{ks}) + B_{round}. \tag{2}$$

For the parameter sets we consider, B_{round} dominates the error, where B_{round} has variance

$$\rho^2_{mult\ error} = \frac{n}{18} + \frac{1}{12}.$$

4 Adversarial Model

Let us first examine the $\mathsf{IND\text{-}CPA}^D$ adversarial model introduced by Li and Micciancio [26]. In their setting, the adversary was a passive observer—as in the IND-CPA security game—but with the additional (limited) power to request decryptions of evaluations of honestly generated ciphertexts. For reasons of space, we present the definition in Appendix **??**.

Definition 4.1 ($\mathsf{IND\text{-}CPA}^D$ Security [26]). *Let $\mathcal{E} = (\mathtt{KeyGen}, \mathtt{Encrypt}, \mathtt{Decrypt}, \mathtt{Eval})$ be a public-key homomorphic, approximate encryption scheme with plaintext space $\mathcal{M}$ and ciphertext space $\mathcal{C}$. We define an experiment $\mathsf{Expr}_b^{\mathsf{indcpa}^D}[\mathcal{A}]$, parametrized by a bit $b \in \{0, 1\}$ and involving an efficient adversary $\mathcal{A}$ that is given access to the following oracles, sharing a common state $S \in (\mathcal{M} \times \mathcal{M} \times \mathcal{C})^*$ consisting of a sequence of message-message-ciphertext triplets:*

- *An encryption oracle $\mathtt{Encrypt}(\mathtt{pk}, m_0, m_1)$ that, given a pair of plaintext messages m_0, m_1, computes $\mathtt{ct} \leftarrow \mathtt{Encrypt}(\mathtt{pk}, m_b)$, extends the state*
$$S := [S; (m_0, m_1, \mathtt{ct})]$$
with one more triplet, and returns the ciphertext $\mathtt{ct}$ to the adversary.
- *An evaluation oracle $H_{\mathtt{evk}}(g, J)$ that, given a function $g : \mathcal{M}^k \to \mathcal{M}$ and a sequence of indices $J = (j_1, \ldots, j_k) \in \{1, \ldots, |S|\}^k$, computes the ciphertext $\mathtt{ct} \leftarrow \mathtt{Eval}(\mathtt{evk}, g, S[j_1].\mathtt{ct}, \ldots, S[j_k].\mathtt{ct})$, extends the state*
$$S := [S; (g(S[j_1].m_0, \ldots, S[j_k].m_1), g(S[j_1].m_1, \ldots, S[j_k].m_1)), \mathtt{ct}]$$
with one more triplet and returns the ciphertext $\mathtt{ct}$ to the adversary.
- *A decryption oracle $\mathtt{Decrypt}(\mathtt{sk}, j)$ that, given an index $j \leq |S|$, checks whether $S[j].m_0 = S[j].m_1$, and, if so, returns $\mathtt{Decrypt}(\mathtt{sk}, S[j].\mathtt{ct})$ to the adversary.*

The experiment is defined as

$$\begin{aligned}\mathsf{Expr}_b^{\mathsf{indcpa}^D}[\mathcal{A}](1^\kappa) :& (\mathtt{sk}, \mathtt{pk}, \mathtt{evk}) \leftarrow \mathtt{KeyGen}(1^\kappa)\\ & S := [\,]\\ & b' \leftarrow \mathcal{A}^{\mathtt{Encrypt}(\mathtt{pk},\cdot,\cdot), H(\mathtt{evk},\cdot,\cdot), \mathtt{Decrypt}(\mathtt{sk},\cdot)}(1^\kappa, \mathtt{pk}, \mathtt{evk})\\ & \mathsf{return}(b')\end{aligned}$$

The advantage of adversary $\mathcal{A}$ against the $\mathsf{IND\text{-}CPA}^D$ security of the scheme is

$$\mathsf{Adv}_{\mathsf{indcpa}^D}[\mathcal{A}](\kappa) = \left| \Pr[\mathsf{Expr}_0^{\mathsf{indcpa}^D}[\mathcal{A}](1^\kappa) = 1] - \Pr[\mathsf{Expr}_1^{\mathsf{indcpa}^D}[\mathcal{A}](1^\kappa) = 1] \right|.$$

In this work, we consider two adversarial models that are variants and/or special cases of the $\mathsf{IND\text{-}CPA}^D$ model presented in Definition 4.1.

The first adversarial model. We introduce a "white-box" variant of the encryption oracle, denoted $\texttt{Encrypt}^*(\texttt{pk}, \cdot, \cdot)$. When queried with two messages $m_0 = m_1$, this oracle returns a ciphertext $\texttt{ct}$, as well as the internal randomness v, e_0, and e_1 generated during the encryption process (see the definition of the encryption function in Section 3). If $m_0 \neq m_1$, it returns the ciphertext *only*. Other than this change, the adversarial model can be viewed as a special case of Li and Micciancio's $\mathsf{IND\text{-}CPA}^D$ in which the $\texttt{Encrypt}^*$ oracle is only called with $m_0 = m_1$ and the $\texttt{Eval}$ oracle is called on the identity function only. Thus, the set S consists only of fresh ciphertexts $\texttt{ct}$, and only those for which $m_0 = m_1$ may be queried to the decryption oracle. The decryption oracle we consider returns $\texttt{Decrypt}(\texttt{sk}, \texttt{ct}) + \mathcal{N}(0, \sigma_\epsilon^2)$, for some noise-flooding variance σ_ϵ^2. The goal of our adversary will be full key recovery, at which point it can trivially break the $\mathsf{IND\text{-}CPA}^D$ security by performing an encryption query with $m_0 \neq m_1$, obtaining $\texttt{ct}$, and using the recovered key to decrypt and find the value of b. In Section 4.1, we formalize the information the adversary observes as "hints" for this adversarial model.

The second adversarial model. This model is a special case of the $\mathsf{IND\text{-}CPA}^D$ model. There is no "white-box" encryption oracle and the adversary is a legal $\mathsf{IND\text{-}CPA}^D$ adversary. The $\texttt{Encrypt}(\texttt{pk}, \cdot, \cdot)$ oracle is only called with $m_0 = m_1$ and the $\texttt{H}_{\texttt{evk}}(\cdot, \cdot)$ oracle is only called with functions $g : \mathcal{M}^k \to \mathcal{M}$ in Class 1 or Class 2 and with input indices $J = (j_1, \ldots, j_k)$ that correspond to k distinct, fresh ciphertexts outputted by calls to $\texttt{Encrypt}(\texttt{pk}, m_0, m_1)$ with $m_0 = m_1$ and have not been included in a set J in a previous call to $\texttt{H}_{\texttt{evk}}(\cdot, \cdot)$. Decryption queries are only made with ciphertexts $\texttt{ct}$ corresponding to the output of calls to $\texttt{H}_{\texttt{evk}}(\cdot, \cdot)$ as described above. The decryption oracle we consider returns $\texttt{Decrypt}(\texttt{sk}, \texttt{ct}) + \mathcal{N}(0, \sigma_\epsilon^2)$, for some noise-flooding variance σ_ϵ^2. As before, the goal of our adversarial model is full key recovery, at which point it can trivially break the $\mathsf{IND\text{-}CPA}^D$ security by performing one more encryption query with $m_0 \neq m_1$. In Section 8, we extend our analysis from Section 4.1 to capture the "hints" obtained in this adversarial model.

4.1 Modeling Noisy Decryptions of Identity Circuit as Hints

We concretely consider an adversary who obtains t independently sampled encryptions and then asks for t decryptions of the constructed ciphertexts. Instantiating this attack with the CKKS + noise-flooding scheme, for each $j \in [t]$, the adversary obtains the (noisy) polynomial $\mathbf{e}_1^j \cdot \mathbf{s} + \mathbf{v}^j \cdot \mathbf{e} \approx \gamma^j$, where multiplication is over the ring R_q. The adversary knows $\mathbf{e}_1^j$ and $\mathbf{v}^j$ whose coordinates are modeled as independent Gaussians with 0 mean and variance $\sigma_{h_s}^2$ and $\sigma_{h_e}^2$, respectively. $(\mathbf{s}||\mathbf{e})$ corresponds to the LWE secret/error used to construct the public key. Since we assume that all the polynomials involved have small magnitude, there is actually no wraparound modulo q. In this case, we can view the multiplication and addition as over the ring of integers $\mathbb{Z}[x]/\Phi_m(x)$, where $\Phi_m(x)$ is the m-th cyclotomic polynomial of degree $n = \phi(m)$, and n is a power of two.

5 Security Loss under a Lattice Reduction Attack

The matrix $\mathbf{\Sigma}$ corresponds to the original covariance matrix for the LWE secret and error. Formally, let $\mathbf{\Sigma}$ be an $2n \times 2n$ diagonal matrix with the first n diagonal entries set to σ_s^2, the second n diagonal entries set to σ_e^2. The matrix $\mathbf{\Sigma}_\varepsilon$ corresponds to the covariance of the noise in the set of linear equations obtained on the LWE secret $\mathbf{s}$ from decrypting a ciphertext. Formally, $\mathbf{\Sigma}_\varepsilon = \sigma_\varepsilon^2 \cdot \mathbf{I}_{tn}$. $\gamma = \gamma^1 || \cdots || \gamma^t$ corresponds to the obtained outputs.

First, note that for $j \in [t]$,

$$\mathbf{e}_1^j \cdot \mathbf{s} = \mathbf{sVBP}\left(\mathbf{M}(\mathbf{e}_1^j)\right)\mathbf{P}^{-1}\mathbf{B}^{-1}\mathbf{V}^{-1},$$

where $\mathbf{V}$ is the canonical embedding transformation into $\mathbb{C}^n$, $\mathbf{B}$ is the matrix corresponding to the isomorphism between $H \subset \mathbb{C}^n$ and $\mathbb{R}^n$, $\mathbf{P}$ is a permutation matrix, and $\mathbf{A}_1^j := \mathbf{M}(\mathbf{e}_1^j)$ is a block diagonal matrix with $n/2$ blocks, each of dimension 2×2, where the i-th block is

$$\mathbf{A}_{1,i}^j := \begin{bmatrix} 1/\sqrt{2}w_{i,h_s}^j & 1/\sqrt{2}w_{n-i,h_s}^j \\ -1/\sqrt{2}w_{n-i,h_s}^j & 1/\sqrt{2}w_{i,h_s}^j, \end{bmatrix}$$

and $\mathbf{w}_{h_s}^j = (w_{1,h_s}^j, \ldots, w_{n,h_s}^j)$ is equal to $\mathbf{w}_{h_s}^j = \mathbf{e}_1^j\mathbf{VB}$. Since $\mathbf{VB}$ is an isometry (an orthogonal matrix scaled by $\sqrt{n}$), we have that $\sigma_{h_s}^2(\mathbf{VB})(\mathbf{VB})^T = n\sigma_{h_s}^2 \cdot \mathbf{I}_n$. So the random variables $[w_{i,h_s}^j, w_{n-i,h_s}^j]_{j\in[t], i\in[n/2]}$

are distributed as independent Gaussians with variance $n\sigma_{h_s}^2$. Note that $\mathbf{R} = (\mathbf{VBP})$ is a real matrix, even though $\mathbf{V}$ and $\mathbf{B}$ themselves are complex.

Similarly, for $j \in [t]$,

$$\mathbf{v}^j \cdot \mathbf{e} = \mathbf{eVBP}\left(\mathbf{M}(\mathbf{v}^j)\right)\mathbf{P}^{-1}\mathbf{B}^{-1}\mathbf{V}^{-1}.$$

In this case, $\mathbf{A}_2^j := \mathbf{M}(\mathbf{v}^j)$ is a block diagonal matrix with $n/2$ blocks, each of dimension 2×2, where the i-th block is

$$\mathbf{A}_{2,i}^j := \begin{bmatrix} 1/\sqrt{2}w_{i,h_e}^j & 1/\sqrt{2}w_{n-i,h_e}^j \\ -1/\sqrt{2}w_{n-i,h_e}^j & 1/\sqrt{2}w_{i,h_e}^j, \end{bmatrix}$$

and $\mathbf{w}_{h_e}^j = (w_{1,h_e}^j, \ldots, w_{n,h_e}^j)$ is equal to $\mathbf{w}_{h_e}^j = \mathbf{v}^j\mathbf{VB}$. Now for each $j \in [t]$, $i \in [n/2]$, w_{i,h_e}^j and w_{n-i,h_e}^j are random variables distributed as independent Gaussians with variance $n\sigma_{h_e}^2$.

Thus, if there are t decryption queries we can represent the hint matrix $\mathbf{H}$ as:

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & 0 \\ 0 & \mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^1 & \mathbf{A}_1^2 & \ldots & \mathbf{A}_1^t \\ \mathbf{A}_2^1 & \mathbf{A}_2^2 & \ldots & \mathbf{A}_2^t \end{bmatrix} \begin{bmatrix} \mathbf{R}^{-1} & & \\ & \ddots & \\ & & \mathbf{R}^{-1} \end{bmatrix},$$

where $\mathbf{R}$ is an orthogonal matrix scaled by $\sqrt{n}$.

Applying the approximate hints of [17], the transformed covariance matrix $\mathbf{\Sigma}'$ and mean μ' are as follows (the dimension and lattice of the DBDD instance remain unchanged):

$$\mathbf{\Sigma}' = \mathbf{\Sigma} - \mathbf{\Sigma H}(\mathbf{H}^T\mathbf{\Sigma H} + \mathbf{\Sigma}_\varepsilon)^{-1}\mathbf{H}^T\mathbf{\Sigma} \tag{3}$$

$$\mu' = \gamma(\mathbf{H}^T\mathbf{\Sigma H} + \mathbf{\Sigma}_\varepsilon)^{-1}\mathbf{H}^T\mathbf{\Sigma}. \tag{4}$$

Our goal is to find $\det(\mathbf{\Sigma}')$. Given this, we can estimate the hardness of the new DBDD instance under a lattice reduction attack. However, instead of computing $\mathbf{\Sigma}'$ and then $\det(\mathbf{\Sigma}')$ exactly, which requires inversion of a $2n \times 2n$ matrix, we will instead compute the expected value of $\det(\mathbf{\Sigma}')$, where the expectation is taken over the choice of the hint matrix $\mathbf{H}$.

Using a generalization of the Matrix Determinant Lemma, we obtain:

$$\mathbb{E}[\det(\mathbf{\Sigma}'^{\sim})] = \mathbb{E}\left[\frac{\det(\mathbf{H}^T\mathbf{\Sigma H} + \mathbf{\Sigma}_\varepsilon)}{\det(\mathbf{\Sigma}_\varepsilon)\det(\mathbf{\Sigma})}\right]. \tag{5}$$

Since $\mathbf{\Sigma}_\varepsilon$ and $\mathbf{\Sigma}$ are diagonal matrices whose entries depend on the parameters of the FHE cryptosystem, their determinants are constants and are easy to compute. Thus, it remains to compute $\mathbb{E}[\det(\mathbf{H}^T\mathbf{\Sigma H} + \mathbf{\Sigma}_\varepsilon)]$, which can then be plugged into (5).

Lemma 5.1. *Let $\mathbf{H}, \mathbf{R}, [\mathbf{A}_1^j = \mathbf{M}(\mathbf{e}_1^j), \mathbf{A}_2^j = \mathbf{M}(\mathbf{v}^j)]_{j\in[t]}$ be as described above. Then*

$$\mathbb{E}[\det(\mathbf{H}^T\mathbf{\Sigma H} + \mathbf{\Sigma}_\epsilon)] =$$

$$\left(\sigma_s^4\sigma_e^4\sigma_\epsilon^{4t-8}\Big(\frac{7}{4}t(t-1)n^4\sigma_{h_s}^4\sigma_{h_e}^4 + tn^2\sigma_\epsilon^4(\frac{\sigma_{h_s}^4}{\sigma_e^4} + \frac{\sigma_{h_e}^4}{\sigma_s^4}) + \Big(t(t-1)n^2\sigma_{h_s}^2\sigma_{h_e}^2 + tn\sigma_\epsilon^2(\frac{\sigma_{h_s}^2}{\sigma_e^2} + \frac{\sigma_{h_e}^2}{\sigma_s^2}) + \frac{\sigma_\epsilon^4}{\sigma_s^2\sigma_e^2}\Big)^2\Big)\right)^{\frac{n}{2}},$$

where the expectation is taken over choice of $\mathbf{e}_1^j \sim \mathcal{N}(0, \sigma_{h_s}^2)^n$ and $\mathbf{v}^j \sim \mathcal{N}(0, \sigma_{h_e}^2)^n$ for all $j \in [t]$.

Proof. We use the fact that if $\mathbf{A}$ is an invertible n-by-n matrix and $\mathbf{U}, \mathbf{V}$ are n-by-m matrices, then

$$\det\left(\mathbf{A} + \mathbf{U}\mathbf{V}^\top\right) = \det\left(\mathbf{I_m} + \mathbf{V}^\top\mathbf{A}^{-1}\mathbf{U}\right)\det(\mathbf{A}),$$

and the definition of $\mathbf{H}$ and $\mathbf{\Sigma}$ to rewrite $\det\left(\mathbf{H}^T\mathbf{\Sigma}\mathbf{H}+\mathbf{\Sigma}_\epsilon\right)$ as

$$
\begin{aligned}
&\det\left(\mathbf{H}^T\mathbf{\Sigma}\mathbf{H}+\mathbf{\Sigma}_\epsilon\right)\\
&=\det\left(\mathbf{I}_{2n}+\frac{1}{\sigma_\epsilon^2}\mathbf{\Sigma}^{1/2}\mathbf{H}\mathbf{H}^T\mathbf{\Sigma}^{1/2}\right)\det(\mathbf{\Sigma}_\epsilon)\\
&=\det\left(\mathbf{I}_{2n}+\frac{1}{\sigma_\epsilon^2}\begin{bmatrix}\sigma_s^2\mathbf{B}_{1,1} & \sigma_s\sigma_e\mathbf{B}_{1,2}\\ \sigma_s\sigma_e\mathbf{B}_{2,1} & \sigma_e^2\mathbf{B}_{2,2}\end{bmatrix}\right)\det(\mathbf{\Sigma}_\epsilon)\\
&=\det\left(\begin{bmatrix}\frac{\sigma_s^2}{\sigma_\epsilon^2}\sum_{j=1}^t\mathbf{A}_1^j(\mathbf{A}_1^j)^T+\mathbf{I}_n & \frac{\sigma_s\sigma_e}{\sigma_\epsilon^2}\sum_{j=1}^t\mathbf{A}_1^j(\mathbf{A}_2^j)^T\\ \frac{\sigma_s\sigma_e}{\sigma_\epsilon^2}\sum_{j=1}^t\mathbf{A}_2^j(\mathbf{A}_1^j)^T & \frac{\sigma_e^2}{\sigma_\epsilon^2}\sum_{j=1}^t\mathbf{A}_2^j(\mathbf{A}_2^j)^T+\mathbf{I}_n\end{bmatrix}\right)\det(\mathbf{\Sigma}_\epsilon)\\
&=\det(\star)\det(\mathbf{\Sigma}_\epsilon),
\end{aligned}
$$

where $\mathbf{B}_{k,l}\coloneqq\mathbf{R}\left(\frac{1}{n}\sum_{j=1}^t\mathbf{A}_k^j(\mathbf{A}_l^j)^T\right)\mathbf{R}^T$. Exchanging two rows and two columns of $\star$ at a time, which does not change the determinant, we obtain

$$
\det(\star)\det(\mathbf{\Sigma}_\epsilon)=\det\begin{bmatrix}\mathbf{S}_1 & 0 & \dots & 0 & 0\\ 0 & \mathbf{S}_2 & \dots & 0 & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ 0 & 0 & \dots & \mathbf{S}_{\frac{n}{2}-1} & 0\\ 0 & 0 & \dots & 0 & \mathbf{S}_{\frac{n}{2}}\end{bmatrix}\det(\mathbf{\Sigma}_\epsilon)=\det(\circledast)
$$

where

$$
\det\mathbf{S}_i=\det\begin{bmatrix}a_i & 0 & c_i & -d_i\\ 0 & a_i & d_i & c_i\\ c_i & d_i & b_i & 0\\ -d_i & c_i & 0 & b_i\end{bmatrix}=\det\begin{bmatrix}A & B\\ C & D\end{bmatrix}=\det(AD-BC), \tag{6}
$$

$$
\det(AD-BC)=\det\left(\begin{bmatrix}a_ib_i & 0\\ 0 & a_ib_i\end{bmatrix}-\begin{bmatrix}c_i & -d_i\\ d_i & c_i\end{bmatrix}\begin{bmatrix}c_i & d_i\\ -d_i & c_i\end{bmatrix}\right)=(a_ib_i-c_i^2-d_i^2)^2,
$$

$$
\begin{aligned}
a_i&=\frac{\sigma_s^2}{2\sigma_\epsilon^2}\left(\sum_{j=1}^t\left((w_{i,h_s}^j)^2+(w_{n-i,h_s}^j)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_s^2}\right),\\
b_i&=\frac{\sigma_e^2}{2\sigma_\epsilon^2}\left(\sum_{j=1}^t\left((w_{i,h_e}^j)^2+(w_{n-i,h_e}^j)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_e^2}\right),\\
c_i&=\frac{\sigma_s\sigma_e}{2\sigma_\epsilon^2}\sum_{j=1}^t\left(w_{i,h_s}^jw_{i,h_e}^j+w_{n-i,h_s}^jw_{n-i,h_e}^j\right),\\
d_i&=\frac{\sigma_s\sigma_e}{2\sigma_\epsilon^2}\sum_{j=1}^t\left(w_{i,h_s}^jw_{n-i,h_e}^j-w_{i,h_e}^jw_{n-i,h_s}^j\right).
\end{aligned}
$$

Note that (6) holds because if the blocks A, B, C, D are square matrices of the same size and, for example, C and D commute (i.e., $CD = DC$), then it holds that

$$
\det\begin{pmatrix}A & B\\ C & D\end{pmatrix}=\det(AD-BC)
$$

We therefore have that

$$
\begin{aligned}
&\det(\circledast) = \det(\boldsymbol{\Sigma}_\epsilon)\prod_{i=1}^{n/2}\det(\mathbf{S}_i) \\
&= \sigma_\epsilon^{2tn}\prod_{i=1}^{n/2}\frac{\sigma_s^4\sigma_e^4}{16\sigma_\epsilon^8}\Bigg(\left(\sum_{j=1}^{t}\left((w_{i,h_s}^j)^2+(w_{n-i,h_s}^j)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_s^2}\right)\left(\sum_{j=1}^{t}\left((w_{i,h_e}^j)^2+(w_{n-i,h_e}^j)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_e^2}\right) \\
&-\left(\sum_{j=1}^{t}\left(w_{i,h_s}^j w_{i,h_e}^j + w_{n-i,h_s}^j w_{n-i,h_e}^j\right)\right)^2-\left(\sum_{j=1}^{t}\left(w_{i,h_s}^j w_{n-i,h_e}^j - w_{i,h_e}^j w_{n-i,h_s}^j\right)\right)^2\Bigg)^2 \\
&= \left(\frac{\sigma_s^4\sigma_e^4}{16\sigma_\epsilon^{8-4t}}\right)^{\frac{n}{2}}\prod_{i=1}^{\frac{n}{2}}\Bigg(\sum_{1\le j\ne k\le t}\Bigg(\left(w_{i,h_s}^j w_{i,h_e}^k\right)^2+\left(w_{i,h_s}^j w_{n-i,h_e}^k\right)^2+\left(w_{n-i,h_s}^j w_{i,h_e}^k\right)^2+\left(w_{n-i,h_s}^j w_{n-i,h_e}^k\right)^2 \\
&-\Big(w_{i,h_s}^j w_{i,h_e}^j w_{i,h_s}^k w_{i,h_e}^k + w_{n-i,h_s}^j w_{n-i,h_e}^j w_{n-i,h_s}^k w_{n-i,h_e}^k \\
&\quad + w_{i,h_s}^j w_{n-i,h_e}^j w_{i,h_s}^k w_{n-i,h_e}^k + w_{n-i,h_s}^j w_{i,h_e}^j w_{n-i,h_s}^k w_{n-i,h_e}^k\Big)\Bigg) \\
&+\frac{2\sigma_\epsilon^2}{\sigma_s^2}\sum_{j=1}^{t}\left(\left((w_{i,h_e}^j\right)^2+\left((w_{n-i,h_e}^j\right)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_e^2}\sum_{j=1}^{t}\left(\left(w_{i,h_s}^j\right)^2+\left(w_{n-i,h_s}^j\right)^2\right)+4\frac{\sigma_\epsilon^4}{\sigma_s^2\sigma_e^2}\Bigg)^2.
\end{aligned}
$$

Now, to analyze the expectation $\mathbb{E}[\det(\circledast)]$ of the above expression, we identify

$$
\begin{aligned}
Y_i := &\sum_{1\le j\ne k\le t}\Bigg(\left(w_{i,h_s}^j w_{i,h_e}^k\right)^2+\left(w_{i,h_s}^j w_{n-i,h_e}^k\right)^2+\left(w_{n-i,h_s}^j w_{i,h_e}^k\right)^2+\left(w_{n-i,h_s}^j w_{n-i,h_e}^k\right)^2 \\
&-\Big(w_{i,h_s}^j w_{i,h_e}^j w_{i,h_s}^k w_{i,h_e}^k + w_{n-i,h_s}^j w_{n-i,h_e}^j w_{n-i,h_s}^k w_{n-i,h_e}^k \\
&\quad + w_{i,h_s}^j w_{n-i,h_e}^j w_{i,h_s}^k w_{n-i,h_e}^k + w_{n-i,h_s}^j w_{i,h_e}^j w_{n-i,h_s}^k w_{n-i,h_e}^k\Big)\Bigg) \\
&+\frac{2\sigma_\epsilon^2}{\sigma_s^2}\sum_{j=1}^{t}\left(\left((w_{i,h_e}^j\right)^2+\left((w_{n-i,h_e}^j\right)^2\right)+\frac{2\sigma_\epsilon^2}{\sigma_e^2}\sum_{j=1}^{t}\left(\left((w_{i,h_s}^j\right)^2+\left((w_{n-i,h_s}^j\right)^2\right)+4\frac{\sigma_\epsilon^4}{\sigma_s^2\sigma_e^2}.
\end{aligned}
$$

Since $\{w_{i,h_s}^j\}_{j=1;i=1}^{t;n}$ and $\{w_{i,h_e}^j\}_{j=1;i=1}^{t;n}$ are mutually independent, the expectation of the product is the product of each expectation,

$$
\mathbb{E}[\det\left(\mathbf{H}^T\boldsymbol{\Sigma}\mathbf{H}+\boldsymbol{\Sigma}_\epsilon\right)] = \left(\frac{\sigma_s^4\sigma_e^4}{16\sigma_\epsilon^{8-4t}}\right)^{\frac{n}{2}}\prod_{i=1}^{\frac{n}{2}}\mathbb{E}Y^2 = \left(\frac{\sigma_s^4\sigma_e^4}{16\sigma_\epsilon^{8-4t}}\right)^{\frac{n}{2}}\prod_{i=1}^{\frac{n}{2}}\left(\mathrm{Var}Y+\mathbb{E}^2Y\right),
$$

where

$$
\mathrm{Var}Y = 28t(t-1)n^4\sigma_{h_s}^4\sigma_{h_e}^4 + 16tn^2\sigma_\epsilon^4\left(\frac{\sigma_{h_s}^4}{\sigma_e^4}+\frac{\sigma_{h_e}^4}{\sigma_s^4}\right),
$$

$$
\mathbb{E}^2Y = \left(4t(t-1)n^2\sigma_{h_s}^2\sigma_{h_e}^2 + 4tn\sigma_\epsilon^2\left(\frac{\sigma_{h_s}^2}{\sigma_e^2}+\frac{\sigma_{h_e}^2}{\sigma_s^2}\right)+4\frac{\sigma_\epsilon^4}{\sigma_s^2\sigma_e^2}\right)^2.
$$

Finally, we obtain that

$$\mathbb{E}[\det(\mathbf{H}^T\mathbf{\Sigma}\mathbf{H}+\mathbf{\Sigma}_\epsilon)] =$$

$$\Big(\sigma_s^4\sigma_e^4\sigma_\epsilon^{4t-8}\Big(\frac{7}{4}t(t-1)n^4\sigma_{h_s}^4\sigma_{h_e}^4+tn^2\sigma_\epsilon^4(\frac{\sigma_{h_s}^4}{\sigma_e^4}+\frac{\sigma_{h_e}^4}{\sigma_s^4})+\Big(t(t-1)n^2\sigma_{h_s}^2\sigma_{h_e}^2+tn\sigma_\epsilon^2(\frac{\sigma_{h_s}^2}{\sigma_e^2}+\frac{\sigma_{h_e}^2}{\sigma_s^2})+\frac{\sigma_\epsilon^4}{\sigma_s^2\sigma_e^2}\Big)^2\Big)\Big)^{\frac{n}{2}}.$$

■

Obtaining the final hardness estimates. One can perform homogenization/isotropization of the DBDD instance (as in [17]) to obtain a uSVP instance and then estimate the BKZ-β for that instance. However, as described in [17], one can obtain the BKZ-β estimates using only the dimension and volume of the lattice after homogenization/isotropization, and the lattice basis itself is not required. The lattice in our DBDD instance is a q_L-ary lattice and thus has log volume $n \cdot \ln(q_L)$. After homogenization/isotropization, the log volume of the lattice increases to $n\ln(q_L)+\ln(\det(\mathbf{\Sigma}'^{\sim}))/2$. Using (5) and Lemma 5.1, we use the expectation of $\det(\mathbf{\Sigma}'^{\sim})$ in the above formula. The dimension remains unchanged after integrating hints. Thus, this information is sufficient for obtaining BKZ-β estimates for the final uSVP instance.

6 Key Recovery via Guessing

When $\mathbf{\Sigma}'$ in (3) has sufficiently small variance, then instead of running a lattice reduction attack, another strategy is to simply guess coordinates of the LWE secret/error by rounding the mean μ' in (4) to the nearest integer. If n coordinates of these coordinates are guessed and all guesses are correct, then the entire LWE secret/error can be recovered by solving a linear system modulo q. To analyze the success of the above attack we begin with the following lemma:

Lemma 6.1. *Let $\mathbf{\Sigma}'$ be defined as in (3). Then $\mathbf{Tr}(\mathbf{\Sigma}') \leq \mathsf{T} = n \cdot \frac{\left(\frac{\sigma_s^2\cdot\sigma_e^2\cdot 2t\cdot n(\sigma_{h_s}^2+\sigma_{h_e}^2)}{2\cdot\sigma_\epsilon^2}+\sigma_s^2+\sigma_e^2\right)}{B} + \frac{3\sqrt{2n\cdot V}}{B}$ with probability at least $0.99 - 3n\cdot e^{-12.25}$ over choice of hint vectors, where*

$$\begin{aligned}
B &= \frac{\sigma_s^2\cdot\sigma_e^2\cdot(2t-7\sqrt{2t})^2\cdot n^2\sigma_{h_s}^2\cdot\sigma_{h_e}^2}{4\cdot\sigma_\epsilon^4} + \frac{\sigma_s^2\cdot(2t-7\sqrt{2t})\cdot n\sigma_{h_s}^2}{2\cdot\sigma_\epsilon^2} \\
&\quad + \frac{\sigma_e^2\cdot(2t-7\sqrt{2t})(n\sigma_{h_e}^2)}{2\cdot\sigma_\epsilon^2} + 1 - \frac{\sigma_s^2\cdot\sigma_e^2\cdot n^2(\sigma_{h_s}^2+\sigma_{h_e}^2)^2(3.5\sqrt{2t}+12.25)^2}{2\cdot\sigma_\epsilon^4} \\
V &= \frac{\sigma_s^4\cdot\sigma_e^4\cdot(\mathbb{E}[R_1^2]+\mathbb{E}[R_2^2])}{4\cdot\sigma_\epsilon^4} + 2\frac{\sigma_s^4\cdot\sigma_e^4\cdot\mathbb{E}[R_1]\cdot\mathbb{E}[R_2]}{4\cdot\sigma_\epsilon^4} \\
&\quad + 2\frac{(\sigma_s^4\cdot\sigma_e^2+\sigma_s^2\cdot\sigma_e^4)\cdot(\mathbb{E}[R_1]+\mathbb{E}[R_2])}{2\cdot\sigma_\epsilon^2} + (\sigma_s^2+\sigma_e^2)^2 \\
&\quad - \Big(\frac{\sigma_s^2\cdot\sigma_e^2\cdot\mathbb{E}[R_1]}{2\cdot\sigma_\epsilon^2} + \frac{\sigma_s^2\cdot\sigma_e^2\cdot\mathbb{E}[R_2]}{2\cdot\sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2\Big)^2 \\
\mathbb{E}[R_1] &= 2t\cdot n\sigma_{h_s}^2 \\
\mathbb{E}[R_2] &= 2t\cdot n\sigma_{h_e}^2 \\
\mathbb{E}[R_1^2] &= 4tn^2\sigma_{h_s}^4 + 4t^2n^2\sigma_{h_s}^4 \\
\mathbb{E}[R_2^2] &= 4tn^2\sigma_{h_e}^4 + 4t^2n^2\sigma_{h_e}^4
\end{aligned}$$

We note that up to parameter setting of $n = 32768$, the success probability in the above claim is at least 0.52.[5]

[5] For the parameter sets with $n = 131072$, we increase 7 to 7.5, 3.5 to 3.75, 12.25 to 14.0625, and increase the probability to $0.99 - 3n\cdot e^{-14} > 0.66$.

Proof. Recall that

$$\mathbf{\Sigma}' = \mathbf{\Sigma} - \mathbf{\Sigma}\mathbf{H}(\mathbf{H}^T\mathbf{\Sigma}\mathbf{H} + \mathbf{\Sigma}_\varepsilon)^{-1}\mathbf{H}^T\mathbf{\Sigma}.$$

The eigenvalues of $\mathbf{\Sigma}'$ consist of the set of $\alpha \in \mathbb{R}$ such that $\mathsf{det}(\mathbf{\Sigma}' - \alpha \cdot \mathbf{I}) = 0$. Equivalently, $\mathsf{det}((\mathbf{\Sigma} - \alpha \cdot \mathbf{I}) - \mathbf{\Sigma}\mathbf{H}(\mathbf{H}^T\mathbf{\Sigma}\mathbf{H} + \mathbf{\Sigma}_\varepsilon)^{-1}\mathbf{H}^T\mathbf{\Sigma}) = 0$.

Using the generalization of the matrix determinant lemma, this is the same as finding α such that $\mathsf{det}(\mathbf{\Sigma}_\varepsilon + \mathbf{H}^T(\mathbf{\Sigma} - \mathbf{\Sigma}(\mathbf{\Sigma}' - \alpha \cdot \mathbf{I})^{-1}\mathbf{\Sigma})\mathbf{H}) = 0$.

Let $\tilde{\mathbf{\Sigma}} = \mathbf{\Sigma} - \mathbf{\Sigma}(\mathbf{\Sigma}' - \alpha \cdot \mathbf{I})^{-1}$. Then we must find α such that $\mathsf{det}(\mathbf{\Sigma}_\varepsilon + \mathbf{H}^T\tilde{\mathbf{\Sigma}}\mathbf{\Sigma}\mathbf{H}) = 0$ Then $\tilde{\mathbf{\Sigma}}$ is a diagonal matrix with entries $\frac{-\sigma_s^2 \cdot \alpha}{\sigma_s^2 - \alpha}$ in the first n positions and entries $\frac{-\sigma_e^2 \cdot \alpha}{\sigma_e^2 - \alpha}$ in the last n positions. Using the analysis from the proof of Lemma 5.1, we have that

$$\mathsf{det}(\mathbf{\Sigma}_\varepsilon + \mathbf{H}^T\tilde{\mathbf{\Sigma}}\mathbf{\Sigma}\mathbf{H}) = \mathit{\Pi}_{i \in [n/2]} = \mathit{\Pi}_{i \in [n/2]}(a_i b_i - c_i^2 - d_i^2)^2, \tag{7}$$

where

$$\begin{aligned} a_i &= \frac{-\sigma_s^2 \cdot \alpha}{2(\sigma_s^2 - \alpha)\sigma_\epsilon^2} R_{1,i} + 1, \\ b_i &= \frac{-\sigma_e^2 \cdot \alpha}{2(\sigma_e^2 - \alpha)\sigma_\epsilon^2} R_{2,i} + 1, \\ c_i &= \frac{-\sigma_s \cdot \sigma_3 \cdot \alpha}{2\sqrt{\sigma_s^2 - \alpha} \cdot \sqrt{\sigma_e^2 - \alpha}\sigma_\epsilon^2} R_{3,i}, \\ d_i &= \frac{-\sigma_s \cdot \sigma_3 \cdot \alpha}{2\sqrt{\sigma_s^2 - \alpha} \cdot \sqrt{\sigma_e^2 - \alpha}\sigma_\epsilon^2} R_{4,i} \end{aligned}$$

and

$$\begin{aligned} R_{1,i} &= \sum_{j=1}^{t} (W_{i,h_s}^j)^2 + (W_{n-i,h_s}^j)^2 \\ R_{2,i} &= \sum_{j=1}^{t} (W_{i,h_e}^j)^2 + (W_{n-i,h_e}^j)^2 \\ R_{3,i} &= \sum_{j=1}^{t} W_{i,h_s}^j W_{i,h_e}^j + W_{n-i,h_s}^j W_{n-i,h_e}^j \\ R_{4,i} &= \sum_{j=1}^{t} W_{i,h_s}^j W_{n-i,h_e}^j - W_{i,h_e}^j W_{n-i,h_s}^j. \end{aligned}$$

So of the four eigenvalues $(\alpha_{4i+1}, \alpha_{4i+2}, \alpha_{4i+3}, \alpha_{4i+4})$ corresponding to the i-th block, we have that $\alpha_{4i+1} = \alpha_{4i+3}$, $\alpha_{4i+2} = \alpha_{4i+4}$. Further, we can solve for α_{4i+1} and α_{4i+2} by finding the roots of the quadratic equation $(a_i b_i - c_i^2 - d_i^2) = 0$. $\sum_{j \in [4]} \alpha_{4i+j}$ is then equal to the sum of those roots, $\frac{-4q_{b,i}}{2q_{a,i}} = \frac{-2q_{b,i}}{q_{a,i}}$, where

$$\begin{aligned} q_{a,i} &= \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{1,i} \cdot R_{2,i}}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot R_{1,i}}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_e^2 \cdot R_{2,i}}{2 \cdot \sigma_\epsilon^2} + 1 - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{3,i}^2}{4 \cdot \sigma_\epsilon^4} - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{4,i}^2}{4 \cdot \sigma_\epsilon^4} \\ q_{b,i} &= -\left(\frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{1,i}}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{2,i}}{2 \cdot \sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2 \right). \end{aligned}$$

Towards bounding $\mathbb{E}[\frac{-q_{b,i}}{q_{a,i}}]$, we first lower bound $q_{a,i}$. Using the fact that $XY = 1/4(X+Y)^2 - 1/4(X-Y)^2$, we can express $R_{3,i}$ as

$$R_{3,i} = \sum_{j=1}^{2t} 1/4(X_j')^2 + \sum_{j=1}^{2t} 1/4(X_j'')^2$$

where X'_j and X''_j are a Gaussian random variable with variance $n(\sigma_{h_s}^2 + \sigma_{h_e}^2)$, $X'_1, \ldots, X'_{2t}$ are independent and $X''_1, \ldots, X''_{2t}$ are independent. The probability that either $1/4 \sum_{j=1}^{2t} (X')_j^2 \notin \frac{2t \cdot n}{4}(\sigma_{h_s}^2 + \sigma_{h_e}^2) \pm \frac{2 \cdot n(\sigma_{h_s}^2 + \sigma_{h_e}^2)(3.5\sqrt{2t}+12.25)}{4}$ or $1/4 \sum_{j=1}^{2t} (X'')_j^2 \notin \frac{2t \cdot n}{4}(\sigma_{h_s}^2 + \sigma_{h_e}^2) \pm \frac{2 \cdot n(\sigma_{h_s}^2 + \sigma_{h_e}^2)(3.5\sqrt{2t}+12.25)}{4}$ is at most $2 \cdot e^{-12.25}$. Thus, $R_{3,i}$ and $R_{4,i}$ are both in $[-n(\sigma_{h_s}^2 + \sigma_{h_e}^2)(5\sqrt{2t}+25), n(\sigma_{h_s}^2 + \sigma_{h_e}^2)(3.5\sqrt{2t}+12.25)]$ with all but $4 \cdot e^{-12.25}$ probability. Further, $R_{3,i}^2$ (and similarly $R_{4,i}^2$) is at most $n^2(\sigma_{h_s}^2 + \sigma_{h_e}^2)^2(3.5\sqrt{2t}+12.25)^2$.

$R_{1,i}$ can be expressed as the sum of $2t$ squares of Gaussians with variance $n\sigma_{h_s}^2$. So $R_{1,i} \geq (2t - 7\sqrt{2t}) \cdot n\sigma_{h_s}^2$ with probability $1 - e^{-12.25}$. Similarly, $R_{2,i} \geq (2t - 7\sqrt{2t})(n\sigma_{h_e}^2)$ with probability $1 - e^{-12.25}$.

Thus, we have that with all but $1 - 6 \cdot e^{-12.25}$ probability,

$$\begin{aligned}
q_{a,i} &= \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{1,i} \cdot R_{2,i}}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot R_{1,i}}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_e^2 \cdot R_{2,i}}{2 \cdot \sigma_\epsilon^2} + 1 \\
&\quad - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{3,i}^2}{4 \cdot \sigma_\epsilon^4} - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{4,i}^2}{4 \cdot \sigma_\epsilon^4} \\
&\geq \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot (2t - 7\sqrt{2t})^2 \cdot n^2 \sigma_{h_s}^2 \cdot \sigma_{h_e}^2}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot (2t - 7\sqrt{2t}) \cdot n\sigma_{h_s}^2}{2 \cdot \sigma_\epsilon^2} \\
&\quad + \frac{\sigma_e^2 \cdot (2t - 7\sqrt{2t})(n\sigma_{h_e}^2)}{2 \cdot \sigma_\epsilon^2} + 1 - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot n^2(\sigma_{h_s}^2 + \sigma_{h_e}^2)^2(3.5\sqrt{2t}+12.25)^2}{2 \cdot \sigma_\epsilon^4} \\
&= B.
\end{aligned}$$

Further, the above is true for all $i \in [n/2]$ with probability at least $1 - 3n \cdot e^{-12.25}$. So we have that

$$\begin{aligned}
\mathbb{E}\left[\frac{-2q_{b,i}}{q_{a,i}}\right] &\leq \frac{\mathbb{E}[-2q_{b,i}]}{B} \\
&= 2\frac{\left(\frac{\sigma_s^2 \cdot \sigma_e^2 \cdot \mathbb{E}[R_1]}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot \mathbb{E}[R_2]}{2 \cdot \sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2\right)}{B} \\
&= 2\frac{\left(\frac{\sigma_s^2 \cdot \sigma_e^2 \cdot 2t \cdot n\sigma_{h_s}^2}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot 2t \cdot n\sigma_{h_e}^2}{2 \cdot \sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2\right)}{B}
\end{aligned}$$

We next bound the variance of $-q_{b,i}$ (where the sum of $\sum_{j \in [4]} \alpha_{4i+j} = -4q_{b,i}$). Note that $\mathbb{E}[R_{1,i}]$ (resp. $\mathbb{E}[R_{2,i}]$, $\mathbb{E}[R_{2,i}]$, $\mathbb{E}[R_{1,i}^2]$, $\mathbb{E}[R_{2,i}^2]$) is the same for all $i \in [n/2]$. Therefore, we denote $\mathbb{E}[R_1] = \mathbb{E}[R_{1,i}]$ (resp. $\mathbb{E}[R_2] = \mathbb{E}[R_{2,i}]$, $\mathbb{E}[R_2] = \mathbb{E}[R_{2,i}]$, $\mathbb{E}[R_1^2] = \mathbb{E}[R_{1,i}^2]$, $\mathbb{E}[R_2^2] = \mathbb{E}[R_{2,i}^2]$). We have:

$$\begin{aligned}
\mathbb{E}[R_1] &= 2t \cdot n\sigma_{h_s}^2 \\
\mathbb{E}[R_2] &= 2t \cdot n\sigma_{h_e}^2 \\
\mathbb{E}[R_1^2] &= 4tn^2\sigma_{h_s}^4 + 4t^2n^2\sigma_{h_s}^4 \\
\mathbb{E}[R_2^2] &= 4tn^2\sigma_{h_e}^4 + 4t^2n^2\sigma_{h_e}^4
\end{aligned}$$

Further, R_1 and R_2 are independent.

$$\begin{aligned}
V &= \mathbb{E}[(q_b)^2] - \mathbb{E}[q_b]^2 \\
&= \frac{\sigma_s^4 \cdot \sigma_e^4 \cdot (\mathbb{E}[R_1^2] + \mathbb{E}[R_2^2])}{4 \cdot \sigma_\epsilon^4} + 2\frac{\sigma_s^4 \cdot \sigma_e^4 \cdot \mathbb{E}[R_1] \cdot \mathbb{E}[R_2]}{4 \cdot \sigma_\epsilon^4} \\
&\quad + 2\frac{(\sigma_s^4 \cdot \sigma_e^2 + \sigma_s^2 \cdot \sigma_e^4) \cdot (\mathbb{E}[R_1] + \mathbb{E}[R_2])}{2 \cdot \sigma_\epsilon^2} + (\sigma_s^2 + \sigma_e^2)^2 \\
&\quad - \Big(\frac{\sigma_s^2 \cdot \sigma_e^2 \cdot \mathbb{E}[R_1]}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot \mathbb{E}[R_2]}{2 \cdot \sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2\Big)^2
\end{aligned}$$

Using Chebyshev, we therefore have that $\Pr[|\mathbb{E}[\mathbf{Tr}(\mathbf{\Sigma}')] - \mathbf{Tr}(\mathbf{\Sigma}')| > 10\frac{\sqrt{2n \cdot V}}{B}] \leq 0.01$. Thus, putting everything together, we have that with $0.99 - 3n \cdot e^{-12.25}$ probability,

$$\mathbf{Tr}(\mathbf{\Sigma}') \leq n \cdot \frac{\left(\frac{\sigma_s^2 \cdot \sigma_e^2 \cdot 2t \cdot n(\sigma_{h_s}^2 + \sigma_{h_e}^2)}{2 \cdot \sigma_\epsilon^2} + \sigma_s^2 + \sigma_e^2\right)}{B} + \frac{10\sqrt{2n \cdot V}}{B}.$$

■

Given the above, we consider the distribution of $\mathbf{e}||\mathbf{s} - \mu'$, where μ' is the mean from equation (4). The random variable $\mathbf{e}||\mathbf{s} - \mu'$ is distributed as the multivariate Gaussian distribution $\mathcal{N}(0, \mathbf{\Sigma}')$. μ' is the correct guess for $\mathbf{e}||\mathbf{s}$ as long as for all $i \in [n]$ $|e_i - \mu_i'| \leq 0.5$ and for all $i \in [n]$ $|s_i - \mu_i'| \leq 0.5$. The probability that the above occurs for each coordinate is the same as the probability weight of the hypercube corresponding to $-0.5 \leq x_i \leq 0.5, i \in [n]$ under the multivariate Gaussian distribution $\mathcal{N}(0, \mathbf{\Sigma}')$. We use the following theorem to lower bound this probability weight:

Theorem 6.2 (Special case of the Gaussian Correlation Inequality [25]). *Let $\mathbf{X}$ be an n-dimensional Gaussian random variable. Then for any $t_1, \ldots, t_n > 0$,*

$$\mathbb{P}(|X_1| \leq t_1, \ldots, X_n \leq t_n] \geq \mathbb{P}(|X_1| \leq t_1) \cdots \mathbb{P}(|X_n| \leq t_n].$$

We instantiate the above theorem with $\mathbf{X}$ consisting of a subset S of size n of the coordinates of the conditional Gaussian distribution $((\mathbf{s}||\mathbf{e}) - \mu') \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}')$, with $t_j = 0.5, j \in S$ We thus have that

$$\mathbb{P}(|X_j| \leq t_j, i \in S) \geq \mathit{\Pi}_{j \in S} \mathbb{P}_{X_j \sim \mathcal{N}(0, \mathbf{e}_j \mathbf{\Sigma}' \mathbf{e}_j^T)}(|X_j| \leq t_j], \tag{8}$$

where the $\mathbf{e}_j$ are the standard basis vectors.

To analyze $\Pr_{X_j \sim \mathcal{N}(0, \mathbf{e}_j \cdot \mathbf{\Sigma}' \cdot \mathbf{e}_j^T)}[X_j \leq 0.5]$, we note that $\sum_{i \in [2n]} \mathbf{e}_i \cdot \mathbf{\Sigma}' \cdot \mathbf{e}_i^T = \mathsf{Tr}(\mathbf{\Sigma}')$. By Lemma 6.1, we have that $\mathsf{Tr}(\mathbf{\Sigma}') \leq \mathsf{T}$ with 53% probability. Thus, the indices j corresponding to the n smallest values among $\{\mathbf{e}_i \cdot \mathbf{\Sigma}' \cdot \mathbf{e}_i^T : i \in [2n]\}$ have sum at most $\frac{\mathsf{T}}{2}$, and average $\frac{\mathsf{T}}{2n}$, which we use in our estimates.[6] Let $S \subseteq [2n]$ of size n be this set of minimum values. For each $j \in S$,

$$\Pr_{X_j \sim \mathcal{N}(0, \mathbf{e}_j \cdot \mathbf{\Sigma}' \cdot \mathbf{e}_j^T)}[|X_j| \leq 0.5] \geq -\mathsf{erf}\left(\frac{-0.5}{\sqrt{2 \cdot \frac{\mathsf{T}}{2n}}}\right). \tag{9}$$

Finally, the attack is as follows: The adversary chooses to guess the values of $\mathbf{e}_j$ or $\mathbf{s}_j$ for these n smallest values (corresponding to the set S), and then use the LWE instance to solve for the remaining n variables. The probability that all of the adversary's guesses are correct is lower bounded by the probability weight on the hypercube corresponding to $|X_j| \leq 0.5, j \in I$ when X is drawn from the multivariate Gaussian distribution $X \sim \mathcal{N}(0, \mathbf{\Sigma}')$. Using (8) and (9), this is at most

$$\mathit{\Pi}_{j \in S} - \mathsf{erf}\left(\frac{-0.5}{\sqrt{2 \cdot \mathbf{e}_j \cdot \mathbf{\Sigma}' \cdot \mathbf{e}_j^T}}\right) \geq -\mathsf{erf}\left(\frac{-0.5}{\sqrt{2 \cdot \frac{\mathsf{T}}{2n}}}\right)^n = -\mathsf{erf}\left(\frac{-0.5}{\sqrt{\frac{\mathsf{T}}{n}}}\right)^n.$$

The final success probability of the attack is:[7]

$$-\mathsf{erf}\left(\frac{-0.5}{\sqrt{\frac{\mathsf{T}}{n}}}\right)^n - 3n \cdot e^{-12.25} - 0.01. \tag{10}$$

[6] A more rigorous but looser analysis can be achieved by upperbounding the largest of the n smallest values by $\frac{\mathsf{T}}{n}$.

[7] And for $n = 131072$, we replace $e^{-12.25}$ with e^{-14}.

7 Hybrid Guessing/Lattice-Reduction Attacks

Recall the structure of the eigenvalues of $\mathbf{\Sigma}'$: There are $[n/2]$ blocks and for each $i \in [n/2]$, the eigenvalues $(\alpha_{4i+1}, \alpha_{4i+2}, \alpha_{4i+3}, \alpha_{4i+4})$, where $\alpha_{4i+1} = \alpha_{4i+3}$, $\alpha_{4i+2} = \alpha_{4i+4}$. For each $i \in [n/2]$, we say that $\{\alpha_{4i+1}, \alpha_{4i+2}\}$ and $\{\alpha_{4i+3}, \alpha_{4i+4}\}$ are pairs. For each i, the adversary computes $\mathbf{e}_i \mathbf{\Sigma}' \mathbf{e}_i^T$ and guesses μ_i for the g minimum values where g is the maximum value such that

$$-\mathsf{erf}\left(\frac{-0.5}{\sqrt{\frac{\mathsf{T}}{n}}}\right)^g \geq p, \tag{11}$$

for some threshold p. These guesses are made and incorporated as perfect hints. After this process, the covariance matrix is a principal submatrix of $\mathbf{\Sigma}'$ of dimension $(2n-g) \times (2n-g)$, which we denote by $\mathbf{\Sigma}''$. We denote by $\mathsf{PSub}_{2n-g}(\mathbf{\Sigma}')$ the set of all principal submatrices of $\mathbf{\Sigma}'$ of dimension $2n-g$. Similarly, the lattice reduces dimension by g and its volume remains the same. The following lemma gives a bound on the determinant of $\mathbf{\Sigma}''$.

Lemma 7.1. *Let $g \in \{0, 1, \ldots, n\}$. Let $\mathbf{\Sigma}'$ be defined as in (3). Let $\mathbf{\Sigma}'' = \mathsf{argmax}_{\widetilde{\mathbf{\Sigma}} \in \mathsf{PSub}_{2n-g}(\mathbf{\Sigma}')} \mathsf{Tr}(\widetilde{\mathbf{\Sigma}})$. With probability $0.99 - 4n \cdot e^{-12.25}$ over choice of hint vectors,*[8]

$$\mathbf{Tr}(\mathbf{\Sigma}') \leq \mathsf{T} \quad \text{and} \quad \mathsf{det}(\mathbf{\Sigma}'') \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{\left(\frac{L}{U}\right)^g},$$

where T and B are defined as in Lemma 6.1, and

$$\begin{aligned}
L &= \frac{G + \sqrt{G^2 - 4 \cdot B \cdot \sigma_s^2 \cdot \sigma_e^2}}{2 \cdot B} \\
U &= \frac{\sigma_s^2 \cdot \sigma_e^2}{B_{max}} \\
G &= \sigma_s^2 \cdot \sigma_e^2 (2t + 7\sqrt{2t} + 24.5) \cdot (n\sigma_{h_e}^2) 2 \cdot \sigma_\epsilon^2 + \sigma_s^2 \cdot \sigma_e^2 (2t + 7\sqrt{2t} + 24.5) \cdot (n\sigma_{h_s}^2) 2 \cdot \sigma_\epsilon^2 + \sigma_s^2 + \sigma_e^2 \\
B_{max} &= \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot (2t + 7\sqrt{2t} + 24.5)^2 \cdot n^2 \sigma_{h_s}^2 \cdot \sigma_{h_e}^2}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot (2t + 7\sqrt{2t} + 24.5) \cdot n\sigma_{h_s}^2}{2 \cdot \sigma_\epsilon^2} \\
&\quad + \frac{\sigma_e^2 \cdot (2t + 7\sqrt{2t} + 24.5)(n\sigma_{h_e}^2)}{2 \cdot \sigma_\epsilon^2} + 1.
\end{aligned}$$

Proof. Lemma 6.1 showed that with probability $0.99 - 3n \cdot e^{-12.25}$ over choice of hint vectors, $\mathbf{Tr}(\mathbf{\Sigma}') \leq \mathsf{T}$.

Let $\alpha_1, \ldots, \alpha_g$ be the g minimum eigenvalues of $\mathbf{\Sigma}'$. Using the Eigenvalue Interlacing Theorem [23], we have that $\mathsf{det}(\mathbf{\Sigma}'') \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{\alpha_1 \cdots \alpha_g}$. We therefore need a lower bound on $\alpha_1 \cdots \alpha_g$.

We consider $\alpha'_1, \ldots, \alpha'_g$ such that for all $i \in [g]$, $\{\alpha_i, \alpha'_i\}$ are a pair. We show an upper bound U on $\alpha'_i \leq U$ for all $i \in [g]$. We further show a lower bound L on all $\alpha'_i \cdot \alpha_i \geq L$ for all $i \in [g]$ with all but $1 - n \cdot e^{-12.25}$ probability. Finally, this allows us to obtain a lower bound $\alpha_i \geq \frac{L}{\alpha'_i} \geq \frac{L}{U}$ for all $i \in [g]$. $\alpha_1 \cdots \alpha_g$ can then be lower bounded by $\left(\frac{L}{U}\right)^g$. which implies that

$$\mathsf{det}(\mathbf{\Sigma}'') \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{\left(\frac{L}{U}\right)^g}.$$

Specifically, assuming the bounds from the proof of Lemma 6.1, and assuming in addition the following upper bounds on $R_{1,i}, R_{2,i}$, which occurs with $1 - 2 \cdot e^{-12.25}$ probability,[9]

$$R_{1,i} \leq (2t + 7\sqrt{2t} + 24.5) \cdot n\sigma_{h_s}^2 \qquad R_{2,i} \leq (2t + 7\sqrt{2t} + 24.5) \cdot n\sigma_{h_e}^2, \tag{12}$$

[8] For the parameter sets with $n = 131072$, we increase 7 to 7.5, 24.5 to 28.125 and increase the probability to $0.99 - 4n \cdot e^{-14}$.

[9] For the parameter sets with $n = 131072$, we increase 7 to 7.5, 24.5 to 28.125 and increase the probability to $1 - 2 \cdot e^{-14}$.

we have that

$$\begin{aligned}
-q_{b_i} &\leq \sigma_s^2 \cdot \sigma_e^2(2t + 7\sqrt{2t} + 24.5) \cdot (n\sigma_{h_e}^2)2 \cdot \sigma_\epsilon^2 \\
&\quad + \sigma_s^2 \cdot \sigma_e^2(2t + 7\sqrt{2t} + 24.5) \cdot (n\sigma_{h_s}^2)2 \cdot \sigma_\epsilon^2 + \sigma_s^2 + \sigma_e^2 \\
&= G.
\end{aligned}$$

Thus,

$$\forall i \in [g], \alpha_i' = \frac{-q_{b,i} + \sqrt{q_{b,i}^2 - 4q_{a,i} \cdot q_{c,i}}}{2 \cdot q_{a,i}} \leq \frac{G + \sqrt{G^2 - 4 \cdot B \cdot \sigma_s^2 \cdot \sigma_e^2}}{2 \cdot B} = L.$$

Using the same upper bounds from (12) we also have the following upper bound on $q_{a,i}$:[10]

$$\begin{aligned}
q_{a_i} &= \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{1,i} \cdot R_{2,i}}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot R_{1,i}}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_e^2 \cdot R_{2,i}}{2 \cdot \sigma_\epsilon^2} + 1 - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{3,i}^2}{4 \cdot \sigma_\epsilon^4} - \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{4,i}^2}{4 \cdot \sigma_\epsilon^4} \\
&\leq \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot R_{1,i} \cdot R_{2,i}}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot R_{1,i}}{2 \cdot \sigma_\epsilon^2} + \frac{\sigma_e^2 \cdot R_{2,i}}{2 \cdot \sigma_\epsilon^2} + 1 \\
&\leq \frac{\sigma_s^2 \cdot \sigma_e^2 \cdot (2t + 7\sqrt{2t} + 24.5)^2 \cdot n^2\sigma_{h_s}^2 \cdot \sigma_{h_e}^2}{4 \cdot \sigma_\epsilon^4} + \frac{\sigma_s^2 \cdot (2t + 7\sqrt{2t} + 24.5) \cdot n\sigma_{h_s}^2}{2 \cdot \sigma_\epsilon^2} \qquad (13) \\
&\quad + \frac{\sigma_e^2 \cdot (2t + 7\sqrt{2t} + 24.5)(n\sigma_{h_e}^2)}{2 \cdot \sigma_\epsilon^2} + 1 \\
&= B_{max}. \qquad (14)
\end{aligned}$$

Thus,

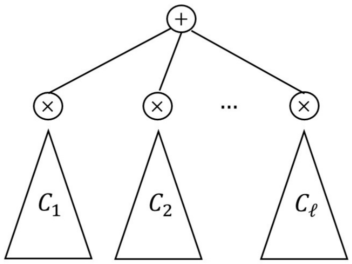
$$\begin{aligned}
\forall i \in [g], \alpha_i' \cdot \alpha_i &= \frac{-q_{b,i} + \sqrt{q_{b,i}^2 - 4q_{a,i} \cdot q_{c,i}}}{2 \cdot q_{a,i}} \cdot \frac{-q_{b,i} - \sqrt{q_{b,i}^2 - 4q_{a,i} \cdot q_{c,i}}}{2 \cdot q_{a,i}} \\
&= \frac{q_{c,i}}{q_{a,i}} \\
&\geq \frac{\sigma_s^2 \cdot \sigma_e^2}{B_{max}} = U.
\end{aligned}$$

■

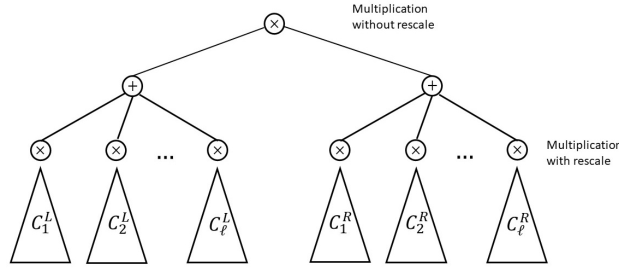
Combining Lemma 6.1 with Theorem 6.2 as before, we estimate that with at least $p - 4n \cdot e^{-12.25} - 0.01$ probability, all g number of guesses are correct, and

$$\mathsf{det}(\mathbf{\Sigma}'') \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{\left(\frac{L}{U}\right)^g}. \qquad (15)$$

We note that for up to $n = 32768$, $4n \cdot e^{-12.5} \leq 0.63$. [11] As before, $\mathbb{E}[\mathsf{det}(\mathbf{\Sigma}')]$ can be computed via Lemma 5.1. Thus, we can use (15) to obtain a bound on the expected value of $\mathsf{det}(\mathbf{\Sigma}'')$ (conditioned on events with probability at least $0.99 - 4n \cdot e^{-12.25}$ occurring), compute the log-volume of the lattice after homogenization/isotropization as described in Section 4.1, and use the log-volume and dimension to estimate the hardness of the residual instance (after guesses) under a lattice reduction attack.



(a) First Class of Circuits



(b) Second Class of Circuits

Fig. 1: A pictorial representation of the two classes of circuits we consider.

8 Extending to Larger Classes of Circuits

8.1 The first class of circuits and lattice reduction attacks

In Figure 1a we present the first class of circuits we consider. The circuits $C_1, \ldots, C_\ell$ that are depicted each consist of $\log(r)$ levels of multiplications as well as any number of additions. The final gate in each of the circuits $C_1, \ldots, C_\ell$ is a multiplication with rescale. Note that the noise after multiplication with rescale in circuit C_i is dominated by $\delta_1^i \cdot s + \delta_0^i$ (see Section 3.5.2), where δ_0^i, δ_1^i are distributed as uniform random variables in the range $[-0.5, 0.5]$.

The final gate of the entire circuit is an addition gate that adds the outputs of each of the C_i circuits. We require ℓ subcircuits and a final addition gate in order to ensure that the linear coefficients of the noise polynomial (which are independent and uniform random in the range in the range $[-0.5, 0.5]$ for each of the ℓ circuits) can be approximated by Gaussian random variables with mean 0 and variance $\frac{\ell}{12}$, which is the setting for which our Lemma 5.1 applies.

Specifically, the lattice reduction attack for circuits of this class can be analyzed by instantiating Lemma 5.1 with the following parameter settings.

- $\sigma^2_{h_s} = \frac{\ell}{12}$
- $\sigma^2_{h_e} = 0$
- σ^2_{ϵ} is set to the noise-flooding noise. The variance of the noise already present in the ciphertext can be computed by taking the noise in each ciphertext before addition (which Section 3.5.2 provides) and multiplying by ℓ.

8.2 The second class of circuits and lattice reduction attacks

In Figure 1b we present the second class of circuits we consider. The circuits $C_1^L, \ldots, C_\ell^L, C_1^R, \ldots, C_\ell^R$ that are depicted each consist of $\log(r)$ levels of multiplications as well as any number of additions. The final gate in each of the circuits $C_1^L, \ldots, C_\ell^L, C_1^R, \ldots, C_\ell^R$ is a multiplication with rescale. Note that the noise after multiplication with rescale in circuit C_i^L (resp. C_i^R) is dominated by $\delta_1^{L,i} \cdot s + \delta_0^{L,i}$ (resp. $\delta_1^{R,i} \cdot s + \delta_0^{R,i}$) (see Section 3.5.2), where $\delta_0^{L,i}, \delta_1^{L,i}$ (resp. $\delta_0^{R,i}, \delta_1^{R,i}$) are distributed as uniform random variables in the range $[-0.5, 0.5]$. Thus, after the summation gates on the second level from the top, the linear and constant coefficients of the noise corresponding to the left and right summations can be approximated by Gaussian random variables $G_{L,1}, G_{L,0}, G_{R,1}, G_{R,0}$ with mean 0 and variance $\frac{\ell}{12}$.

These outputs are then multiplied via a multiplication *without rescale* gate. For most parameter settings, the dominating terms of the error after the final multiplication without rescale will correspond to $\frac{m^r}{\Delta^{r-1}} \cdot (G_{L,1} + G_{R,1}) \cdot s$. Further, the dominating linear coefficients of s are again (well approximated by) a Gaussian of variance $\sigma^2_{h_s} = \frac{\ell}{6} \cdot (\frac{m^r}{\Delta^{r-1}})^2$. Since the error term does not include information about e, we can set $\sigma^2_{h_e} = 0$.

[10] For the parameter sets with $n = 131072$, we increase 7 to 7.5, and 24.5 to 28.

[11] And for $n = 131072$, $4n \cdot e^{-14} \leq 0.44$.

We compute the noise variance that is already present in the ciphertext, as a contribution of the following terms $\frac{m^r}{\Delta^{r-1}} \cdot (G_{L,0} + G_{R,0})$, $\frac{m^r}{\Delta^{r-1}} \cdot (G_{L,1} + G_{R,1}) \cdot s$, $(G_{L,1} \cdot G_{R,1}) \cdot s^2$, $(G_{L,0} \cdot G_{R,1}) \cdot s$, $(G_{L,1} \cdot G_{R,0}) \cdot s$, $G_{L,0} \cdot G_{R,0}$. Since the covariance of the above terms is 0, the total variance is the sum of the variances each term above. Recall that $G_{L,1}, G_{L,0}, G_{R,1}, G_{R,0}$ are Gaussian random variables with mean 0 and variance $\frac{\ell}{12}$.

Since the arithmetic is coordinate-wise on the canonical space, it suffices to consider the arithmetic of i-th component of each vector in $\mathbb{C}^{\mathbb{Z}_m^*}$. Specifically, we compute the variance of the real and imaginary coordinate of each vector. Since all pairs among all the resulting real and imaginary coordinates have covariance of 0, and since there is an isometry (orthogonal transformation scaled by $\frac{1}{\sqrt{n}}$) from the vector consisting of the real/imaginary parts of the canonical embedding multiplied by $\sqrt{2}$, we can obtain the coordinate-wise variance in the coefficient embedding by scaling the results we obtain by $\frac{2}{n}$.

Contribution of $\frac{m^r}{\Delta^{r-1}} \cdot (G_{L,0} + G_{R,0})$. The variance is immediately computed as $2(\frac{m^r}{\Delta^{r-1}})^2 \cdot \frac{\ell}{12} \cdot \sigma_s^2$.

Contribution of $\frac{m^r}{\Delta^{r-1}} \cdot G_{L,1} \cdot s$. Note that the contribution of $\frac{m^r}{\Delta^{r-1}} \cdot G_{R,1} \cdot s$ is the same as the above. By symmetry, we only need to compute the variance of the real part of the multiplication.

$$\begin{aligned}
&\mathrm{Var}\left[G_{L,1,i,\mathrm{Re}} s_{i,\mathrm{Re}} - G_{L,0,i,\mathrm{Im}} s_{i,\mathrm{Im}}\right] \\
&= 2 \cdot \frac{n}{2} \cdot \frac{\ell}{12} \cdot \frac{n\sigma_s^2}{2}
\end{aligned}$$

In the coefficient domain, the total contribution will be $(\frac{m^r}{\Delta^{r-1}})^2 \cdot \frac{n}{4} \cdot \frac{\ell}{12} \cdot \sigma_s^2$.

Contribution of $(G_{L,1} \cdot G_{R,1}) \cdot s^2$. By symmetry, we only need to compute the variance of the real part of the multiplication.

$$\begin{aligned}
&\mathrm{Var}\Big[(G_{L,1,i,\mathrm{Re}} G_{R,1,i,\mathrm{Re}} - G_{L,1,i,\mathrm{Im}} G_{R,1,i,\mathrm{Im}}) \left(s_{i,\mathrm{Re}}^2 - s_{i,\mathrm{Im}}^2\right) \\
&\qquad - 2 s_{i,\mathrm{Re}} s_{i,\mathrm{Im}} \left(G_{L,1,i,\mathrm{Re}} G_{R,1,i,\mathrm{Im}} + G_{L,1,i,\mathrm{Im}} G_{R,1,i,\mathrm{Re}}\right) \Big] \\
&= \mathrm{Var}[G_{L,1,i,\mathrm{Re}} G_{R,1,i,\mathrm{Re}} s_{i,\mathrm{Re}}^2 - G_{L,1,i,\mathrm{Re}} G_{R,1,i,\mathrm{Re}} s_{i,\mathrm{Im}}^2 - G_{L,1,i,\mathrm{Im}} G_{L,1,i,\mathrm{Im}} s_{i,\mathrm{Re}}^2 \\
&\quad + G_{L,1,i,\mathrm{Im}} G_{L,1,i,\mathrm{Im}} s_{i,\mathrm{Im}}^2 - 2 s_{i,\mathrm{Re}} s_{i,\mathrm{Im}} G_{L,1,i,\mathrm{Re}} G_{R,1,i,\mathrm{Im}} - 2 s_{i,\mathrm{Re}} s_{i,\mathrm{Im}} G_{L,1,i,\mathrm{Im}} G_{R,1,i,\mathrm{Re}}] \\
&= 4\mathbb{E}\left[G_{L,1,i,\mathrm{Re}}^2 (G_{R,1,i,\mathrm{Re}})^2 s_{i,\mathrm{Re}}^4\right] + 8\mathbb{E}\left[s_{i,\mathrm{Re}}^2 s_{i,\mathrm{Im}}^2 G_{L,1,i,\mathrm{Re}}^2 (G_{R,1,i,\mathrm{Im}})^2\right] \\
&= 4\left(\frac{n}{2} \cdot \frac{\ell}{12}\right)^2 \cdot \frac{3n^2}{4} \sigma_s^4 + 8\left(\frac{n}{2}\sigma_s^2\right)^2 \left(\frac{n}{2} \cdot \frac{\ell}{12}\right)^2 \\
&= \frac{5}{4} n^4 \left(\frac{\ell}{12}\right)^2 \sigma_s^4
\end{aligned}$$

In the coefficient domain, the contribution will be $\frac{5}{2} n^3 \left(\frac{\ell}{12}\right)^2 \sigma_s^4$.

Contribution of $(G_{L,0} \cdot G_{R,1}) \cdot s$. By symmetry, we only need to compute the variance of the real part of the multiplication.

$$\begin{aligned}
&\mathrm{Var}\Big[G_{L,0,i,\mathrm{Re}} G_{R,1,i,\mathrm{Re}} s_{i,\mathrm{Re}} - G_{L,0,i,\mathrm{Im}} G_{R,1,i,\mathrm{Im}} s_{i,\mathrm{Re}} \\
&\qquad - G_{L,0,i,\mathrm{Im}} G_{R,1,i,\mathrm{Re}} s_{i,\mathrm{Im}} - G_{L,0,i,\mathrm{Re}} G_{R,1,i,\mathrm{Im}} s_{i,\mathrm{Im}}\Big] \\
&= 4 \cdot \left(\frac{n}{2} \cdot \frac{\ell}{12}\right)^2 \cdot \frac{n\sigma_s^2}{2}
\end{aligned}$$

In the coefficient domain, the contribution will be $n^2 \left(\frac{\ell}{12}\right)^2 \sigma_s^2$.

Contribution of $G_{L,0} \cdot G_{R,0}$. By symmetry, we only need to compute the variance of the real part of the multiplication.

$$\begin{aligned}&\mathrm{Var}\left[G_{L,0,i,\mathrm{Re}} G_{R,0,i,\mathrm{Re}} - G_{L,0,i,\mathrm{Im}} G_{R,1,i,\mathrm{Im}}\right] \\ &= 2 \cdot \left(\frac{n}{2} \cdot \frac{\ell}{12}\right)^2\end{aligned}$$

In the coefficient domain, the contribution will be $n \cdot \left(\frac{\ell}{12}\right)^2$.

Total noise present. The total noise in the ciphertext has variance:

$$2\left(\frac{m^r}{\Delta^{r-1}}\right)^2 \cdot \frac{\ell}{12} \cdot \sigma_s^2 + \left(\frac{m^r}{\Delta^{r-1}}\right)^2 \cdot \frac{n \cdot \ell}{6} \cdot \sigma_s^2 + \frac{5}{2} n^3 \left(\frac{\ell}{12}\right)^2 \sigma_s^4 + 2n^2 \left(\frac{\ell}{12}\right)^2 \sigma_s^2 + n \cdot \left(\frac{\ell}{12}\right)^2$$

Obtaining the hardness estimates. We can now apply Lemma 5.1 with the following parameter settings:

- $\sigma_{h_s}^2 = \frac{\ell}{6} \cdot (\frac{m^r}{\Delta^{r-1}})^2$
- $\sigma_{h_e}^2 = 0$
- σ_ϵ^2 is set to the noise-flooding noise plus an additional $\frac{5}{2} n^3 \left(\frac{\ell}{12}\right)^2 \sigma_s^4 + 2n^2 \left(\frac{\ell}{12}\right)^2 \sigma_s^2$, the noise from the quadratic terms and the linear but non-Guassian terms (which comes from the terms of the form $(G_{L,0} \cdot G_{R,1}) \cdot s$).

Note that the noise-flooding noise has variance at least $(\frac{m^r}{\Delta^{r-1}})^2 \cdot \frac{n \cdot \ell}{6} \cdot \sigma_s^2$, since the noise already in the ciphertext is larger than this quantity. Thus, for $n \in \mathbb{N}$, when

$$(\frac{m^r}{\Delta^{r-1}})^2 \gg \frac{5}{2} n^2 \cdot \frac{\ell}{24} + 2 \cdot n \cdot \frac{\ell}{24} > \frac{9}{2} n^2 \cdot \frac{\ell}{24}, \tag{16}$$

and m achieves the maximum allowed magnitude B_{msg} of each coordinate in the *encoded* plaintext (in which the message is viewed as a vector in the canonical embedding and is scaled up by Δ), we have that the noise-flooding noise dominates the additional $\frac{5}{2} n^3 \left(\frac{\ell}{12}\right)^2 \sigma_s^4 + 2n^2 \left(\frac{\ell}{12}\right)^2 \sigma_s^2$. Typically, after encoding, the maximum allowed magnitude of m in the canonical embedding is $\approx \Delta$. Thus, (16) is satisfied when $\Delta \geq \frac{3n}{4} \cdot \sqrt{\frac{\ell}{3}}$, which is typically satisfied for most parameter settings (in fact, Δ is typically far larger).

Thus, we can plug the above parameter settings into Lemma 5.1 to obtain the hardness estimates for these circuits under a lattice reduction attack.

8.3 Guessing Attack for Class 1 and 2 Circuits

Now that we have determined $\sigma_{h_s}^2$, $\sigma_{h_e}^2$, and σ_ϵ^2 for Class 1 and Class 2 circuits, we can use those values to derive formulas for the concrete security for guessing and hybrid attacks as well.

Recall that for Class 1 and Class 2 circuits, the hints are only on the $\mathbf{s}$ coordinates. So $\mathbf{\Sigma}'$ is a block matrix where the lower right hand $n \times n$ submatrix is a diagonal matrix with diagonal $(\sigma_e^2, \ldots, \sigma_e^2)$ and the upper left hand $n \times n$ submatrix has n eigenvalues of the form $[(\alpha_{2i+1}, \alpha_{2i+2})]_{i \in [n/2]}$ and for all $i \in [n/2]$, $\alpha_{2i+1} = \alpha_{2i+2}$. Further, for each $i \in [n/2]$,

$$\alpha_{2i+1} = \frac{\sigma_s^2}{1 + \frac{\sigma_s^2 \cdot R_{1,i}}{2\sigma_\epsilon^2}}.$$

Since with all but e^{-11} probability[12], $R_{1,i} \geq (2t - 6.63\sqrt{2t}) \cdot n\sigma_{h_s}^2$, we have that with probability $1 - n/2 \cdot e^{-11}$ all eigenvalues are less than

$$\sigma_{max}^2 \leq \frac{\sigma_s^2}{1 + \frac{\sigma_s^2 \cdot (2t - 6.63\sqrt{2t}) \cdot n\sigma_{h_s}^2}{2\sigma_\epsilon^2}}, \tag{17}$$

[12] For the parameter sets with $n = 131072$, we increase 6.63 below to 7.2 and decrease the probability to e^{-13}.

and so for every standard basis vector $\mathbf{e}_i$, $\mathbf{e}_i \mathbf{\Sigma}'_S \mathbf{e}_i^T \leq \sigma_{max}^2$.

Finally, using the same techniques as above, this means that the guessing probability is at least

$$-\mathsf{erf}\left(\frac{-0.5}{\sqrt{2\sigma_{max}^2}}\right)^n. \tag{18}$$

Thus the total success probability of the attack is $-\mathsf{erf}\left(\frac{-0.5}{\sqrt{2\sigma_{max}^2}}\right)^n - n/2 \cdot e^{-11}$. We note that for up to parameter $n = 32768$, $n/2 \cdot e^{-11} \leq 0.28$.[13]

8.4 Hybrid Attack for Class 1 and 2 Circuits

Again, the attack for both Class 1 and Class 2 circuits is the same, with the only difference being the settings of $\sigma_{h_s}^2$, $\sigma_{h_e}^2$, and σ_ϵ^2 in the two cases.

The guessing strategy for the hybrid attack is as follows: For each i, the adversary computes $\mathbf{e}_i \mathbf{\Sigma}' \mathbf{e}_i^T$ and guesses μ_i for the g number of indices i with the minimum values of $\mathbf{e}_i \mathbf{\Sigma}' \mathbf{e}_i^T$, where g is the maximum value such that

$$-\mathsf{erf}\left(\frac{-0.5}{\sqrt{2\sigma_{max}^2}}\right)^g \geq p, \tag{19}$$

for some probability threshold p. These guesses are made and incorporated as perfect hints. After this process, the covariance matrix is a principal submatrix of $\mathbf{\Sigma}'_S$ of dimension $(n-g) \times (n-g)$, which we denote by $\mathbf{\Sigma}''_S$. Similarly, the lattice reduces dimension by g and its volume remains the same.

Let $\alpha_1, \ldots, \alpha_g$ be the g minimum eigenvalues of $\mathbf{\Sigma}'_S$. Using the Eigenvalue Interlacing Theorem [23], we have that $\mathsf{det}(\mathbf{\Sigma}''_S) \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{\alpha_1 \cdots \alpha_g}$. We therefore need a lower bound on $\alpha_1 \cdots \alpha_g$. Since with all but e^{-11} probability[14], $R_{1,i} \leq (2t + 6.63\sqrt{2t} + 22) \cdot n\sigma_{h_s}^2$, we have that with probability $1 - n/2 \cdot e^{-11}$ all eigenvalues are greater than

$$L = \frac{\sigma_s^2}{1 + \frac{\sigma_s^2 \cdot (2t + 6.63\sqrt{2t} + 22) \cdot n\sigma_{h_s}^2}{2\sigma_\epsilon^2}}. \tag{20}$$

Combining the above, we have that with at least $p - n \cdot e^{-11}$ probability, all g number of guesses are correct, and

$$\mathsf{det}(\mathbf{\Sigma}'') \leq \frac{\mathsf{det}(\mathbf{\Sigma}')}{L^g}. \tag{21}$$

We note that for the maximum setting of parameters $n = 32768$, $n \cdot e^{-11} \leq 0.55$.[15] Further, $\mathsf{det}(\mathbf{\Sigma}'')$ can be computed by plugging the parameter settings from Sections 8.1 and 8.2 into Lemma 5.1. Thus, we can use (21) to estimate the hardness of the residual instance (after guesses) under a lattice reduction attack.

9 Experiments

9.1 Experimental set-up

Parameter sets. We consider the parameter sets proposed by the homomorphicencryption.org standards [2], which were proposed with target security levels of 128, 192 or 256 bits. We update the target estimates using the concrete hardness given by the tool of [17].[16] This is presented in the column "Original Security" in all the tables below. An entry of x/y represents the original target security level x, and y represents the concrete (updated) security level. The standards only consider a ring dimension of up to $n = 32768$, i.e. $\log_2(n) = 15$, but some FHE applications may require a larger ring dimension, up to $\log_2(n) = 17$. We additionally provide estimates for the concrete security of CKKS for values of $\log_2(n) = 17$ by using the parameters given in [28].

[13] And for $n = 131072$, $n/2 \cdot e^{-13} \leq 0.15$.

[14] For the parameter sets with $n = 131072$, we increase 6.63 below to 7.2 and decrease the probability to e^{-13}.

[15] And for $n = 131072$, $n \cdot e^{-13} \leq 0.30$.

[16] Our analysis may give slightly different concrete hardness estimates than the LWE Estimator [4], since [17] takes into account the ellipsoidal distribution of the original secret/error.

Experimental validation. Before the experiments on the concrete security estimation of CKKS, we first provide experimental validation of Lemma 5.1, in Section 9.2. We also provide concrete security estimation for provably secure (statistical) noise-flooding, as presented in [27]. We provide these as a baseline, and to validate our methods. Since there is no reduction in security when applying statistical noise-flooding, the results of those experiments are presented in Appendix A.

Concrete security experiments set-up. Then, we consider the following experiments. We consider a lattice reduction attack, a guessing attack and a hybrid attack, as outlined in Sections 5, 6 and 7, respectively. We consider these on three types of circuit: the identity circuit, the class of circuits C1 and the class of circuits C2. Recall that these are described in Section 8.

Noise-flooding countermeasures. We use the results of [15] to estimate the output variance of the noise ρ^2_{circ}, where circ is one of Identity, C1 or C2. We then consider noise-flooding by ρ^2_{circ}, $100 \cdot \rho^2_{\mathsf{circ}}$ and $t \cdot \rho^2_{\mathsf{circ}}$, where t is the number of decryption queries. For guessing attacks, we do not include results for noise-flooding variance of $t \cdot \rho^2_{circ}$, since in this case, the guessing probability does not go above 10^{-200} for any parameter set. Similarly, for hybrid attacks, we do not include results for noise-flooding variance of $t \cdot \rho^2_{fresh}$, since no coordinates can be guessed with high confidence for any parameter set, and so the attack is equivalent to a lattice reduction attack [17]

9.2 Experimental Validation of Lemma 5.1

We first provide a verification of the theoretical results from Section 5, to demonstrate that the estimations hold in practice. In particular, Lemma 5.1 assumes that the distribution of the coefficients of $\mathbf{e}_1^j$ and $\mathbf{v}^j$ are independent Gaussians, while in practice this is not the case. The quantity of interest is $\det(\mathbf{\Sigma}'^{\sim})$, as defined in Section 5. In the proof of Lemma 5.1, we use the following fact:

$$\det(\mathbf{\Sigma}'^{\sim}) = \frac{\det(\mathbf{H}\mathbf{\Sigma}\mathbf{H}^T + \mathbf{\Sigma}_\varepsilon)}{\det(\mathbf{\Sigma}_\varepsilon)\det(\mathbf{\Sigma})} = \frac{\det\left(\mathbf{I}_{2n} + \frac{1}{\sigma_\epsilon^2}\mathbf{\Sigma}^{1/2}\mathbf{H}\mathbf{H}^T\mathbf{\Sigma}^{1/2}\right)}{\det(\mathbf{\Sigma})}. \tag{22}$$

In order to validate the canonical embedding transformation used in the analysis of Lemma 5.1, we sample a random hint matrix $\mathbf{H}$, directly compute $\mathbf{I}_{2n} + \mathbf{\Sigma}^{1/2}\mathbf{H}\mathbf{H}^T\mathbf{\Sigma}^{1/2}/\sigma_\epsilon^2$, and calculate its determinant. In order to construct the hint matrix, we sample $\mathbf{e}_1^j \leftarrow \chi$ and $\mathbf{v}^j \leftarrow S$ as defined in Appendix 3. We perform this experiment for various settings for the dimension of the LWE secret and error, and for various numbers of hints applied. For each parameter set, we perform 256 trials and take the average of the results in order to compare to the expected value predicted by Lemma 5.1. Figure 2 reports the experimental results, which very closely match the predictions. Notably, we see that the predictions become more accurate as the number of applied hints increases.

We perform this experiment using the SageMath library and run the calculations on an Intel Ice Lake XCC server. Calculating the determinant for larger parameter sets proves computationally infeasible with our experimental setup due to the extreme scaling, as each trial requires multiplying matrices of size $2n \times tn$ and $tn \times 2n$, as well as calculating the determinant of a matrix of size $2n \times 2n$, where n is the dimension and t is the number of hints. Additionally, in order to accurately calculate the final determinant, the numerical values within the matrix require increasingly high floating-point precision (e.g. hundreds or even thousands of bits), further slowing the computation. Our experiments take roughly a week to verify the largest parameter set in Figure 2 ($n = 256$, $t = 16$).

9.3 Concrete Security of Lattice Attacks on Identity Circuits

We begin by considering a lattice-reduction attack where the adversary may request any number of decryptions of fresh ciphertexts (i.e. evaluation of the identity circuit on a fresh ciphertext) with various noise-flooding levels. See Figure 3. To calculate the concrete hardness, we apply Lemma 5.1 to obtain the expected volume and dimension of the lattice after hints are integrated and homogenization/isotropization

[17] After ~ 200 million decryption queries, the estimated variance does not go below 3.6 for identity circuits, and after ~ 100 million decryption queries does not go below 0.33 and 0.36 for C1 and C2 circuits, respectively.

Dim	Num Hints	Predicted Determinant	Experimental Determinant
64	16	708.60	708.76
64	32	799.19	799.28
64	64	888.87	889.14
64	128	978.08	978.10
64	256	1067.04	1067.00
64	512	1155.89	1155.87
128	16	1594.55	1591.58
128	32	1175.78	1775.55
128	64	1955.17	1954.82
128	128	2133.59	2133.49
128	256	2311.52	2311.44
128	512	2489.22	2489.23
256	16	3543.88	3539.04

Fig. 2: **Summary of results for experimental validation of Lemma 5.1.** Each parameter set is specified by the dimension of the LWE secret and error (column 1) and the number of hints applied (column 2). The third column indicates the (natural log of) the expected value of the determinant as predicted by Lemma 5.1. The final column reports the determinant calculated by performing the experiment, as averaged over 256 trials.

is completed. As in [17], after homogenization/isotropization are performed, the hardness estimates for BKZ require only the volume and dimension of the lattice. These are reported in the final column.

9.4 Concrete Security of Guessing Attacks on Identity Circuits

We next consider a guessing-only attack, where the adversary may request any number of decryptions of fresh ciphertexts (i.e. evaluation of the identity circuit on a fresh ciphertext) with various noise-flooding levels. See Figure 4. In this attack, the adversary requests enough decryptions so that n LWE secret/error coordinates can be guessed correctly with high probability. Once these coordinates are guessed correctly, the LWE system of equations has a unique solution which can be recovered efficiently using Gaussian elimination. To determine the number of decryptions required to recover the LWE secret/error with some threshold probability, we apply Lemma 6.1 and (10).

9.5 Concrete Security of Hybrid Attacks on Identity Circuits

We next consider a hybrid attack, where the adversary may request some number of decryptions of fresh ciphertexts (i.e. evaluation of the identity circuit on a fresh ciphertext) with various noise-flooding levels. See Figure 5. The adversary requests enough decryptions so that some number of LWE secret/error coordinates can be guessed correctly with high probability. The adversary then integrates these guesses into its DBDD instance as perfect hints (as in [17]). Finally, the adversary performs homogenization/isotropizaton to obtain an SVP instance, and uses a BKZ solver to recover the LWE secret/error. For a fixed number of decryptions, we use (11) to determine the number of guesses g that can be made such that all guesses are correct with high probability. The dimension of the lattice reduces by g, and we compute the volume of the resulting lattice by applying (15). As in [17], after homogenization/isotropization are performed, the hardness estimates for BKZ require only the volume and dimension of the lattice. These are reported in the final column.

9.6 Concrete Security of Lattice Attacks on Class 1 and 2 Circuits

Class 1: This is the same attack as in Section 9.3, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 1 circuit (see Section 8.1 for the definition of this class) on fresh ciphertexts. To calculate the concrete hardness, we apply Lemma 5.1 to obtain the expected volume and dimension of the lattice after hints are integrated with the parameter settings for $\sigma^2_{h_s}, \sigma^2_{h_e}, \sigma^2_\epsilon$ given in Section 8.1 The results are reported in Figure 10 in Appendix A.

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
		ρ^2_{fresh}	1000	247 bikz $\approx$ 65 bits
$\log_2 n = 10, \log_2 q_L = 25$	128/102	$100 \cdot \rho^2_{\text{fresh}}$	1000	336 bikz $\approx$ 89 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	374 bikz $\approx$ 99 bits
		ρ^2_{fresh}	1000	366 bikz $\approx$ 97 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	$100 \cdot \rho^2_{\text{fresh}}$	1000	536 bikz $\approx$ 142 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	615 bikz $\approx$ 163 bits
		ρ^2_{fresh}	1000	461 bikz $\approx$ 122 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	$100 \cdot \rho^2_{\text{fresh}}$	1000	714 bikz $\approx$ 189 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	840 bikz $\approx$ 223 bits
		ρ^2_{fresh}	1000	288 bikz $\approx$ 76 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	$100 \cdot \rho^2_{\text{fresh}}$	1000	340 bikz $\approx$ 90 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	359 bikz $\approx$ 95 bits
		ρ^2_{fresh}	1000	450 bikz $\approx$ 119 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	$100 \cdot \rho^2_{\text{fresh}}$	1000	557 bikz $\approx$ 148 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	599 bikz $\approx$ 159 bits
		ρ^2_{fresh}	1000	590 bikz $\approx$ 157 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	$100 \cdot \rho^2_{\text{fresh}}$	1000	761 bikz $\approx$ 201 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	831 bikz $\approx$ 220 bits
		ρ^2_{fresh}	1000	322 bikz $\approx$ 85 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	$100 \cdot \rho^2_{\text{fresh}}$	1000	352 bikz $\approx$ 93 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	362 bikz $\approx$ 96 bits
		ρ^2_{fresh}	1000	517 bikz $\approx$ 137 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	$100 \cdot \rho^2_{\text{fresh}}$	1000	580 bikz $\approx$ 154 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	602 bikz $\approx$ 160 bits
		ρ^2_{fresh}	1000	701 bikz $\approx$ 186 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	$100 \cdot \rho^2_{\text{fresh}}$	1000	807 bikz $\approx$ 214 bits
		$t \cdot \rho^2_{\text{fresh}}$	1000	845 bikz $\approx$ 224 bits

Fig. 3: **Concrete security of lattice reduction attacks after observing decryptions of fresh ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption. ρ^2_{fresh} is the variance of the noise that is already present in a fresh ciphertext (see Section 3.5.1). The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Class 2: This is the same attack as in Section 9.3, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 2 circuit (see Section 8.2 for the definition of this class) on fresh ciphertexts. To calculate the concrete hardness, we apply Lemma 5.1 to obtain the expected volume and dimension of the lattice after hints are integrated with the parameter settings for $\sigma^2_{h_s}, \sigma^2_{h_e}, \sigma^2_{\epsilon}$ given in Section 8.2. The results are reported in Figure 11 in Appendix A.

9.7 Concrete Security of Guessing Attacks on Class 1 and 2 Circuits

Class 1: This is the same attack as in Section 9.4, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 1 circuit (see Section 8.1 for the definition of this class) on fresh ciphertexts. To determine the number of decryptions required to recover the LWE secret with high probability, we apply (18) with the settings of $\sigma_{h_s}, \sigma_{h_e}, \sigma^2_{\epsilon}$ given in Section 8.1. The results for various noise-flooding levels are reported in Figure 12 in Appendix A.

Class 2: This is the same attack as in Section 9.4, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 2 circuit (see Section 8.2 for the definition of this class) on

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
		ρ^2_{fresh}	1000	340 bikz $\approx$ 90 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	356 bikz $\approx$ 94 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	361 bikz $\approx$ 96 bits
		ρ^2_{fresh}	1000	553 bikz $\approx$ 146 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	587 bikz $\approx$ 156 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	598 bikz $\approx$ 159 bits
		ρ^2_{fresh}	1000	765 bikz $\approx$ 203 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	823 bikz $\approx$ 218 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	843 bikz $\approx$ 223 bits
		ρ^2_{fresh}	1000	341 bikz $\approx$ 90 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	349 bikz $\approx$ 93 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	352 bikz $\approx$ 93 bits
		ρ^2_{fresh}	1000	570 bikz $\approx$ 151 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/158	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	587 bikz $\approx$ 156 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	593 bikz $\approx$ 157 bits
		ρ^2_{fresh}	1000	796 bikz $\approx$ 211 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	$t \cdot \rho^2_{\mathsf{fresh}}$	1000	826 bikz $\approx$ 219 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	836 bikz $\approx$ 222 bits
		ρ^2_{fresh}	1000	343 bikz $\approx$ 91 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	347 bikz $\approx$ 92 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	348 bikz $\approx$ 92 bits
		ρ^2_{fresh}	1000	577 bikz $\approx$ 153 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	586 bikz $\approx$ 155 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	589 bikz $\approx$ 156 bits
		ρ^2_{fresh}	1000	810 bikz $\approx$ 215 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	826 bikz $\approx$ 219 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	831 bikz $\approx$ 220 bits
		ρ^2_{fresh}	1000	548 bikz $\approx$ 145 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	550 bikz $\approx$ 146 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	551 bikz $\approx$ 146 bits
		ρ^2_{fresh}	1000	703 bikz $\approx$ 186 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{fresh}}$	1000	706 bikz $\approx$ 187 bits
		$t \cdot \rho^2_{\mathsf{fresh}}$	1000	707 bikz $\approx$ 187 bits

Fig. 3: **Concrete security of lattice reduction attacks after observing decryptions of fresh ciphertexts, cont'd.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption. ρ^2_{fresh} is the variance of the noise that is already present in a fresh ciphertext (see Section 3.5.1). The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

fresh ciphertexts. To determine the number of decryptions required to recover the LWE secret with high probability, we apply (18) with the settings of $\sigma_{h_s}, \sigma_{h_e}, \sigma^2_\epsilon$ given in Section 8.2. The results for various noise-flooding levels are reported in Figure 13 in Appendix A.

9.8 Concrete Security of Hybrid Attacks on Class 1 and 2 Circuits

Class 1: This is the same attack as in Section 9.5, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 1 circuit (see Section 8.1 for the definition of this class) on fresh ciphertexts. For a fixed number of decryptions, we use (11), with the settings of $\sigma^2_{h_2}, \sigma^2_{h_e}$, and σ^2_ϵ given in Section 8.1, to determine the number of guesses g that can be made such that all guesses are correct with

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{fresh}	1160	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	62,180	0.80
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{fresh}	1160	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	62,180	0.80
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{fresh}	1160	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	62,180	0.80
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{fresh}	1220	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	67,950	0.80
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{fresh}	1220	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	67,950	0.80
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{fresh}	1220	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	67,950	0.80
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{fresh}	1290	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	73,760	0.80
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{fresh}	1290	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	73,760	0.80
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{fresh}	1290	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	73,760	0.80

Fig. 4: **Concrete security of guessing attacks after observing decryptions of fresh ciphertexts.** For each parameter set, the second column provides the target security and the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding variance added before returning the decryption. ρ^2_{fresh} is the noise variance already present in a fresh ciphertext. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in Lemma 6.1 occurring.

high probability. The dimension of the lattice reduces by g, and we compute the volume of the resulting lattice by applying (15), with the settings of $\sigma^2_{h_2}, \sigma^2_{h_e}$, and σ^2_{ϵ} given in Section 8.1. The results are reported in Figure 14 in Appendix A.

Class 2: This is the same attack as in Section 9.5, except the adversary requests decryptions of ciphertexts that correspond to the evaluation of a Class 2 circuit (see Section 8.2 for the definition of this class) on fresh ciphertexts. For a fixed number of decryptions, we use (11), with the settings of $\sigma^2_{h_2}, \sigma^2_{h_e}$, and σ^2_{ϵ} given in Section 8.2, to determine the number of guesses g that can be made such that all guesses are correct with high probability. The dimension of the lattice reduces by g, and we compute the volume of the resulting lattice by applying (15), with the settings of $\sigma^2_{h_2}, \sigma^2_{h_e}$, and σ^2_{ϵ} given in Section 8.2. The results are reported in Figure 15 in Appendix A.

10 Discussion of the results

Trends for noise-flooding level of ρ^2_{circ}. Our experimental data is summarized via the graphs in Figure 6. Figure 6(a) shows the reduction in bit security for a lattice reduction attack when considering an adversary who obtains 1000 decryptions of identity, Class 1, and Class 2 circuits with noise-flooding level ρ^2_{circ} equal to the noise that is already present in the ciphertext. We note that the graph exhibits a greater reduction in bit-security for identity circuits vs. Class 1 and 2 circuits. We believe the reason for this is that the hints in identity circuits involve all $2n$ coordinates in the LWE secret/error, so the variance of all $2n$ coordinates is reduced after each hint. On the other hand, hints in Class 1 and Class 2 circuits involve only the n coordinates from the LWE secret, so only the variance of these n coordinates is reduced after each hint. We also note that there is a greater security reduction for higher target security level vs. lower target security level. For example, for the lattice reduction attack, we see that for $\log_2(n) = 10$, identity circuits, and for

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{fresh}	1350	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	79,600	0.80
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{fresh}	1350	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	79,600	0.80
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{fresh}	1350	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	79,600	0.80
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{fresh}	1420	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	85,450	0.80
$\log_2 n = 14, \log_2 q_L = 284$	192/158	ρ^2_{fresh}	1420	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	85,450	0.80
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{fresh}	1420	0.81
		$100 \cdot \rho^2_{\mathsf{fresh}}$	85,450	0.80
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{fresh}	1480	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	91,320	0.80
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{fresh}	1480	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	91,320	0.80
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{fresh}	1480	0.80
		$100 \cdot \rho^2_{\mathsf{fresh}}$	91,320	0.80
		ρ^2_{fresh}	1690	0.81
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{fresh}}$	103,360	0.80
		ρ^2_{fresh}	1690	0.81
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{fresh}}$	103,360	0.82

Fig. 4: **Concrete security of guessing attacks after observing decryptions of fresh ciphertexts, cont'd.** For each parameter set, the second column provides the target security and the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding variance added before returning the decryption. ρ^2_{fresh} is the noise variance already present in a fresh ciphertext. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in Lemma 6.1 occurring.

a security level target of 192, the value of the bit security is reduced by slightly over 70 bits. On the other hand, for the same circuit and target security level and the same attack, for $\log_2(n) = 15$, the reduction in the bit security level is less than 5 bits. In fact, the reduction in security seems to be highly correlated with the setting of the modulus. When fixing the dimension n, target security level of 192 have smaller modulus q_L, compared to target security level of 128 and as the modulus q_L becomes smaller, "hints" obtained from decryption have more of an impact on the bit-security for lattice reduction attacks. The same trends can be seen in the Hybrid attack.

Figure 6(b) shows the number of queries required for guessing n coordinates with high probability for identity, Class 1 and Class 2 circuits. We note that guessing attacks perform significantly better for Class 1 and 2 circuits versus identity circuits. For identity circuits, there are a total of $2n$ eigenvalues that are reduced by obtaining hints, but n of these eigenvalues have relatively larger expectation, while n have smaller expectation (we believe this occurs because for identity circuits, hints correspond to linear combinations of both the s and e variables in the LWE instance, in which the s variables have variance $2/3$, while the e variables have variance 3.2^2). The eigenvectors corresponding to these eigenvalues do not align with the standard basis. Therefore, for purposes of fast estimates, we only take into account the trace (i.e. sum of the eigenvalues) and, given trace T, we argue that the average variance of the n secret or error coordinates with smallest variance is at most $\mathsf{T}/(2n)$. However, in practice, the n coordinates with the smallest variance may have variance significantly smaller than $\mathsf{T}/(2n)$. On the other hand, for Class 1 and 2 circuits, hints correspond to linear combinations of *only* the s variables from the LWE instance. Thus, we restrict our

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{fresh}	1130	793	0.80	62 bikz ≈ 17 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	621	0.80	48 bikz ≈ 13 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{fresh}	1130	793	0.80	109 bikz ≈ 29 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	621	0.80	160 bikz ≈ 42 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{fresh}	1130	793	0.80	160 bikz ≈ 42 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	621	0.80	222 bikz ≈ 59 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{fresh}	1130	801	0.80	154 bikz ≈ 41 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	623	0.80	184 bikz ≈ 49 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{fresh}	1130	801	0.80	261 bikz ≈ 69 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	623	0.80	306 bikz ≈ 81 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{fresh}	1130	801	0.80	358 bikz ≈ 95 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	623	0.80	415 bikz ≈ 110 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{fresh}	1130	807	0.80	240 bikz ≈ 63 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	624	0.80	260 bikz ≈ 69 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{fresh}	1130	807	0.80	395 bikz ≈ 105 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	624	0.80	427 bikz ≈ 113 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{fresh}	1130	807	0.80	544 bikz ≈ 144 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	624	0.80	587 bikz ≈ 156 bits

Fig. 5: **Concrete security of hybrid attacks after observing decryptions of fresh ciphertexts.** For each parameter set, the second column provides the target security and the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding variance added before returning the decryption. ρ^2_{fresh} is the noise variance that is already present in a fresh ciphertext. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret/error that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in Lemma 6.1 and (15) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

attention to a subspace with only n eigenvalues that are reduced by obtaining hints. All of these eigenvalues have the same distribution, and our proof shows that *all* the eigenvalues are less than maximum value σ^2_{max}.

Figure 6(c) shows the reduction in bit-security for a hybrid attack when considering an adversary who obtains decryptions of identity, Class 1, and Class 2 circuits. Figure 6(d) shows the number of queries obtained in each of these attacks. We chose the number of queries for the identity, Class 1, and Class 2 circuits so that a significant number of guesses can be made for each parameter set (otherwise the attack will be very similar to a lattice reduction attack). Based on the discussion above, this means that the number of queries required is far higher for identity circuits than Class 1 and Class 2 circuits. Thus, after guesses are made, the residual instance has lower variance in the case of identity circuits (since more hints have been incorporated, with each hint slightly reducing the variance). This explains why for approximately the same number of guesses, the reduction in bit-security is greater for identity circuits versus Class 1 and Class 2 circuits, as can be observed from the graph.

Trends across various noise-flooding levels. We first validate that there is no security drop in our experiments when using the statistically-secure noise-flooding levels proposed in [27]. Our results are presented in Figures 7, 8 and 9 in Appendix A. Indeed, we see in these tables that there is *no* reduction in either the security level or in the bikz for any parameter setting.

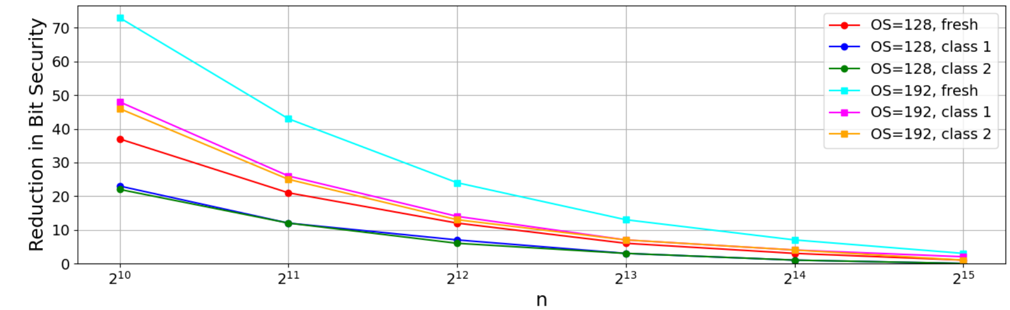
Recall that we investigate the effectiveness of noise-flooding levels ρ^2_{circ}, $100 \cdot \rho^2_{\mathsf{circ}}$, and $t \cdot \rho^2_{\mathsf{circ}}$, where t is the number of decryption queries, circ is one of Identity, C1 or C2, and ρ^2_{circ} is the noise variance present in the ciphertext. As expected, we see that the biggest drop in bit security is observed when noise-flooding by ρ^2_{circ}, across all parameter sets and across all circuits.

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{fresh}	1130	810	0.80	294 bikz ≈ 78 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	306 bikz ≈ 81 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{fresh}	1130	810	0.80	482 bikz ≈ 128 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	502 bikz ≈ 133 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{fresh}	1130	810	0.80	671 bikz ≈ 178 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	699 bikz ≈ 185 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{fresh}	1130	813	0.80	317 bikz ≈ 84 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	324 bikz ≈ 86 bits
$\log_2 n = 14, \log_2 q_L = 184$	192/158	ρ^2_{fresh}	1130	813	0.80	532 bikz ≈ 141 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	543 bikz ≈ 144 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{fresh}	1130	813	0.80	745 bikz ≈ 197 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	625	0.80	760 bikz ≈ 202 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{fresh}	1130	814	0.80	331 bikz ≈ 88 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	626	0.80	334 bikz ≈ 89 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{fresh}	1130	814	0.80	558 bikz ≈ 148 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	626	0.80	564 bikz ≈ 149 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{fresh}	1130	814	0.80	783 bikz ≈ 208 bits
		$100 \cdot \rho^2_{\mathsf{fresh}}$	58,000	626	0.80	792 bikz ≈ 210 bits
		ρ^2_{fresh}	1260	1557	0.80	539 bikz ≈ 143 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{fresh}}$	65,000	1411	0.80	540 bikz ≈ 143 bits
		ρ^2_{fresh}	1260	1557	0.80	692 bikz ≈ 183 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{fresh}}$	65,000	1411	0.80	691 bikz ≈ 183 bits

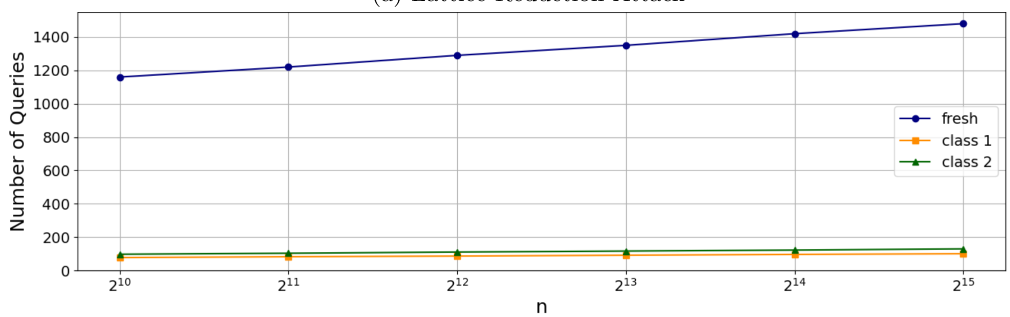
Fig. 5: **Concrete security of hybrid attacks after observing decryptions of fresh ciphertexts, cont'd.** For each parameter set, the second column provides the target security and the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding variance added before returning the decryption. ρ^2_{fresh} is the noise variance that is already present in a fresh ciphertext. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret/error that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in Lemma 6.1 and (15) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

In contrast, we observe that noise-flooding by $t \cdot \rho^2_{\mathsf{circ}}$ leads to a very low reduction in the security level, if at all. As opposed to a 70-bit security drop seen for lattice attacks with $\log_2(n) = 10$ and 192-bit security for identity circuits with noise level ρ^2_{fresh}, we see in Figure 3 that when noise-flooding by $t \cdot \rho^2_{\mathsf{fresh}}$, the security level drops by only a few bits. Further, as the value of $\log_2(n)$ (and thus also q_L increases), the security level drop decreases. We see for example in Figure 3 that for $\log_2(n) = 17$, there is no change in the security level.

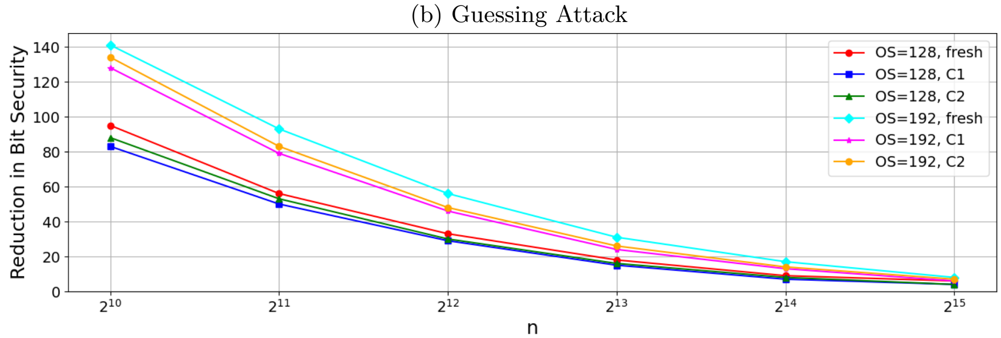
Conclusions: The first conclusion of this work is that, in practice, it is enough to noise flood by $t \cdot \rho^2_{\mathsf{circ}}$, where t is the number of decryption queries available to the adversary, and rho^2_{circ} is the variance of the noise, as predicted by an average-case noise analysis. Perhaps a less cautious approach is to noise flood by $\alpha \cdot t \cdot \rho$, where $0 < \alpha < 1$, if it is acceptable to have the security level drop by a few bits. We note that there is no definitive setting of α which is "best," and one can rather think of α as a parameter to be fine-tuned depending on the application. In particular, we note that one can think of increasing α as a way to allow for more decryption queries. Finally, we note that the techniques developed in this paper, as well as the experimental results presented, can be used as a way to establish key refreshing policies in a concrete application. Specifically, if the noise level is set to $\alpha \cdot t \cdot \rho$, the keys should be refreshed after releasing t number of decryptions. Thus, there can be a tradeoff among frequency of key refresh, an acceptable precision loss, and an acceptable drop in bit-security.



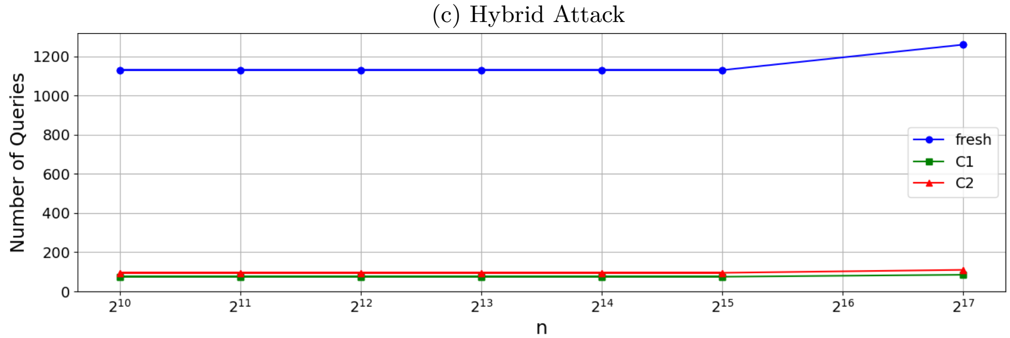
(a) Lattice Reduction Attack



(b) Guessing Attack



(c) Hybrid Attack



(d) Number of Queries in Hybrid Attack

Fig. 6: **Trends for the various attacks.** We compare the efficacy of lattice reduction, guessing, and hybrid attacks for various parameter sets, and for identity, Class 1, and Class 2 circuits with noise-flooding level equal to ρ^2_{fresh}, ρ^2_{C1}, and ρ^2_{C2}, respectively. (a) Shows the reduction in bit security for a lattice reduction attack against an adversary who obtains 1000 decryptions; (b) Shows the number of queries required for guessing n coordinates with probability at least 0.80. (c) Shows the reduction in bit security for a hybrid attack against an adversary who obtains a variable number of decryptions. The number of decryption queries for each parameter set is displayed in (d).

Acknowledgements

We thank Tikaram Sanyashi and Alexander Viand for helpful discussions and comments.

References

1. Ahmad Al Badawi, Jack Bates, Flavio Bergamaschi, David Bruce Cousins, Saroja Erabelli, Nicholas Genise, Shai Halevi, Hamish Hunt, Andrey Kim, Yongwoo Lee, et al. Openfhe: Open-source fully homomorphic encryption library. In *Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography*, pages 53–63, 2022.
2. Martin 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 standard. `homomorphicencryption.org`, 2018.
3. Martin R. Albrecht, Florian Göpfert, Fernando Virdia, and Thomas Wunderer. Revisiting the expected cost of solving uSVP and applications to LWE. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part I*, volume 10624 of *Lecture Notes in Computer Science*, pages 297–322, Hong Kong, China, December 3–7, 2017. Springer, Heidelberg, Germany.
4. Martin R. Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. Cryptology ePrint Archive, Report 2015/046, 2015. `https://eprint.iacr.org/2015/046`.
5. Andreea Alexandru, Ahmad Al Badawi, Daniele Micciancio, and Yuriy Polyakov. Application-aware approximate homomorphic encryption: Configuring fhe for practical use. Cryptology ePrint Archive, Paper 2024/203, 2024. `https://eprint.iacr.org/2024/203`.
6. Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange - A new hope. In Thorsten Holz and Stefan Savage, editors, *USENIX Security 2016: 25th USENIX Security Symposium*, pages 327–343, Austin, TX, USA, August 10–12, 2016. USENIX Association.
7. Private Communication, anonymized for submission.
8. Diego F. Aranha, Anamaria Costache, Antonio Guimarães, and Eduardo Soria-Vazquez. Heliopolis: Verifiable computation over homomorphically encrypted data from interactive oracle proofs is practical. Cryptology ePrint Archive, Paper 2023/1949, 2023. `https://eprint.iacr.org/2023/1949`.
9. Ahmad Al Badawi and Yuriy Polyakov. Demystifying bootstrapping in fully homomorphic encryption. Cryptology ePrint Archive, Report 2023/149, 2023. `https://eprint.iacr.org/2023/149`.
10. Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical GapSVP. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 868–886, Santa Barbara, CA, USA, August 19–23, 2012. Springer, Heidelberg, Germany.
11. Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. In Shafi Goldwasser, editor, *ITCS 2012: 3rd Innovations in Theoretical Computer Science*, pages 309–325, Cambridge, MA, USA, January 8–10, 2012. Association for Computing Machinery.
12. Jung Hee Cheon, Hyeongmin Choe, Alain Passelègue, Damien Stehlé, and Elias Suvanto. Attacks against the indcpa-d security of exact fhe schemes. Cryptology ePrint Archive, Paper 2024/127, 2024. `https://eprint.iacr.org/2024/127`.
13. Jung Hee Cheon, Andrey Kim, Miran Kim, and Yong Soo Song. Homomorphic encryption for arithmetic of approximate numbers. In Tsuyoshi Takagi and Thomas Peyrin, editors, *Advances in Cryptology – ASIACRYPT 2017, Part I*, volume 10624 of *Lecture Notes in Computer Science*, pages 409–437, Hong Kong, China, December 3–7, 2017. Springer, Heidelberg, Germany.
14. Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachène. TFHE: Fast fully homomorphic encryption over the torus. *Journal of Cryptology*, 33(1):34–91, January 2020.
15. 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, Report 2022/162, 2022. `https://eprint.iacr.org/2022/162`.
16. Anamaria Costache, Lea Nürnberger, and Rachel Player. Optimisations and tradeoffs for HElib. In Mike Rosulek, editor, *Topics in Cryptology – CT-RSA 2023*, volume 13871 of *Lecture Notes in Computer Science*, pages 29–53, San Francisco, CA, USA, April 24–27, 2023. Springer, Heidelberg, Germany.
17. Dana Dachman-Soled, Léo Ducas, Huijing Gong, and Mélissa Rossi. LWE with side information: Attacks and concrete security estimation. In Daniele Micciancio and Thomas Ristenpart, editors, *Advances in Cryptology – CRYPTO 2020, Part II*, volume 12171 of *Lecture Notes in Computer Science*, pages 329–358, Santa Barbara, CA, USA, August 17–21, 2020. Springer, Heidelberg, Germany.

18. Dana Dachman-Soled, Huijing Gong, Tom Hanson, and Hunter Kippen. Revisiting security estimation for LWE with hints from a geometric perspective. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023, Part V*, volume 14085 of *Lecture Notes in Computer Science*, pages 748–781, Santa Barbara, CA, USA, August 20–24, 2023. Springer, Heidelberg, Germany.
19. Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Report 2012/144, 2012. https://eprint.iacr.org/2012/144.
20. Sanjam Garg, Aarushi Goel, and Mingyuan Wang. How to prove statements obliviously? Cryptology ePrint Archive, Paper 2023/1609, 2023. https://eprint.iacr.org/2023/1609.
21. Rosario Gennaro, Craig Gentry, and Bryan Parno. Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 465–482, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.
22. Qian Guo, Denis Nabokov, Elias Suvanto, and Thomas Johansson. Key recovery attacks on approximate homomorphic encryption with non-worst-case noise flooding countermeasures. In *33rd USENIX Security Symposium (USENIX Security 24). Philadelphia, PA: USENIX Association*, 2024.
23. Suk-Geun Hwang. Cauchy's interlace theorem for eigenvalues of hermitian matrices. *American Mathematical Monthly*, pages 157–159, 2004.
24. Duhyeong Kim, Dongwon Lee, Jinyeong Seo, and Yongsoo Song. Toward practical lattice-based proof of knowledge from hint-MLWE. In Helena Handschuh and Anna Lysyanskaya, editors, *Advances in Cryptology – CRYPTO 2023, Part V*, volume 14085 of *Lecture Notes in Computer Science*, pages 549–580, Santa Barbara, CA, USA, August 20–24, 2023. Springer, Heidelberg, Germany.
25. Rafał Latała and Dariusz Matlak. Royen's proof of the gaussian correlation inequality. In *Geometric Aspects of Functional Analysis: Israel Seminar (GAFA) 2014–2016*, pages 265–275. Springer, 2017.
26. Baiyu Li and Daniele Micciancio. On the security of homomorphic encryption on approximate numbers. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021, Part I*, volume 12696 of *Lecture Notes in Computer Science*, pages 648–677, Zagreb, Croatia, October 17–21, 2021. Springer, Heidelberg, Germany.
27. Baiyu Li, Daniele Micciancio, Mark Schultz, and Jessica Sorrell. Securing approximate homomorphic encryption using differential privacy. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology – CRYPTO 2022, Part I*, volume 13507 of *Lecture Notes in Computer Science*, pages 560–589, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Heidelberg, Germany.
28. George Teseleanu. Subliminal hash channels. Cryptology ePrint Archive, Report 2019/1112, 2019. https://eprint.iacr.org/2019/1112.

A Additional Experimental Results

Parameter Set	Original Security	Noise Variance	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102.34 (386.21)	ρ^2_{stat}	386.21 bikz $\approx$ 102.34 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170.04 (641.65)	ρ^2_{stat}	641.65 bikz $\approx$ 170.04 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234.29 (884.13)	ρ^2_{stat}	884.13 bikz $\approx$ 234.29 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/96.84 (365.43)	ρ^2_{stat}	365.43 bikz $\approx$ 96.84 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162.31 (612.49)	ρ^2_{stat}	612.49 bikz $\approx$ 162.31 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226.11 (853.25)	ρ^2_{stat}	853.25 bikz $\approx$ 226.11 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/96.81 (365.34)	ρ^2_{stat}	365.34 bikz $\approx$ 96.81 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161.41 (609.11)	ρ^2_{stat}	609.11 bikz $\approx$ 161.41 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227.10 (856.98)	ρ^2_{stat}	856.98 bikz $\approx$ 227.10 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96.11 (362.66)	ρ^2_{stat}	362.66 bikz $\approx$ 96.11 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159.40 (601.49)	ρ^2_{stat}	601.49 bikz $\approx$ 159.40 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/224.89 (848.63)	ρ^2_{fresh}	848.63 bikz $\approx$ 224.89 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93.37 (352.34)	ρ^2_{stat}	352.34 bikz $\approx$ 93.37 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/157.62 (594.78)	ρ^2_{stat}	594.78 bikz $\approx$ 157.62 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222.42 (839.32)	ρ^2_{stat}	839.32 bikz $\approx$ 222.42 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92.37 (348.55)	ρ^2_{stat}	348.55 bikz $\approx$ 92.37 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156.35 (590.00)	ρ^2_{stat}	590.00 bikz $\approx$ 156.35 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220.52 (832.15)	ρ^2_{stat}	832.15 bikz $\approx$ 220.52 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/145.88 (550.51)	ρ^2_{stat}	550.51 bikz $\approx$ 145.88 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187.40 (707.17)	ρ^2_{stat}	707.17 bikz $\approx$ 187.40 bits

Fig. 7: **Concrete security of lattice reduction attacks after observing 1000 decryptions of fresh ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17] (bikz are provided in parenthesis). The third column indicates the noise-flooding noise added before returning the decryption to the adversary. $\rho^2_{\mathsf{stat}} = 12 \cdot t \cdot 2^{\kappa} \cdot \rho^2_{\mathsf{fresh}}$, where ρ^2_{fresh} is the variance of the noise that is already present in a fresh ciphertext (see Section 3.5.1), and $\kappa = 30$ is the statistical security parameter. The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Original Security	Noise Variance	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102.34 (386.21)	ρ^2_{stat}	386.21 bikz $\approx$ 102.34 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170.04 (641.65)	ρ^2_{stat}	641.65 bikz $\approx$ 170.04 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234.29 (884.13)	ρ^2_{stat}	884.13 bikz $\approx$ 234.29 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/96.84 (365.43)	ρ^2_{stat}	365.43 bikz $\approx$ 96.84 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162.31 (612.49)	ρ^2_{stat}	612.49 bikz $\approx$ 162.31 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226.11 (853.25)	ρ^2_{stat}	853.25 bikz $\approx$ 226.11 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/96.81 (365.34)	ρ^2_{stat}	365.34 bikz $\approx$ 96.81 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161.41 (609.11)	ρ^2_{stat}	609.11 bikz $\approx$ 161.41 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227.10 (856.98)	ρ^2_{stat}	856.98 bikz $\approx$ 227.10 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96.11 (362.66)	ρ^2_{stat}	362.66 bikz $\approx$ 96.11 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159.40 (601.49)	ρ^2_{stat}	601.49 bikz $\approx$ 159.40 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/224.89 (848.63)	ρ^2_{fresh}	848.63 bikz $\approx$ 224.89 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93.37 (352.34)	ρ^2_{stat}	352.34 bikz $\approx$ 93.37 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/157.62 (594.78)	ρ^2_{stat}	594.78 bikz $\approx$ 157.62 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222.42 (839.32)	ρ^2_{stat}	839.32 bikz $\approx$ 222.42 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92.37 (348.55)	ρ^2_{stat}	348.55 bikz $\approx$ 92.37 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156.35 (590.00)	ρ^2_{stat}	590.00 bikz $\approx$ 156.35 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220.52 (832.15)	ρ^2_{stat}	832.15 bikz $\approx$ 220.52 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/145.88 (550.51)	ρ^2_{stat}	550.51 bikz $\approx$ 145.88 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187.40 (707.17)	ρ^2_{stat}	707.17 bikz $\approx$ 187.40 bits

Fig. 8: **Concrete security of lattice reduction attacks after observing 1000 decryptions of Class 1 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17] (bikz are provided in parenthesis). The third column indicates the noise-flooding noise added before returning the decryption to the adversary. $\rho^2_{\mathsf{stat}} = 12 \cdot t \cdot 2^{\kappa} \cdot \rho^2_{\mathsf{C1}}$, where ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 1 circuit on fresh encryptions, and $\kappa = 30$ is the statistical security parameter. ρ^2_{fresh} is the variance of the noise that is already present in a fresh ciphertext (see Section 3.5.1). The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Original Security	Noise Variance	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102.34 (386.21)	ρ^2_{stat}	386.21 bikz $\approx$ 102.34 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170.04 (641.65)	ρ^2_{stat}	641.65 bikz $\approx$ 170.04 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234.29 (884.13)	ρ^2_{stat}	884.13 bikz $\approx$ 234.29 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/96.84 (365.43)	ρ^2_{stat}	365.43 bikz $\approx$ 96.84 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162.31 (612.49)	ρ^2_{stat}	612.49 bikz $\approx$ 162.31 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226.11 (853.25)	ρ^2_{stat}	853.25 bikz $\approx$ 226.11 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/96.81 (365.34)	ρ^2_{stat}	365.34 bikz $\approx$ 96.81 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161.41 (609.11)	ρ^2_{stat}	609.11 bikz $\approx$ 161.41 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227.10 (856.98)	ρ^2_{stat}	856.98 bikz $\approx$ 227.10 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96.11 (362.66)	ρ^2_{stat}	362.66 bikz $\approx$ 96.11 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159.40 (601.49)	ρ^2_{stat}	601.49 bikz $\approx$ 159.40 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/224.89 (848.63)	ρ^2_{fresh}	848.63 bikz $\approx$ 224.89 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93.37 (352.34)	ρ^2_{stat}	352.34 bikz $\approx$ 93.37 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/157.62 (594.78)	ρ^2_{stat}	594.78 bikz $\approx$ 157.62 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222.42 (839.32)	ρ^2_{stat}	839.32 bikz $\approx$ 222.42 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92.37 (348.55)	ρ^2_{stat}	348.55 bikz $\approx$ 92.37 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156.35 (590.00)	ρ^2_{stat}	590.00 bikz $\approx$ 156.35 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220.52 (832.15)	ρ^2_{stat}	832.15 bikz $\approx$ 220.52 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/145.88 (550.51)	ρ^2_{stat}	550.51 bikz $\approx$ 145.88 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187.40 (707.17)	ρ^2_{stat}	707.17 bikz $\approx$ 187.40 bits

Fig. 9: **Concrete security of lattice reduction attacks after observing 1000 decryptions of Class 2 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17] (bikz are provided in parenthesis). The third column indicates the noise-flooding noise added before returning the decryption to the adversary. $\rho^2_{\mathsf{stat}} = 12 \cdot t \cdot 2^\kappa \cdot \rho^2_{\mathsf{C2}}$, where ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions, and $\kappa = 30$ is the statistical security parameter. The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security. The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{C1}	1000	298 bikz $\approx$ 79 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	352 bikz $\approx$ 93 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	376 bikz $\approx$ 100 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{C1}	1000	460 bikz $\approx$ 122 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	568 bikz $\approx$ 150 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	619 bikz $\approx$ 164 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{C1}	1000	598 bikz $\approx$ 158 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	764 bikz $\approx$ 202 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	847 bikz $\approx$ 224 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{C1}	1000	319 bikz $\approx$ 85 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	348 bikz $\approx$ 92 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	360 bikz $\approx$ 95 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{C1}	1000	513 bikz $\approx$ 136 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	575 bikz $\approx$ 152 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	601 bikz $\approx$ 159 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{C1}	1000	689 bikz $\approx$ 183 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	790 bikz $\approx$ 209 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	834 bikz $\approx$ 221 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{C1}	1000	341 bikz $\approx$ 90 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	356 bikz $\approx$ 94 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	363 bikz $\approx$ 96 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{C1}	1000	555 bikz $\approx$ 147 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	589 bikz $\approx$ 156 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	603 bikz $\approx$ 160 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{C1}	1000	764 bikz $\approx$ 203 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	1000	823 bikz $\approx$ 218 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	847 bikz $\approx$ 224 bits

Fig. 10: **Concrete security of lattice reduction attacks after observing decryptions of Class 1 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 1 circuit on fresh encryptions. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
		ρ^2_{C1}	1000	350 bikz ≈ 93 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	358 bikz ≈ 95 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	361 bikz ≈ 96 bits
		ρ^2_{C1}	1000	574 bikz ≈ 152 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	592 bikz ≈ 157 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	599 bikz ≈ 159 bits
		ρ^2_{C1}	1000	800 bikz ≈ 212 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	831 bikz ≈ 220 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	844 bikz ≈ 224 bits
		ρ^2_{C1}	1000	346 bikz ≈ 92 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	350 bikz ≈ 93 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	352 bikz ≈ 93 bits
		ρ^2_{C1}	1000	581 bikz ≈ 154 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/158	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	590 bikz ≈ 156 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	593 bikz ≈ 157 bits
		ρ^2_{C1}	1000	815 bikz ≈ 216 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	831 bikz ≈ 220 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	837 bikz ≈ 222 bits
		ρ^2_{C1}	1000	345 bikz ≈ 92 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	347 bikz ≈ 92 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	348 bikz ≈ 92 bits
		ρ^2_{C1}	1000	583 bikz ≈ 154 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	588 bikz ≈ 156 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	589 bikz ≈ 156 bits
		ρ^2_{C1}	1000	820 bikz ≈ 217 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	828 bikz ≈ 219 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	831 bikz ≈ 220 bits
		ρ^2_{C1}	1000	549 bikz ≈ 145 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	550 bikz ≈ 146 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	551 bikz ≈ 146 bits
		ρ^2_{fresh}	1000	705 bikz ≈ 187 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{C1}}$	1000	706 bikz ≈ 187 bits
		$t \cdot \rho^2_{\mathsf{C1}}$	1000	707 bikz ≈ 187 bits

Fig. 10: **Concrete security of lattice reduction attacks after observing decryptions of Class 1 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 1 circuit on fresh encryptions. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
		ρ^2_{C2}	1000	302 bikz ≈ 80 bits
$\log_2 n = 10, \log_2 q_L = 25$	128/102	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	354 bikz ≈ 94 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	377 bikz ≈ 100 bits
		ρ^2_{C2}	1000	467 bikz ≈ 124 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	568 bikz ≈ 150 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	622 bikz ≈ 165 bits
		ρ^2_{C2}	1000	609 bikz ≈ 161 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	772 bikz ≈ 205 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	851 bikz ≈ 226 bits
		ρ^2_{C2}	1000	321 bikz ≈ 85 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	349 bikz ≈ 93 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	361 bikz ≈ 96 bits
		ρ^2_{C2}	1000	517 bikz ≈ 137 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	577 bikz ≈ 153 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	603 bikz ≈ 160 bits
		ρ^2_{C2}	1000	695 bikz ≈ 184 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	794 bikz ≈ 210 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	836 bikz ≈ 222 bits
		ρ^2_{C2}	1000	342 bikz ≈ 91 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	357 bikz ≈ 95 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	363 bikz ≈ 96 bits
		ρ^2_{C2}	1000	557 bikz ≈ 148 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	591 bikz ≈ 157 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	604 bikz ≈ 160 bits
		ρ^2_{C2}	1000	769 bikz ≈ 204 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	848 bikz ≈ 225 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	825 bikz ≈ 219 bits

Fig. 11: **Concrete security of lattice reduction attacks after observing decryptions of Class 2 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security. The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Original Security	Noise Variance	Num Queries (t)	Final Security
		ρ^2_{C2}	1000	351 bikz $\approx$ 93 bits
$\log_2 n = 13, \log_2 q_L = 202$	128/96	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	359 bikz $\approx$ 95 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	362 bikz $\approx$ 96 bits
		ρ^2_{C2}	1000	575 bikz $\approx$ 152 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	592 bikz $\approx$ 157 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	599 bikz $\approx$ 159 bits
		ρ^2_{C2}	1000	802 bikz $\approx$ 213 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	832 bikz $\approx$ 221 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	844 bikz $\approx$ 224 bits
		ρ^2_{C2}	1000	346 bikz $\approx$ 92 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	350 bikz $\approx$ 93 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	352 bikz $\approx$ 93 bits
		ρ^2_{C2}	1000	581 bikz $\approx$ 154 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/158	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	590 bikz $\approx$ 156 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	594 bikz $\approx$ 157 bits
		ρ^2_{C2}	1000	816 bikz $\approx$ 216 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	831 bikz $\approx$ 220 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	837 bikz $\approx$ 222 bits
		ρ^2_{C2}	1000	346 bikz $\approx$ 92 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	348 bikz $\approx$ 92 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	348 bikz $\approx$ 92 bits
		ρ^2_{C2}	1000	583 bikz $\approx$ 155 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	588 bikz $\approx$ 156 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	589 bikz $\approx$ 156 bits
		ρ^2_{C2}	1000	820 bikz $\approx$ 217 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	828 bikz $\approx$ 219 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	831 bikz $\approx$ 220 bits
		ρ^2_{C2}	1000	549 bikz $\approx$ 145 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	550 bikz $\approx$ 146 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	551 bikz $\approx$ 146 bits
		ρ^2_{C2}	1000	705 bikz $\approx$ 187 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{C2}}$	1000	706 bikz $\approx$ 187 bits
		$t \cdot \rho^2_{\mathsf{C2}}$	1000	707 bikz $\approx$ 187 bits

Fig. 11: **Concrete security of lattice reduction attacks after observing decryptions of Class 2 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security. The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{C1}	77	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	3850	0.80
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{C1}	77	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	3850	0.80
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{C1}	77	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	3850	0.80
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{C1}	82	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	4200	0.80
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{C1}	82	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	4200	0.80
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{C1}	82	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	4200	0.80
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{C1}	86	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	4570	0.80
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{C1}	86	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	4570	0.80
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{C1}	86	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	4570	0.80

Fig. 12: **Concrete security of guessing attacks after observing decryptions of Class 1 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained by evaluating a Class 1 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in (17) occurring.

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{C1}	91	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	4930	0.80
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{C1}	91	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	4930	0.80
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{C1}	91	0.81
		$100 \cdot \rho^2_{\mathsf{C1}}$	4930	0.80
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{C1}	96	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	5290	0.80
$\log_2 n = 14, \log_2 q_L = 284$	192/158	ρ^2_{C1}	96	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	5290	0.80
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{C1}	96	0.82
		$100 \cdot \rho^2_{\mathsf{C1}}$	5290	0.80
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{C1}	100	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	5660	0.80
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{C1}	100	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	5660	0.80
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{C1}	100	0.80
		$100 \cdot \rho^2_{\mathsf{C1}}$	5660	0.80
		ρ^2_{C1}	115	0.80
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{C1}}$	6420	0.80
		ρ^2_{C1}	115	0.80
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{C1}}$	6420	0.80

Fig. 12: **Concrete security of guessing attacks after observing decryptions of Class 1 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained by evaluating a Class 1 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in (17) occurring.

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{C2}	97	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	4620	0.80
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{C2}	97	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	4620	0.80
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{C2}	97	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	4620	0.80
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{C2}	103	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5050	0.80
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{C2}	103	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5050	0.80
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{C2}	103	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5050	0.80
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{C2}	110	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	5490	0.80
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{C2}	110	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	5490	0.80
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{C2}	110	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	5490	0.80

Fig. 13: **Concrete security of guessing attacks after observing decryptions of Class 2 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in (17). The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Orig Security	Noise Var	Num Queries	Succ Prob
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{C2}	116	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5930	0.80
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{C2}	116	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5930	0.80
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{C2}	116	0.81
		$100 \cdot \rho^2_{\mathsf{C2}}$	5930	0.80
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{C2}	122	0.80
		$100 \cdot \rho^2_{\mathsf{C2}}$	6370	0.80
$\log_2 n = 14, \log_2 q_L = 284$	192/158	ρ^2_{C2}	122	0.80
		$100 \cdot \rho^2_{\mathsf{C2}}$	6370	0.80
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{C2}	122	0.80
		$100 \cdot \rho^2_{\mathsf{C2}}$	6370	0.80
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{C2}	129	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	6810	0.80
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{C2}	129	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	6810	0.80
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{C2}	129	0.82
		$100 \cdot \rho^2_{\mathsf{C2}}$	6810	0.80
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	ρ^2_{C2}	147	0.80
		$100 \cdot \rho^2_{\mathsf{C2}}$	7720	0.80
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	ρ^2_{C2}	147	0.80
		$100 \cdot \rho^2_{\mathsf{C2}}$	7720	0.80

Fig. 13: **Concrete security of guessing attacks after observing decryptions of Class 2 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The final column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the event in (17). The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{C1}	75	821	0.80	72 bikz $\approx$ 19 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	531	0.80	150 bikz $\approx$ 40 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{C1}	75	821	0.80	159 bikz $\approx$ 42 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	531	0.80	273 bikz $\approx$ 72 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{C1}	75	821	0.80	248 bikz $\approx$ 66 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	531	0.80	394 bikz $\approx$ 104 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{C1}	75	825	0.80	178 bikz $\approx$ 47 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	534	0.80	231 bikz $\approx$ 61 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{C1}	75	825	0.80	314 bikz $\approx$ 83 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	534	0.80	395 bikz $\approx$ 105 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{C1}	75	825	0.80	447 bikz $\approx$ 118 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	534	0.80	553 bikz $\approx$ 147 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{C1}	75	827	0.80	258 bikz $\approx$ 68 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	291 bikz $\approx$ 77 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{C1}	75	827	0.80	436 bikz $\approx$ 115 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	487 bikz $\approx$ 129 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{C1}	75	827	0.80	615 bikz $\approx$ 163 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	683 bikz $\approx$ 181 bits

Fig. 14: **Concrete security of hybrid attacks after observing decryptions of Class 1 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 1 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in (17) and (20) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{C1}	75	828	0.80	305 bikz $\approx$ 81 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	323 bikz $\approx$ 86 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{C1}	75	828	0.80	508 bikz $\approx$ 135 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	537 bikz $\approx$ 142 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{C1}	75	828	0.80	716 bikz $\approx$ 190 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	535	0.80	756 bikz $\approx$ 200 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{C1}	75	828	0.80	323 bikz $\approx$ 86 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	333 bikz $\approx$ 88 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/158	ρ^2_{C1}	75	828	0.80	547 bikz $\approx$ 145 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	562 bikz $\approx$ 149 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{C1}	75	828	0.80	770 bikz $\approx$ 204 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	791 bikz $\approx$ 210 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{C1}	75	829	0.80	334 bikz $\approx$ 88 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	339 bikz $\approx$ 90 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{C1}	75	829	0.80	565 bikz $\approx$ 150 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	573 bikz $\approx$ 152 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{C1}	75	829	0.80	797 bikz $\approx$ 211 bits
		$100 \cdot \rho^2_{\mathsf{C1}}$	3500	536	0.80	808 bikz $\approx$ 214 bits
		ρ^2_{C1}	85	1722	0.80	540 bikz $\approx$ 143 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{C1}}$	4000	1334	0.80	542 bikz $\approx$ 144 bits
		ρ^2_{C1}	85	1722	0.80	693 bikz $\approx$ 184 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{C1}}$	4000	1334	0.80	696 bikz $\approx$ 184 bits

Fig. 14: **Concrete security of hybrid attacks after observing decryptions of Class 1 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C1} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 1 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in (17) and (20) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security.

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 10, \log_2 q_L = 25$	128/102	ρ^2_{C2}	95	892	0.80	53 bikz ≈ 14 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	620	0.80	125 bikz ≈ 33 bits
$\log_2 n = 10, \log_2 q_L = 17$	192/170	ρ^2_{C2}	95	892	0.80	134 bikz ≈ 36 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	620	0.80	235 bikz ≈ 62 bits
$\log_2 n = 10, \log_2 q_L = 13$	256/234	ρ^2_{C2}	95	892	0.80	216 bikz ≈ 57 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	620	0.80	345 bikz ≈ 91 bits
$\log_2 n = 11, \log_2 q_L = 51$	128/97	ρ^2_{C2}	95	896	0.80	166 bikz ≈ 44 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	622	0.80	214 bikz ≈ 57 bits
$\log_2 n = 11, \log_2 q_L = 35$	192/162	ρ^2_{C2}	95	896	0.80	297 bikz ≈ 79 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	622	0.80	370 bikz ≈ 98 bits
$\log_2 n = 11, \log_2 q_L = 27$	256/226	ρ^2_{C2}	95	896	0.80	424 bikz ≈ 112 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	622	0.80	521 bikz ≈ 138 bits
$\log_2 n = 12, \log_2 q_L = 101$	128/97	ρ^2_{C2}	95	897	0.80	251 bikz ≈ 67 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	623	0.80	281 bikz ≈ 74 bits
$\log_2 n = 12, \log_2 q_L = 70$	192/161	ρ^2_{C2}	95	897	0.80	425 bikz ≈ 113 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	623	0.80	471 bikz ≈ 125 bits
$\log_2 n = 12, \log_2 q_L = 54$	256/227	ρ^2_{C2}	95	897	0.80	601 bikz ≈ 159 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	623	0.80	664 bikz ≈ 176 bits

Fig. 15: **Concrete security of hybrid attacks after observing decryptions of Class 2 ciphertexts.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in (17) and (20) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security. The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.

Parameter Set	Orig Security	Noise Var	Num Queries	Num Guess	Succ Prob	Final Security
$\log_2 n = 13, \log_2 q_L = 202$	128/96	ρ^2_{C2}	95	898	0.80	301 bikz $\approx$ 80 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	318 bikz $\approx$ 84 bits
$\log_2 n = 13, \log_2 q_L = 141$	192/159	ρ^2_{C2}	95	898	0.80	502 bikz $\approx$ 133 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	529 bikz $\approx$ 140 bits
$\log_2 n = 13, \log_2 q_L = 109$	256/225	ρ^2_{C2}	95	898	0.80	709 bikz $\approx$ 188 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	745 bikz $\approx$ 197 bits
$\log_2 n = 14, \log_2 q_L = 411$	128/93	ρ^2_{C2}	95	898	0.80	321 bikz $\approx$ 85 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	330 bikz $\approx$ 87 bits
$\log_2 n = 14, \log_2 q_L = 284$	192/158	ρ^2_{C2}	95	898	0.80	543 bikz $\approx$ 144 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	557 bikz $\approx$ 148 bits
$\log_2 n = 14, \log_2 q_L = 220$	256/222	ρ^2_{C2}	95	898	0.80	766 bikz $\approx$ 203 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	786 bikz $\approx$ 208 bits
$\log_2 n = 15, \log_2 q_L = 827$	128/92	ρ^2_{C2}	95	899	0.80	333 bikz $\approx$ 88 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	337 bikz $\approx$ 89 bits
$\log_2 n = 15, \log_2 q_L = 571$	192/156	ρ^2_{C2}	95	899	0.80	564 bikz $\approx$ 149 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	571 bikz $\approx$ 151 bits
$\log_2 n = 15, \log_2 q_L = 443$	256/220	ρ^2_{C2}	95	899	0.80	795 bikz $\approx$ 211 bits
		$100 \cdot \rho^2_{\mathsf{C2}}$	4300	624	0.80	805 bikz $\approx$ 213 bits
		ρ^2_{C2}	110	2508	0.80	535 bikz $\approx$ 142 bits
$\log_2 n = 17, \log_2 q_L = 2400$	140/146	$100 \cdot \rho^2_{\mathsf{C2}}$	4900	1551	0.80	541 bikz $\approx$ 143 bits
		ρ^2_{C2}	110	2508	0.80	688 bikz $\approx$ 182 bits
$\log_2 n = 17, \log_2 q_L = 2000$	193/187	$100 \cdot \rho^2_{\mathsf{C2}}$	4900	1551	0.80	695 bikz $\approx$ 184 bits

Fig. 15: **Concrete security of hybrid attacks after observing decryptions of Class 2 ciphertexts, continued.** For each parameter set, the second column provides the target security as well as the number of bits of security computed by the tool of [17]. The third column indicates the noise-flooding noise added before returning the decryption to the adversary. ρ^2_{C2} is the variance of the noise that is already present in a ciphertext obtained from evaluating a Class 2 circuit on fresh encryptions. The fourth column indicates the number of decryptions observed by the adversary. The fifth column indicates the number of coordinates of the LWE secret that are guessed by the adversary. The sixth column indicates the success probability of the attack, which corresponds to the probability that all guesses are correct, conditioned on the events in (17) and (20) occurring. The final column provides the reduced security level after the attack in terms of bikz (see [17]) and bit-security. The (encoded) message magnitude is equal to $n \cdot \sqrt{\ell/3}$ in all rows, where ℓ is set to 20.