On Circuit Private, Multikey and Threshold Approximate Homomorphic Encryption

Kamil Kluczniak and Giacomo Santato

CISPA Helmholtz Center for Information Security
`{kamil.kluczniak,giacomo.santato}@cispa.de`

Abstract. Homomorphic encryption for approximate arithmetic allows one to encrypt discretized real/complex numbers and evaluate arithmetic circuits over them. The first scheme, called CKKS, was introduced by Cheon et al. (Asiacrypt 2017) and gained tremendous attention. The enthusiasm for CKKS-type encryption stems from its potential to be used in inference or multiparty computation tasks that do not require an exact output.

A desirable property for homomorphic encryption is circuit privacy, which requires that a ciphertext leaks no information on the computation performed to obtain it. Despite numerous improvements directed toward improving efficiency, the question of circuit privacy for approximate homomorphic encryption remains open.

In this paper, we give the first formal study of circuit privacy for homomorphic encryption over approximate arithmetic. We introduce formal models that allow us to reason about circuit privacy. Then, we show that approximate homomorphic encryption can be made circuit private using tools from differential privacy with appropriately chosen parameters. In particular, we show that by applying an exponential (in the security parameter) Gaussian noise on the evaluated ciphertext, we remove useful information on the circuit from the ciphertext. Crucially, we show that the noise parameter is tight, and taking a lower one leads to an efficient adversary against such a system.

We expand our definitions and analysis to the case of multikey and threshold homomorphic encryption for approximate arithmetic. Such schemes allow users to evaluate a function on their combined inputs and learn the output without leaking anything on the inputs. A special case of multikey and threshold encryption schemes defines a so-called partial decryption algorithm where each user publishes a "masked" version of its secret key, allowing all users to decrypt a ciphertext. Similarly, in this case, we show that applying a proper differentially private mechanism gives us IND-CPA-style security where the adversary additionally gets as input the partial decryptions. This is the first security analysis of approximate homomorphic encryption schemes that consider the knowledge of partial decryptions.

As part of our study, we scrutinize recent proposals for the definition and constructions of threshold homomorphic encryption schemes and show new random oracle uninstantiability results that may be of independent interest.

1 Introduction

Fully Homomorphic Encryption (FHE) allows for computations to be performed on encrypted data. A client encrypts a message m and sends the ciphertext to a server, which, given a function F, returns a ciphertext that decrypts to $F(m)$. The concept of FHE was first introduced by Rivest and Dertouzos [RAD78] and later realized by Gentry [Gen09b].

FHE has numerous applications in cryptography. Among others, it is used to build private information retrieval [ABFK16, ALP+21, ACLS18, GH19, CHK22, MW22, HHC+22], secure function delegation [QWW18] and obfuscation schemes [BDGM20, GP21]. Note, however, that the security of fully homomorphic encryption protects only the encrypted message and, in particular, does not offer any protection for the server's computation. In other words, the ciphertexts that a server returns may completely leak the function F.

Circuit privacy, sometimes called function privacy, is a critical property in FHE, where the ciphertext produced by the server, computing a function F on encrypted data, should not reveal any information about F, except for the fact that the ciphertext decrypts to $F(m)$. Circuit private FHE enables semi-honest two-party computation with optimal communication, requiring only one round of communication, and its communication complexity is independent of the size of the computation. Furthermore, the ciphertexts produced by the evaluation process can be reused, making FHE suitable for applications such as private set intersection [HFH99, Mea86, CLR17], neural network inference [DGBL+16, CdWM+17, LJLA17, JKLS18, JVC18, BGGJ18, ABSdV19, CDKS19, RSC+19, BGPG20, KS22], analysis of genomic data [KSK+18, KSK+20, BGPG20], and many more.

Multikey and Threshold Homomorphic Encryption. Extensions of homomorphic encryption like multikey [LTV12, CM15, BP16, MW16, CZW17, CCS19, CDKS19, AJJM20] or threshold homomorphic [BGG+18] encryption allow computing on ciphertexts that come from different parties, but require a subset of secret keys of the different parties to decrypt the outcome of the computation. In particular, many variants of these schemes introduce a so-called partial decryption algorithm, in which each party publishes a secret key capable to "remove an encryption layer" from the evaluated ciphertext. Multikey or threshold homomorphic encryption schemes seem to be related to circuit private encryption schemes, as both give us the means to build two-round multiparty computation if the homomorphic encryption satisfies the right security notion. Namely, whether IND-CPA holds against an adversary that is given partial decryptions of non-corrupted parties. In fact, there is a folklore construction of a circuit private scheme from a multikey homomorphic encryption scheme for at least two keys.

Homomorphic Encryption for Approximate Arithmetic. While we have seen significant advancements in the practical efficiency of fully homomorphic encryption (FHE) schemes and their circuit private versions, realizing practical instances of neural network inference, data analysis problems, or collaborative learning is still relatively slow. In their seminal paper [CKKS17] Cheon et al. noticed that many of these problems do not require the computation on the encrypted data to be exact. In particular, in many applications, it is sufficient for the homomorphic computation to return an approximation of $F(m)$. As a result, they design a homomorphic encryption scheme with a plaintext space of approximations of real or complex numbers.

Due to its native support of real or complex numbers, CKKS-style schemes are believed to be the most competitive solutions for private machine learning inference problems, data analytics, and even training of machine learning models. The focus of researchers is to make CKKS more efficient and increase its plaintext precision. For example, [CDKS19] introduces an efficient multikey version of [CKKS17]. However, it is not clear whether the application is secure and with respect to which security notion. In particular, [CDKS19] states the standard IND-CPA definition, but in applications of multikey homomorphic encryption, we need to make sure that IND-CPA holds even when given partial decryptions.

On the other hand, we may argue that, running an MPC protocol computing the decryption function by inputting the secret keys of all users, can solve the problem. After all, the solution solves the decryption problem in the case of "exact" homomorphic encryption, since the MPC protocol reveals nothing aside from the result of the homomorphic computation. But, unfortunately, in the approximate setting, the decryption gives only an approximation of the exact result, where the approximation error may carry information on the plaintexts of other parties. This means that we need to be careful when trying to apply techniques from the "exact" setting in the approximate setting.

1.1 Our Contributions

In this work, we are the first to formally address the issue of circuit privacy and ciphertext sanitization for homomorphic encryption over approximate arithmetic. Our contributions are as follows.

Formal Definitions. We introduce formal definitions that allow us to reason about circuit privacy for approximate homomorphic encryption. In particular, we expand on some formalism introduced by Li et al. [LMSS22] with regard to the approximate correctness of the computation on ciphertexts. After that, we introduce a indistinguishability-based definition. We note that this is the first indistinguishability-based definition for circuit/function privacy; previously, all definitions were simulation-based, and this also applies in the case of "exact" homomorphic encryption. In particular, the simulation-based definitions imply our, but our is more convenient when dealing with approximate homomorphic computation and showing lower bounds.

Circuit Privacy and Lower Bounds. We give an analysis based on Kullback-Leibler divergence, showing that applying a differentially private mechanism with appropriate parameters gives us circuit privacy. In particular, we can use the Gaussian mechanism to "flood" the approximation errors in a ciphertext. Noise flooding is a known technique, and in particular, [LMSS22] analyzed it in the context of $\mathsf{IND\text{-}CPA^D}$-security [LM21]. Our analysis is inspired by [LMSS22], but we stress that our setting is different in many ways and comes with its own technical challenges which we discuss in the main body of the paper when having the right context. Importantly, we show that the applied noise must be exponential in the security parameter. In particular, we show that, if we apply only a subexponential noise, then there exists an efficient adversary that breaks circuit privacy with non-negligible probability.

Multikey and Threshold Approximate Homomorphic Encryption. We give the first formal study of multikey and threshold homomorphic encryption for approximate arithmetic. There are constructions of such schemes based on CKKS [CDKS19, KKL+22]. However, none of them addresses the relevant security properties. We introduce definitions for indistinguishability security, where an adversary obtains partial decryptions. First, we show that our definitions are meaningful, and multikey and threshold homomorphic encryption satisfying our security notion imply homomorphic encryption satisfying our notion of circuit privacy. Then, we give a Kullback-Leibler-divergence-based proof that applying the Gaussian differential-privacy mechanism in partial decryptions with exponential Gaussian noise is sufficient to satisfy our security notion. On the downside, we show that the applied noise parameters are tight, and using smaller parameters leads to the break of the relevant security property. We note that we can easily adapt our lower bounds to the "exact" setting. Our result in this manner is especially relevant due to the following.

- There is a folklore belief that circuit privacy can be accomplished via multikey (F)HE. The idea is that the server encrypts the circuit with its key, and a client encrypts the query with its key. Then the server computes a universal circuit over both ciphertexts and returns a partial decryption of the evaluated ciphertext back to the client. If the multikey/threshold encryption with partial decryptions gives us a secure MPC protocol, then this approach seems to be correct. Our analysis and lower bounds for the approximate arithmetic setting show that we can indeed use the folklore solution. However, encrypting the circuit does not seem helpful in reducing the flooding noise significantly in comparison to just sanitizing single-key homomorphic encryption.
- There are several recent proposals [DWF22, CSS+22, BS23] to use a noise bounded by a polynomial in the security parameter to implement partial decryption. The idea is to make an analysis based on Rényi divergence. Indeed, in some situations, analysis using the Rényi divergence may result in better parameters [BLR+18]. In this work, we also emphasize certain risks that emerge when considering new security definitions for Threshold (F)HE. We give a new random oracle uninstantiability result against the $\mathsf{OW\text{-}CPA}$ to $\mathsf{IND\text{-}CPA}$ Transform [BS23] (to appear at Asiacrypt 2023). In particular, we show that when instantiating the random oracle with any hash function, the resulting threshold HE scheme is not $\mathsf{IND\text{-}CPA}$-secure if the base

threshold HE is not already IND-CPA-secure. We note that our uninstantiability result works also against the transform from Hofheinz, H ovelmanns, Kiltz [HHK17] on which [BS23] is based. Our counterexample is of the same type as [GKW17] regarding Fujisaki-Okamoto transform [FO99], and [WZ17] regarding Black, Rogaway, Shrimpton transform [BRS03] in that exploits the fact that we can homomorphically evaluate the circuit of a hash function, whereas a random oracle doesn't have a circuit representation. While [FO99] isn't typically used with an FHE scheme, and the counterexamples [GKW17, WZ17] are mainly of theoretical interest, our result shows that the application of the OW-CPA to IND-CPA transform does not upgrade the FHE scheme in question.
Finally, we scrutinize the security definitions from [DWF22] and a previous version of [CSS+22]. Specifically, we demonstrate how these definitions are ineffective in accurately describing the security of Threshold HE schemes because they fail to consider the impact of partial decryptions on the secrecy of messages encrypted by the parties.
- Our results lead to tight estimates of the precision when applying the differential privacy mechanism to CKKS and its multikey/threshold versions. Additionally, we provide revisited parameters for these schemes to achieve the desired security definitions.

1.2 Related Work on Circuit Privacy and Multikey Homomorphic Encryption

Circuit privacy, or sometimes called function or server privacy, was studied before the first secure fully homomorphic encryption schemes were proposed [IP07, Gen09a]. There are two ways to build a circuit private homomorphic encryption scheme. The first is to use a multiparty computation protocol to compute the decryption function on the ciphertext [IP07, GHV10, CO17]. Another way is to sanitize a ciphertext from any information on the circuit. In other words, we apply a random process to the ciphertext in order to make its distribution independent of the circuit. Current approaches to sanitize a ciphertext include noise flooding [Gen09a], repeated bootstrapping [DS16], and re-randomizing computation [BDPMW16, Klu22]. Note that all of these mechanisms apply to "exact" homomorphic encryption. In particular, there is no formal treatment on circuit privacy for approximate homomorphic encryption [CKKS17].

Multikey fully homomorphic encryption was first introduced in [LTV12], and the related concept of threshold homomorphic encryption was introduced in [BGG+18]. For the case of approximate arithmetic, [CDKS19] gave an efficient construction for the multikey setting based on [CKKS17]. They propose to use noise flooding for partially decrypting ciphertexts. However, there is no security proof or even formal definition of what it means for such encryption scheme to be secure aside of IND-CPA security that does not consider adversaries with knowledge of partial decryptions. Mukherjee and Wichs [MW16] define a simulator for partial decryptions in the setting of "exact" GSW [GSW13] encryption to capture the security properties needed to build multiparty computation protocols. Note that such a definition often requires that the homomorphic encryption scheme evaluates the exact circuit, as opposed to approximate. Unfortunately, it is not clear whether we can use such definitions for approximate homomorphic encryption.

2 Preliminaries

We denote an n dimensional column vector as $[f(.,i)]_{i=1}^{n}$, where $f(.,i)$ defines the i-th coordinate. For brevity, we will also denote as $[n]$ the vector $[i]_{i=1}^{n}$.For a random variable $x \in \mathbb{Z}$ we denote as $\mathsf{Var}(x)$ the variance of x, as $\mathsf{stddev}(x)$ its standard deviation and as $\mathsf{E}(x)$ its expectation. By $\mathsf{Ham}(\vec{a})$ we denote the hamming weight of the vector $\vec{a}$, i.e., the number of non-zero coordinates of $\vec{a}$.

We say that an algorithm is **PPT** if it is a probabilistic polynomial-time algorithm. We denote any polynomial as $\texttt{poly}(.)$. We denote as $\texttt{negl}(\lambda)$ a negligible function in $\lambda \in \mathbb{N}$. That is, for any positive polynomial $\texttt{poly}(.)$ there exists $c \in \mathbb{N}$ such that for all $\lambda \geq c$ we have $\texttt{negl}(\lambda) \leq \frac{1}{\texttt{poly}(\lambda)}$. Given two distributions X, Y over a finite domain D, their statistical distance is defined as $\Delta(X,Y) = \frac{1}{2}\sum_{v \in D} |X(v) - Y(v)|$. We say that two distributions are statistically close if their statistical distance is negligible.

Usually, we assume that a probabilistic algorithm $\mathsf{Alg}(x)$ chooses its random coins internally. However, sometimes we write $\mathsf{Alg}(x;r)$ to denote that the random coins $r \xleftarrow{\$} \mathcal{U}$ are used as a seed for Alg, and $\mathsf{Alg}(x;r)$ is deterministic.

2.1 Homomorphic Encryption

We review the definition of Homomorphic Encryption in the public key setting with a particular focus on classical and (static) approximate correctness.

Definition 1 (Homomorphic Encryption). *We define a homomorphic encryption scheme* HE *for a class of circuits* $\mathcal{L}$ *as a tuple of four algorithms* $\mathsf{HE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{Dec})$ *with the following syntax.*

$\texttt{KeyGen}(\lambda) \to (\mathsf{pk}, \mathsf{sk})$: *Given a security parameter* λ*, returns a public key* pk *and a secret key* sk*.*

$\texttt{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$: *Given a public key* pk *and a message* m*, returns a ciphertext* ct*.*

$\texttt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k) \to \mathsf{ct}$: *Given a public key* pk*, a circuit* $C \in \mathcal{L}$ *and ciphertexts* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$*, returns a ciphertext* ct*.*

$\texttt{Dec}(\mathsf{sk}, \mathsf{ct}) \to m$: *Given a secret key* sk *and a ciphertext* ct*, returns a message* m*.*

We denote as $\mathcal{M}$ *the message space,* $\mathcal{C}$ *the ciphertext space and* $\mathcal{L}$ *the class of circuits.*

In this paper, we consider different notions of correctness. In particular, we consider the classical correctness definition and approximate correctness that was recently introduced in [LMSS22] to reason about approximate homomorphic encryption schemes.

Definition 2 (Correctness). *We say that a homomorphic encryption scheme* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *is correct if for all* $C \in \mathcal{L}$*, all* $m_1, \ldots, m_k \in \mathcal{M}$*, all* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathtt{KeyGen}(\lambda)$ *and for all* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$ *such that* $m_i = \mathtt{Dec}(\mathsf{sk}, \mathsf{ct}_i)$ *for* $i \in [k]$*, we have that*

$$\Pr[\mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)) \neq C(m_1, \ldots, m_k)] \leq \mathtt{negl}(\lambda).$$

Below we recall the definition of approximate correctness from [LMSS22]. First, however, we need to formally define the notion of a ciphertext error.

Definition 3 (Ciphertext Error). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be an homomorphic encryption scheme with message space* $\mathcal{M}$*. Furthermore, let* $\mathcal{M}$ *be a normed space with norm* $||\cdot|| : \mathcal{M} \mapsto \mathbb{R}_{\geq 0}$*. For all public/secret key pairs* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathtt{KeyGen}(\lambda)$*, any ciphertext* $\mathsf{ct} \in \mathcal{C}$ *and message* $m \in \mathcal{M}$ *the ciphertext error is defined as*

$$\mathtt{Error}(\mathsf{sk}, \mathsf{ct}, m) = ||\mathtt{Dec}(\mathsf{sk}, \mathsf{ct}) - m||.$$

We can now introduce the approximate correctness notion for approximate HE schemes.

Definition 4 (Approximate Correctness [LMSS22]). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be a homomorphic encryption scheme with message space* $\mathcal{M} \subseteq \widetilde{\mathcal{M}}$ *that is a normed space with norm* $||\cdot|| : \widetilde{\mathcal{M}} \mapsto \mathbb{R}_{\geq 0}$*. Let* $\mathcal{L}$ *be the class of circuits,* $\mathcal{L}_k \subseteq \mathcal{L}$ *be the subset of circuits with* k *input wires, and let* $\mathtt{Estimate} : \bigsqcup_{k \in \mathbb{N}} \mathcal{L}_k \times \mathbb{R}^k_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ *be an efficiently computable function. We call* HE *an approximate homomorphic encryption scheme (w.r.t.* $\mathtt{Estimate}$*) if for all* $k \in \mathbb{N}$*, for all* $C \in \mathcal{L}_k$*, for all* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathtt{KeyGen}(\lambda)$*, if* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$ *and* $m_1, \ldots, m_k$ *are such that* $\mathtt{Error}(\mathsf{sk}, \mathsf{ct}_i, m_i) \leq t_i$*, for some* $t_1, \ldots, t_k \in \mathbb{R}_{\geq 0}$*, then*

$$\mathtt{Error}(\mathsf{sk}, \mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k), C(m_1, \ldots, m_k)) \leq \mathtt{Estimate}(C, t_1, \ldots, t_k).$$

To compute $\mathtt{Estimate}$, we only need the circuit C and upper bounds t_i on the ciphertext errors. This means that the function is publicly and efficiently computable without needing a secret key.

To keep track of the errors when computing on encrypted data, we associate a tag with every ciphertexts. In particular, we define a *tagged ciphertext* $\mathsf{ct} = (\ldots, t)$ where $t \in \mathbb{R}_{\geq 0}$ is an extension of an ordinary ciphertext that also stores t, a *provable upper bound* estimate of the ciphertext error. The noise bound is set to t_{fresh} by $\mathtt{Enc}$ when a ciphertext ct is created. After that, the value of $\mathsf{ct}.t$ is updated using $\mathtt{Estimate}$ every time that a circuit is homomorphically evaluated on ct.

We also recall the definition of $\mathsf{IND\text{-}CPA}$ security for HE schemes.

Definition 5 (IND-CPA security game). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be a homomorphic encryption scheme. We define the* IND-CPA *game as the experiment* $\mathtt{Exp}_b^{\mathsf{IND\text{-}CPA}}$*, where* $b \in \{0,1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The experiment is defined as follow:*

$$\begin{aligned}\mathtt{Exp}_b^{\mathsf{IND\text{-}CPA}}[\mathcal{A}](\lambda): \quad & (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathtt{KeyGen}(\lambda), \\ & b' \leftarrow \mathcal{A}^{\mathsf{E}^b(\mathsf{pk},\cdot,\cdot)}(\mathsf{pk}), \\ & \textbf{\textit{return}}\ b',\end{aligned}$$

where the adversary has access to an encryption oracle $\mathsf{E}^b(\mathsf{pk},\cdot,\cdot)$ *that takes as input* $m_0, m_1 \in \mathcal{M}$ *and returns* $\mathtt{Enc}(\mathsf{pk}, m_b)$*.*

2.2 The CKKS Approximate HE Scheme

We recall the definition of the CKKS approximate HE scheme following the notation used in [LMSS22]. A more detailed description of CKKS can be found in [CKKS17].

Given N, a positive integer, let $\Phi_N(X) = \prod_{j \in \mathbb{Z}_N^*}(X - \omega^j)$ be the N-th cyclotomic polynomial, where $\omega \in \mathbb{C}$ is one of the principal N-th root of unity and $\mathbb{Z}_N^*$ is the group of invertible integers modulo N and has order $\varphi(N)$. We denote by $\mathcal{R}_Q$ the ring $\mathbb{Z}_Q[X]/(\Phi_N(X))$, where $\mathbb{Z}_Q$ is the ring of integers modulo Q. We will omit Q when it is clear form the context. The CKKS scheme is able to encrypt complex ciphertext by using the canonical embedding $\tau : \mathbb{Q}[X]/(\Phi_N(X)) \to \mathbb{C}^{\varphi(N)}$; this embedding is defined by sending the polynomial $a(X)$ in the tuple of its evaluations in the principal N-th complex roots of unity, so in the tuple $(a(\omega^j))_{j \in \mathbb{Z}_N^*}$. Moreover, the $n = \varphi(N)$ complex values in each image come in conjugate pairs $(a(\omega^j), a(\omega^{N-j}))$, so it is possible to obtain a projection π to $\mathbb{C}^{n/2}$ by considering only one of the two elements for every complex pair. Using this function, vectors $z \in \mathbb{C}^{n/2}$ are considered as messages in CKKS. Complex messages are transformed to polynomials in $\mathcal{R}$ using the inverses of π and φ on a scaled vector $\delta \cdot z$, for some scaling factor $\delta \in \mathbb{R}$ such that $\|\delta \cdot z\| \ll Q$ and then by rounding the result to a polynomial in $\mathcal{R}$. More in detail, the functions that link vectors in $\mathbb{C}^{n/2}$ to plaintext polynomials in $\mathcal{R}$ are

$$\begin{aligned}\mathsf{CKKS}.\mathtt{Encode}(z \in \mathbb{C}^{n/2}, \delta) &= \lfloor \delta \cdot \varphi^{-1}(\pi^{-1}(z)) \rceil; \\ \mathsf{CKKS}.\mathtt{Decode}(a(X) \in \mathcal{R}, \delta) &= \pi(\varphi(\delta^{-1} \cdot a(X))).\end{aligned}$$

These two functions do not require the knowledge of any secret key nor public key. In the main implementations of CKKS they are, respectively, included in encryption and decryption but for theoretical analysis we will consider them separately. This allow us to study express more clearly the error that arise from the message encoding and to differentiate it from the other errors in this scheme.

Another useful tool to track the ciphertext error in CKKS is the norm induced on $\mathcal{R}$ by the canonical embedding $\pi \circ \varphi$. This norm is defined as $\|a\|_{\mathsf{can}} = \|\pi \circ \varphi(a)\|_\infty$.

We now give a broad description of the main algorithms in the CKKS scheme that we still have not introduced. The parameters of the scheme are: the plaintext polynomial ring $\mathcal{R}$ with ring dimension N typically chosen as a power of two, a ciphertext modulo Q and a discrete Gaussian error χ with standard deviation σ.

$\mathsf{CKKS}.\mathtt{KeyGen}(\lambda)$**:** Given the security parameter λ choose $p \in \mathbb{N}$ and $Q \in \mathbb{N}$, the ring $\mathcal{R}$ and the noise distribution χ. Sample $s \in \mathcal{R}_{pQ}$ by sampling each coefficient uniformly from $\{-1, 0, 1\}$ and set $\mathsf{sk} = s$. Sample $\mathsf{pk}.a \xleftarrow{\$} \mathcal{R}_Q$, $e \xleftarrow{\$} \chi$ and compute $\mathsf{pk}.b = -as + e$. Then sample $\mathsf{pk}.a' \xleftarrow{\$} \mathcal{R}_Q$, $e' \xleftarrow{\$} \chi$ and compute $\mathsf{pk}.b' = -a's + e + s^2$.

$\mathsf{CKKS}.\mathtt{Enc}(\mathsf{pk}, m \in \mathcal{R}_Q)$**:** Choose $r \in \mathcal{R}$ such that every coefficient (chosen independently) has probability 1/4 to be 1 and -1, and probability 1/2 to be 0. Sample $e_0, e_1 \leftarrow \chi$. Set $\mathsf{ct}.a = r\mathsf{pk}.a + e_1$, $\mathsf{ct}.b = r\mathsf{pk}.b + e_2 + m$ and return ct.

$\mathsf{CKKS}.\mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ **:** The algorithm evaluates the arithmetic circuit C by means of addition and multiplication:

- $\mathsf{CKKS}.\mathtt{Add}(\mathsf{pk}, \mathsf{ct}_0, \mathsf{ct}_1 \in \mathcal{R}_Q)$**:** Set $\mathsf{ct}.a = \mathsf{ct}_0.a + \mathsf{ct}_1.a$, $\mathsf{ct}.b = \mathsf{ct}_0.b + \mathsf{ct}_1.b$ and return ct.
- $\mathsf{CKKS}.\mathtt{Mul}(\mathsf{pk}, \mathsf{ct}_0, \mathsf{ct}_1 \in \mathcal{R}_Q)$**:** Set $\mathsf{ct}.b = \mathsf{ct}_0.b \cdot \mathsf{ct}_1.b + \lfloor (\mathsf{ct}_0.a \cdot \mathsf{ct}_1.a \cdot \mathsf{pk}.b')/p \rceil$, and $\mathsf{ct}.a = \mathsf{ct}_0.a \cdot \mathsf{ct}_1.b + \mathsf{ct}_1.a \cdot \mathsf{ct}_0.b + \lfloor (\mathsf{ct}_0.a \cdot \mathsf{ct}_1.a \cdot \mathsf{pk}.a')/p \rceil$. Return ct.

$\mathsf{CKKS}.\mathtt{Dec}(\mathsf{sk}, \mathsf{ct})$**:** Return $\mathsf{ct}.b + \mathsf{ct}.a \cdot \mathsf{sk}$.

We also recall the basic expressions of noise growth during addition and multiplication in CKKS.

Lemma 1 (Lemma 3 of [CKKS17]). *Let* $\mathsf{ct}_i = \mathsf{CKKS}.\mathtt{Enc}(\mathsf{pk}, m_i)$ *for* $i \in \{0, 1\}$ *and their ciphertext error be, respectively,* $\mathtt{Error}(\mathsf{sk}, ct_i, m_i) = e_i$*. The ciphertext error of the sum of both ciphertexts is equal to* $e_0 + e_1$ *and the ciphertext error their product is equal to* $m_0 e_1 + m_1 e_0 + e_0 e_1 + e_{\mathsf{mult}}$*, where the term* e_{mult} *depends on the parameters of the scheme and on the two ciphertexts* $\mathsf{ct}_0, \mathsf{ct}_1$*.*

We now give a brief explanation on how the tagged ciphertext and the `Estimate` function are handled by the algorithms of the CKKS scheme. $\mathsf{CKKS}.\mathtt{Enc}$ assigns to the returned ciphertext an upper bound of the ciphertext error for fresh encryptions. $\mathsf{CKKS}.\mathtt{Add}$ and $\mathsf{CKKS}.\mathtt{Mul}$ follow the noise growth rules of Lemma 1 to assign to the returned ciphertext a noise estimate. More in general, when homomorphic evaluating a circuit C in CKKS by computing $\mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$, it is always possible to publicly compute the resulting noise estimate by combining the two noise growth rules for sum and product using as an input only the description of C and the noise estimates on the input ciphertexts.

2.3 Probability

A probability ensemble $(\mathcal{P}_\theta)_\theta$ is a family of probability distributions parameterized by a variable θ. The KL Divergence is a useful tool to handle probability distributions. In particular, it gives us a way to understand how close (or far) are two distributions from each other.

Definition 6 (KL divergence). *Let $\mathcal{P}$ and $\mathcal{Q}$ be two probability distributions with common support X. The Kullback-Leibler Divergence between $\mathcal{P}$ and $\mathcal{Q}$ is* $D(\mathcal{P}||\mathcal{Q}) := \sum_{x \in X} \Pr[\mathcal{P} = x] \ln\left(\frac{\Pr[\mathcal{P}=x]}{\Pr[\mathcal{Q}=x]}\right)$.

Lemma 2 (Subadditivity of KL divergence for Joint Distributions, Theorem 2.2 of [PW14]). *If $(\mathcal{X}_0, \mathcal{X}_1)$ and $(\mathcal{Y}_0, \mathcal{Y}_1)$ are pairs of (possibly dependent) random variables, then*

$$D((\mathcal{X}_0, \mathcal{X}_1)||(\mathcal{Y}_0, \mathcal{Y}_1)) \leq \max_x D((\mathcal{X}_1|x)||(\mathcal{Y}_1||x)) + D(\mathcal{X}_0, \mathcal{Y}_1)$$

Computing the advantages of adversaries from Subsection 4.3 and from Subsection 5.5 will require the following inequality about the total variation distance between two Gaussian distributions.

Theorem 1 (Theorem 1.3 of [DMR18]). *Let $\sigma_0, \sigma_1 > 0$. Then*

$$\Delta(\mathcal{N}(\mu_0, \sigma_0^2), \mathcal{N}(\mu_1, \sigma_1^2)) \geq \frac{1}{200} \min\{1, \frac{40|\mu_0 - \mu_1|}{\sigma_0}\}.$$

2.4 KL Differential Privacy

In [LMSS22], Li et al. introduce the new notion of *Norm Rényi Differential Privacy* by generalizing the notion of Rényi differential privacy [Mir17] to different norms. This innovative technique aims to address the primary technical challenges encountered when applying differential privacy in environments with arbitrary norms. Specifically, within the context of Differential Privacy, the concept of "adjacent" values is commonly assessed using the Hamming norm, whereas Approximate HE revolves around Euclidean and Infinity norms. In this paper, we will focus exclusively on the specific instance of this definition that uses KL divergence.

Definition 7 (Norm KL Diff. Privacy, Definition 14 of [LMSS22]). *For $t \in \mathbb{R}_{\geq 0}$, let $M_t : B \to C$ be a family of randomized algorithms, where B is a normed space with norm $||\cdot|| : B \to \mathbb{R}_{\geq 0}$. Let $\rho \in \mathbb{R}$ be a privacy bound. We say that the family M_t is ρ-KL differentially private (ρ-KLDP) if, for all $x, x' \in B$ with $||x - x'|| \leq t$,*

$$D(M_t(x)||M_t(x')) \leq \rho.$$

Definition 8. *Let $\rho > 0$ and $n \in \mathbb{N}$. Define the (discrete) Gaussian Mechanism $M_t : \mathbb{Z}^n \to \mathbb{Z}^n$ be the mechanism that, on input $x \in \mathbb{Z}^n$, outputs a sample from $\mathcal{N}_{\mathbb{Z}^n}(x, \frac{t^2}{2\rho} I_n)$.*

Theorem 2. *For any $\rho > 0$, $n \in \mathbb{N}$, the Gaussian mechanism is ρ-KLDP.*

2.5 Bit security

One of the original motivations of this work was to extend the security analysis beyond the use of statistical distance in the hope of providing tighter noise bounds and improving the parameters. Using Rényi divergence when studying decisional problems is an important technique introduced in [BLR+18], and that has been proved useful in lattice-based cryptography to obtain a tighter security analysis and to improve the parameters. Finally, we choose to analyze bit security due to the technical synergies with KL divergence (Theorem 3) and KL Differential Privacy (Theorem 4).

We briefly recall the notion of bit security from [MW18].

Definition 9 (Indistinguishability Game). *Let $\{\mathcal{D}^0_\theta\}$ and $\{\mathcal{D}^1_\theta\}$ be two distributions ensembles. The indistinguishability game is defined as follows: the challenger C chooses $b \leftarrow \mathcal{U}(\{0,1\})$. At any time after that, the adversary $\mathcal{A}$ may send (adaptively chosen) query strings θ_i to C and obtain samples $c_i \leftarrow \mathcal{D}^b_{\theta_i}$. The goal of the adversary is to output $b' = b$.*

Definition 10 (Bit Security). *For any adversary $\mathcal{A}$ playing an indistinguishability game $\mathcal{G}$, we define its*

output probability as $\alpha^{\mathcal{A}} = \Pr[\mathcal{A} \neq \perp]$ and its
conditional success probability as $\beta^{\mathcal{A}} = \Pr[b' = b | \mathcal{A} \neq \perp]$.

where the probabilities are taken over the randomness of the entire indistinguishability game (including the internal randomness of $\mathcal{A}$). We also define $\mathcal{A}$'s

conditional distinguishing advantage as $\delta^{\mathcal{A}} = 2\beta^{\mathcal{A}} - 1$ and
the advantage of $\mathcal{A}$ as $\mathsf{adv}^{\mathcal{A}} = \alpha^{\mathcal{A}}(\delta^{\mathcal{A}})^2$.

The bit security of the indistinguishability game is $\min_{\mathcal{A}} \log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}}$, where $T(\mathcal{A})$ is the running time of $\mathcal{A}$.

We can use bit security on the indistinguishability game from Definition 5.

Definition 11 (IND-CPA-security). *A homomorphic encryption scheme* HE *is said to be λ-bit* IND-CPA*-secure if, for any adversary $\mathcal{A}$ in the* IND-CPA *security game, we have that $\lambda \leq \log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}}$, where $\mathsf{adv}^{\mathcal{A}}$ is defined as in Definition 10.*

Theorem 3. *[Theorem 1 of [LMSS22]] Let $\mathcal{G}^{\mathcal{P}}$ be an indistinguishability game with black-box access to a probability ensemble $\mathcal{P}_\theta$. If $\mathcal{G}^{\mathcal{P}_\theta}$ is k-bit secure, and also $\max_\theta D(\mathcal{P}_\theta || \mathcal{Q}_\theta) \leq 2^{-k+1}$, then $\mathcal{G}^{\mathcal{Q}_\theta}$ is $(k-8)$-bit secure.*

Theorem 4. *[Lemma 5 of [LMSS22]] Let $\mathcal{G}$ be the indistinguishability game instantiated with distribution ensembles $\{\mathcal{X}_\theta\}_\theta$ and $\{\mathcal{Y}_\theta\}_\theta$, where $\theta \in \Theta$. Let $q \in \mathbb{N}$. Then, for any (potentially computationally unbounded) adversary $\mathcal{A}$ making at most q queries to its oracle, we have that*

$$\mathsf{adv}^{\mathcal{A}} \leq \frac{q}{2} \max_{\theta \in \Theta} D(\mathcal{X}_\theta || \mathcal{Y}_\theta).$$

3 Defining Circuit Privacy for Approximate HE

In this section, we recall the (classic) simulation-based definition of circuit privacy introduced by Gentry [Gen09a]. Then we give our relaxed indistinguishability definition.

We start by stating Gentry's [Gen09a] simulation-based definition below.

Definition 12 (Circuit Privacy). *A homomorphic encryption scheme* HE *for a class of circuits* $\mathcal{L}$ *is said to be circuit private if there exists a* **PPT** *simulator* $\mathtt{Sim}$ *such that, for any* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$ *valid ciphertexts,*

$$\Delta(\mathtt{Sim}(\mathsf{pk}, m_{\mathsf{out}}), \mathtt{Eval}(\mathsf{pk}, \mathsf{ct}_1, \ldots, \mathsf{ct}_k, C)) \leq \mathtt{negl}(\lambda),$$

where $C \in \mathcal{L}$, $[m_i \leftarrow \mathtt{Dec}(\mathsf{sk}, \mathsf{ct}_i)]_{i=1}^{k}$, $m_{\mathsf{out}} \leftarrow C(m_1, \ldots, m_k)$ *and* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathtt{KeyGen}(\lambda)$.

Definition 12 gives us a very strong privacy guarantee. In particular, the simulator should produce a ciphertext that is statistically indistinguishable from the homomorphic computation while obtaining only the outcome of an evaluation. This means that the evaluation process reveals no information on the circuit aside from the output of the circuit evaluation. On the other hand, as we discussed in Section 2, homomorphic encryption for approximate arithmetic introduces errors to the outcome of the evaluation. Consequently, the output of the computation may depend somehow on the evaluated circuit. For instance, already the magnitude of the error reveals the size of the circuit or its topology. Finally, note that the simulation definition implicitly induces a requirement that the homomorphic computation is exact. In particular, using $m_{\mathsf{out}} \leftarrow C(m_1, \ldots, m_k)$ to simulate a ciphertext completely ignores the fact that the homomorphic evaluation is approximate, and that the resulting ciphertext $\mathsf{ct}_{\mathsf{res}} \leftarrow \mathtt{Eval}(\mathsf{pk}, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ is now encrypting the message $\mathtt{Dec}(\mathsf{sk}, \mathsf{ct}_{\mathsf{res}})$, that is different from m_{out}. Unfortunately, due to this correctness requirement, we cannot use such a definition to reason about circuit privacy for approximate homomorphic encryption. This state of affairs motivates us to state a relaxed definition of circuit privacy which is sufficient for many applications and gives us a framework to analyze circuit privacy in the case of approximate homomorphic encryption.

We give our definition below.

Definition 13 (Indistinguishability Circuit Privacy). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be a homomorphic encryption scheme for circuits in* $\mathcal{L}$. *We define the experiment* $\mathtt{Exp}_b^{\mathsf{IND\text{-}CP}}[\mathcal{A}]$, *where* $b \in \{0, 1\}$ *is a bit and* $\mathcal{A}$ *is an adversary.*

The experiment is defined as follow:

$$\begin{aligned}
\mathtt{Exp}_b^{\mathsf{IND\text{-}CP}}[\mathcal{A}](\lambda): \quad & r, r_1, \ldots, r_n \xleftarrow{\$} \mathcal{U}, \\
& m_1, \ldots, m_n, C_0, C_1, \mathsf{st} \leftarrow \mathcal{A}(\lambda, r, r_1, \ldots, r_n), \\
& (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda; r), \\
& [\mathsf{ct}_i \leftarrow \mathtt{Enc}(\mathsf{pk}, m_i; r_i)]_{i=1}^n, \\
& \mathsf{ct} \leftarrow \mathtt{Eval}(\mathsf{pk}, C_b, \mathsf{ct}_1, \ldots, \mathsf{ct}_n), \\
& b' \leftarrow \mathcal{A}(\mathsf{st}, \mathsf{ct}), \\
& \textbf{\textit{return}}\ b'.
\end{aligned}$$

where $C_0, C_1 \in \mathcal{L}$ and $C_0(m_1, \ldots, m_n) = C_1(m_1, \ldots, m_n)$. The scheme HE *is said to be λ-bit* IND-CP*-secure if, for any adversary $\mathcal{A}$, we have that $\lambda \leq \log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}}$, where $\mathsf{adv}^{\mathcal{A}}$ is defined as in Definition 10.*

In this definition, the adversary receives the random coins used by the `KeyGen` and the `Enc` algorithms. Therefore, sk,pk and the ct_i are honestly generated, and the adversary can compute sk and pk.

4 Circuit Privacy in CKKS

In Subsection 4.1 we present a modification of the CKKS approximate homomorphic encryption scheme that satisfies indistinguishability circuit privacy as given by Definition 13. In particular, we show that re-randomized CKKS ciphertexts are circuit private when we apply an appropriate differential privacy mechanism that floods the ciphertexts noise with an exponential Gaussian sample. In Subsection 4.2 we show how to choose parameters for the differential privacy mechanism for the class of circuits that consists of multivariate polynomials of bounded degree. Finally, in Subsection 4.3, we show that the parameters are tight. Namely, the Gaussian noise must be exponential in the security parameter, and a significantly lower noise parameter leads to an efficient adversary against IND-CP-security.

4.1 IND-CP-secure CKKS

To get circuit privacy we modify the CKKS.`Eval` algorithm, which we describe at Algorithm 1. The main idea is to post-process the ciphertext after evaluation. Namely, we re-randomize the ciphertext with a freshly sampled encryption of zero, and we apply a proper differential privacy mechanism.

Note that to run the discrete Gaussian mechanism we need to redefine the `Estimate` algorithm such that it outputs an upper bound which depends on a class of circuits instead of just the noise upper bound for a given circuit. Concretely we estimate the noise tag as $\max_{C \in \mathcal{L}} \{\mathtt{Estimate}(C, t_{\mathsf{fresh}}, \ldots, t_{\mathsf{fresh}})\}$ for a class of circuits $\mathcal{L}$; we refer to this noise estimate as $T_{\max}$.

Algorithm 1: The modified CKKS evaluation $\texttt{Eval}_{\mathcal{L}}$

Data: A public key pk, circuit $C \in \mathcal{L}$, a vector of ciphertexts $\mathsf{ct}_1, \dots, \mathsf{ct}_k$.

begin

$\mathsf{ct} \leftarrow \texttt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \dots, \mathsf{ct}_k)$;

$\mathsf{ct}.t \leftarrow \max_{D \in \mathcal{L}}\{\texttt{Estimate}(D, \mathsf{ct}_1.t, \dots, \mathsf{ct}_k.t)\}$;

$\mathsf{ct} \leftarrow \mathsf{ct} + \texttt{Enc}(\mathsf{pk}, 0)$;

$\mathsf{ct}.b \leftarrow M_{\mathsf{ct}.t}(\mathsf{ct}.b)$;

return ct ;

Theorem 5. *Let* $\mathsf{CKKS} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{Dec})$ *be the CKKS approximate encryption scheme, with the normed plaintext space* $\mathcal{R}$ *and estimate function* $\texttt{Estimate}$. *Let* M_t *be a* ρ*-KLDP mechanism on* $\mathcal{R}$ *where* $\rho \leq 2^{-\lambda-7}$. *Then,* CKKS *with the modified* $\texttt{Eval}_{\mathcal{L}}$ *given by Algorithm 1 is* λ*-bit secure in the* $\mathsf{IND\text{-}CP}$ *game for the circuit space* $\mathcal{L}$.

Proof. We give a brief overview of the structure of the proof. First, we construct a new $(\lambda + 8)$-bit secure indistinguishability game. After that, we consider the output to any adversary's query in this game and in the $\mathsf{IND\text{-}CP}$ game, and we study the KL-divergence between them. In order to bound the KL-divergence, we compute the difference of some entries in the outputs, upper-bound their norm, and then use subadditivity (Lemma 2) and differential privacy (Definition 7). Finally, once we have obtained a bound on the KL-divergence, we can link the bit security of the two games and conclude the proof.

The full version of the proof is deferred to Appendix B.

Analysis of the post-processing noise. We give an analysis of the precision lost when modifying the CKKS scheme as in Theorem 5. We instantiate the differential privacy mechanism from Definition 8 with $\rho = 2^{-\lambda-7}$. Considering that the static estimate $\mathsf{ct}.t$ is expressed in the infinity canonical norm and not in the euclidean norm, we obtain that a Gaussian noise of standard deviation $8\sqrt{n2^{\lambda}}T_{\max}$ is added to each coordinate, where n is the dimension of the ring. We obtain that the bits of precision lost are $\lambda/2 + 3 + \log_2(T_{\max}) + \log_2(\sqrt{n})$.

Parameters for Machine Learning Inference. Tables 1 gives parameters for one of the most common applications of FHE which benefits from circuit privacy: privacy-preserving machine learning inference on a model with depth d and width w. For the base CKKS scheme, we consider parameters such as ring dimension and ciphertext modulus from [MA18]. In particular, we set the ring dimension to be smaller or equal to 2^{15} and the standard deviation for fresh encryption σ_{fresh} to be 3.2.

4.2 Managing and obtaining $T_{\max}$

In this section, we will show how to set the noise bound $T_{\max}$ for the differential privacy mechanism. Remind that the usual noise estimation algorithm estimates

		width			
		$w = 1$	$w = 2^3$	$w = 2^5$	$w = 2^8$
depth	$d = 1$	85.50	87.67	89.54	92.50
	$d = 2$	97.08	100.99	104.63	110.51
	$d = 3$	108.08	113.45	118.76	127.53

Table 1. Bits of additional Gaussian noise added in the modified CKKS of Theorem 5 to achieve 128-bits $\mathsf{IND\text{-}CP}$-security. We use the estimates on $T_{\max}$ from subsection 4.2 with message bound $B = 2^{10}$.

the noise based on the circuit, which is enough for $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$-security when post-processing decryption as in [LMSS22]. To obtain circuit privacy, instead, we estimate the noise as the maximum noise over all circuits in a given class of circuits. In particular, we run $T_{\max} := \max_{D \in \mathcal{L}}\{\mathtt{Estimate}(D, t_{\mathsf{fresh}}, \ldots, t_{\mathsf{fresh}})\}$. Note that the estimation algorithm depends on the class of circuits; hence the evaluation process may still leak some information on the computation, like the multiplicative depth of the circuit. Below we show how to estimate the noise tag for the class of multivariate polynomials of degree bounded by some $d \in \mathbb{N}$.

Theorem 6. *Let $k, d \in \mathbb{N}$. Let $C(x_1, \ldots, x_k)$ be a multivariate polynomial of degree smaller or equal to d. Let $B \in \mathbb{N}$ such that $\|m_i\|_{\mathsf{can}} \leq B$ for $i \in [k]$, then*

$$\mathtt{Estimate}(\mathsf{sk}, \mathsf{CKKS}.\mathtt{Eval}(\mathsf{pk}, C, [\mathsf{ct}_i]_{i\in[k]}), C([m_i]_{i\in[k]})) = d\binom{k+d}{d} O(B^d t_{\mathsf{fresh}})$$

where $\mathsf{ct}_i \leftarrow \mathtt{Enc}(\mathsf{pk}, m_i)$ for $i \in [k]$.

Proof. Deferred to Appendix C.

4.3 Tightness of the Differential Privacy Parameters

As shown by Theorem 5, the proposed modified version of CKKS achieves λ-bit $\mathsf{IND\text{-}CP}$-security by applying a differentially private mechanism on the outcome of the evaluation algorithm. In practice, we instantiate the differential privacy mechanism by the Gaussian mechanism with Gaussian noise of variance $\sigma_{\mathsf{max}} \leftarrow \frac{T_{\max}^2}{2\rho}$. Remind that $\rho \leq 2^{-\lambda-7}$ is the privacy bound for ρ-KL differential privacy (Definition 7), and $T_{\max}$ is the noise upper bound for the class of circuits. We show that trying to use an appreciably smaller variance $\sigma_{\mathsf{s}} \ll \sigma_{\mathsf{max}}$ leads to the existence of an adversary that wins the $\mathsf{IND\text{-}CP}$ game with a non-negligible advantage. In other words, we show that the noise parameters are tight when using the Gaussian mechanism, and the added Gaussian noise must be exponential in the security parameter.

Theorem 7. *Let $\sigma_{\mathsf{s}} > 0$. Let $\mathtt{Eval}^{\sigma_{\mathsf{s}}}_{\mathcal{L}_d}$ be the modified CKKS evaluation given by Algorithm 1 but where the post-processing noise is sampled from the discrete*

Algorithm 2: Adversary $\mathcal{A}(\lambda)$.

Data: A security parameter λ. The adversary has oracle access to $\mathtt{Eval}_{\mathcal{L}_d}^{\sigma_{\mathsf{s}}}$.

begin

$r, r_1 \xleftarrow{\$} \mathcal{U}$;
$(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda; r)$;
$m, C_0, C_1 \leftarrow B, x^d, x^d + Bx^{d-1} - B^d$;
$\mathsf{ct} \leftarrow \mathtt{Enc}(\mathsf{pk}, m; r_1)$;
$\mathsf{ct}_{\mathsf{res}} \leftarrow \mathcal{O}^{\mathtt{Eval}_{\mathcal{L}_d}^{\sigma_{\mathsf{s}}}(\mathsf{pk}, \cdot, \cdot, \mathsf{ct})}(C_0, C_1)$;
$e_0 \leftarrow \mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(C_0, \mathsf{ct})) - B^d$;
$e_1 \leftarrow \mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(C_1, \mathsf{ct})) - B^d$;
$e_{\mathsf{res}} \leftarrow \mathtt{Dec}(\mathsf{sk}, \mathsf{ct}_{\mathsf{res}}) - B^d$;
Choose $i \in \{0, \dots, n-1\}$ such that $|e_{0,i} - e_{1,i}|$ is maximal;
If $|e_{\mathsf{res},i} - e_{0,i}| \leq |e_{\mathsf{res},i} - e_{1,i}|$ then return 0. Otherwise output 1;

Gaussian $\mathcal{N}_{\mathbb{Z}^n}(0, \sigma_{\mathsf{s}}^2 T_{\max}^2 I_n)$. Then there exists an adversary $\mathcal{A}$ (Algorithm 2) against $\mathsf{CKKS}_{\mathcal{L}_d}^{\sigma_{\mathsf{s}}}$ in the $\mathsf{IND\text{-}CP}$-game such that $\mathsf{adv}^{\mathcal{A}} = \Omega(\frac{1}{\sigma_{\mathsf{s}}^2 B^2 t_{\mathsf{fresh}}^2})$, where B is an upper bound on the messages norm modulus and t_{fresh} is the noise tag associated to freshly encrypted messages.

To prove Theorem 7 we need the following inequality that we can derive, for this case, from Theorem 1.

Lemma 3 (Theorem 1.3 of [DMR18]). *Let $\sigma > 0$. Then*

$$\Delta(\mathcal{N}(\mu_0, \sigma^2), \mathcal{N}(\mu_1, \sigma^2)) \geq \frac{1}{50} \frac{|\mu_0 - \mu_1|}{\sigma}.$$

Again, to prove Theorem 7 we need the following lemma.

Lemma 4. *Let $d \in \mathbb{N}$. Let B be the plaintext modulus and $\mathsf{ct} \leftarrow \mathtt{Enc}(\mathsf{pk}, B)$, then*

$$\mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(x^d, \mathsf{ct})) - B^d = dB^{d-1}\mathsf{ct}.e + f$$

where $\|f\|_{\mathsf{can}} = O(B^{d-1})$.

Proof. Deferred to Appendix D.

Proof (of Theorem 7). Deferred to Appendix E.

Theorem 8. *If the CKKS scheme with the modified evaluation $\mathtt{Eval}_{\mathcal{L}_d}^{\sigma_{\mathsf{s}}}$ is λ-bit $\mathsf{IND\text{-}CP}$-secure, then $\sigma_{\mathsf{s}} = \Omega(2^{\lambda/2}/(B^2 t_{\mathsf{fresh}}^2))$. This implies that one must add at least $\lambda/2 - \log_2 \tilde{\Omega}(B^2 t_{\mathsf{fresh}}^2)$ bits of additional Gaussian noise.*

Proof. By using the definition of bit security, we know that $\lambda \leq \log_2 O(\frac{T(A)}{\mathsf{adv}^A}) \leq \log_2 O(\sigma_s^2 B^2 t_{\mathsf{fresh}}^2)$; this immediately implies that $\sigma_s \geq 2^{\lambda/2}/(B^2 t_{\mathsf{fresh}}^2)$ and $\lambda/2 - \log_2 \Omega(B^2 t_{\mathsf{fresh}}^2) \leq \log_2 \sigma_s$.

5 Threshold FHE and MPC

In Subsection 5.1, we give definitions for threshold homomorphic encryption over approximate arithmetic. In Subsection 5.2, we analyze recent alternative definitions for threshold homomorphic encryption in the exact setting and highlight the risks that arise when considering new security definitions for these schemes. In Subsection 5.3, we give definitions for multikey homomorphic encryption over approximate arithmetic. In Subsection 5.4 we present a modification of the MK-CKKS multikey homomorphic encryption scheme that satisfies the indistinguishability security definition as given by Definition 21. In particular, we show that re-randomized MK-CKKS ciphertexts and decryption shares does not reveal information about messages and secret keys of non-corrupted parties when we apply an appropriate differential privacy mechanism that floods them with an exponential Gaussian sample. Finally, in Subsection 5.5, we show that the parameters are tight. Namely, the Gaussian noise must be exponential in the security parameter, and a significantly lower noise parameter leads to an efficient adversary against IND-MKHE-security.

5.1 Threshold Homomorphic Encryption

We base our definition for threshold approximate homomorphic encryption on the definition introduced by [BGG$^+$18]. We have the same syntax and we have the same indistinguishability definition as [BGG$^+$18], but we redefine the correctness definition for the case of approximate arithmetic. Regarding the indistinguishability, we discuss in Remark 2 a slight strengthening of the definition that lets us construct a meaningful circuit private homomorphic encryption scheme.

Recall that a monotone access structure $\mathbb{A}$ on $[n]$ is a collection $\mathbb{A} \subseteq \mathcal{P}([n])$, where $\mathcal{P}([n])$ contains all subsets of $[n]$, such that whenever we have sets B, C satisfying $B \in \mathbb{A}$ and $B \subseteq C \subseteq [n]$ then $C \in \mathbb{A}$. The sets in $\mathbb{A}$ are called the valid sets and the sets in $\mathcal{P}([n]) \setminus \mathbb{A}$ are called invalid sets.A class of monotone access structures is a collection $\mathcal{S} = (\mathbb{A}_1, \ldots, \mathbb{A}_t) \subseteq \mathcal{P}(\mathcal{P}([n]))$ of monotone access structures on $[n]$. A set $S \subseteq [n]$ is a maximal invalid share set if $S \notin \mathbb{A}$ and for every $i \in [n] \setminus S$ we have that $S \cup \{i\} \in \mathbb{A}$.

Definition 14 (Threshold Homomorphic Encryption). *Let $d \in \mathbb{N}$ and let $\mathcal{L}_d$ be a class of circuits of multiplicative depth smaller or equal to d. A threshold homomorphic encryption scheme* THE *on $\mathcal{L}_d$ is a tuple of five algorithms* THE $=$ (`KeyGen`, `Enc`, `Eval`, `PDec`, `Combine`) *with the following syntax.*

$\texttt{KeyGen}(\lambda, d, n, \mathbb{A}) \to (\mathsf{pk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_n)$**:** *Given a security parameter λ, the maximal multiplicative depth of evaluatable circuits d, the number of parties n, and access structure $\mathbb{A}$, returns a public key* pk *and n secret keys* $\mathsf{sk}_1, \ldots, \mathsf{sk}_n$*.*

$\texttt{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$**:** *Given a public key* pk *and a message m, returns a ciphertext* ct*.*

$\texttt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k) \to \mathsf{ct}$**:** *Given a public key* pk*, a circuit $C \in \mathcal{L}_d$ and ciphertexts* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$*, returns a ciphertext* ct*.*

$\mathtt{PDec}(\mathsf{sk}_i, \mathsf{ct}) \to \mu$**:** *Given a secret key* sk_i *and a ciphertext* ct*, returns a partial decryption* μ*.*

$\mathtt{Combine}(\{\mu_i\}_{i\in S}, \mathsf{ct}) \to m$**:** *Given a set of partial decryptions* $\{\mu_i\}_{i\in S}$ *where* $S \in \mathbb{A}$*, and a ciphertext* ct*, returns a message* m*.*

Definition 15 (Ind-secure THE). *Let* $d, n \in \mathbb{N}$ *and let* $\mathcal{L}_d$ *be a class of circuits of multiplicative depth smaller or equal to* d*. Let* $\mathsf{THE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{PDec}, \mathtt{Combine})$ *be a threshold fully homomorphic encryption scheme for a class of access structures* $\mathbb{S}$ *and circuits in* $\mathcal{L}_d$*. We define the experiment* $\mathtt{Exp}_b^{\mathsf{IND\text{-}THE}}[\mathcal{A}]$*, where* $b \in \{0,1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The experiment is defined as follows:*

$$
\begin{aligned}
\mathtt{Exp}_b^{\mathsf{IND\text{-}THE}}[\mathcal{A}](\lambda): \quad & \mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}),\\
& (\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda, \mathbb{A}),\\
& S \leftarrow \mathcal{A}(\mathsf{pk}) \textit{ s.t. } S \notin \mathbb{A} \textit{ and } S \textit{ is a maximal invalid set},\\
& (m_1^{(0)}, \ldots, m_k^{(0)}, m_1^{(1)}, \ldots, m_k^{(1)}), \mathsf{st} \leftarrow \mathcal{A}([\mathsf{sk}_i]_{i\in S}),\\
& [\mathsf{ct}_i \leftarrow \mathsf{THE}.\mathtt{Enc}(\mathsf{pk}, m_i^{(b)})]_{i=1}^{k},\\
& b' \leftarrow \mathcal{A}^{\mathtt{Eval}(\mathsf{pk}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)}(\mathsf{st}, \mathsf{ct}_1, \ldots, \mathsf{ct}_n),\\
& \textbf{\textit{return}}\ b'.
\end{aligned}
$$

The $\mathtt{Eval}(\mathsf{pk}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *oracle takes as input circuit in* $C_i \in \mathcal{L}_d$ *is such that* $C_i(m_1^{(0)}, \ldots, m_k^{(0)}) = C_i(m_1^{(1)}, \ldots, m_k^{(1)})$*. The oracle computes and outputs* $\mathsf{ct}_{\mathsf{res}} \leftarrow \mathtt{Eval}(\mathsf{pk}, C_i, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *and* $\mu_j \leftarrow \mathtt{PDec}(\mathsf{sk}_j, \mathsf{ct}_{\mathsf{res}})$ *for all* $j \in [n]$*.*

The scheme THE *is said to be* λ*-bit* $\mathsf{IND\text{-}THE}$*-secure if, for any adversary* $\mathcal{A}$*, we have that* $\lambda \leq \log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}}$*, where* $\mathsf{adv}^{\mathcal{A}}$ *is defined as in Definition 10.*

Remark 1 (Impact of our results on Exact Threshold HE). Our study originates from the desire to analyze and formalize Threshold HE in the approximate setting. However, it is noteworthy that our results also have implications for schemes in the exact setting. The resemblance between these two scenarios becomes evident when considering the partial decryption phase. Due to the propagation of noise within the scheme, partial decryption shares encapsulate information about secret keys, messages encrypted by the parties, and the circuits that have been evaluated. Even in the exact setting, these shares are frequently derived using a partial decryption algorithm that avoids rounding or other non-linear post processing, resembling the behavior of decryption in the approximate setting, preserving the error magnitude. Traditionally, the prevailing approach to mitigate information leakage from partial decryption shares involves the addition of a substantial amount of noise. Hence, for Threshold Exact HE schemes where the growth of evaluation noise follows a pattern similar to the CKKS scheme (as observed in the majority of those based on BGV or BFV), our results can be directly applicable. Our established bounds on the minimal and optimal amount of noise to be added remain valid.

5.2 Recent Threshold HE definitions and Rényi divergence

Several recent papers in the literature ([DWF22],[CSS+22],[BS23]) have employed Rényi divergence as a means to achieve Threshold FHE in the exact setting. In these schemes the noise growth during evaluation follows a pattern similar to the TFHE[1]/FHEW schemes. Consequently, our results are not directly applicable. However, we emphasize some of the risks that arise when considering new security definitions for Threshold HE.

Uninstantiability of the OW-CPA to IND-CPA Transform [HHK17, BS23]. In [BS23], the authors give a construction of an exact TFHE scheme based on Rényi divergence, achieving One-Way-CPA (OW-CPA) security. Subsequently, they apply a transform, adapted from [HHK17], to achieve a IND-CPA-like security definition in the random oracle model (ROM). We provide a simplify overview of this transform.

Definition 16 (OW-CPA-security). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be a homomorphic encryption scheme with message space* $\mathcal{M}$*. We define the experiment* $\mathtt{Exp}^{\mathsf{OW\text{-}CPA}}[\mathcal{A}]$*, where* $\mathcal{A}$ *is a* **PPT** *adversary, as follow:*

$$\begin{aligned}\mathtt{Exp}^{\mathsf{OW\text{-}CPA}}[\mathcal{A}](\lambda): \quad & (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda),\\ & m \xleftarrow{\$} \mathcal{M},\\ & \mathsf{ct} \leftarrow \mathtt{Enc}(\mathsf{pk}, m),\\ & m' \leftarrow \mathcal{A}(\lambda, \mathsf{ct}),\\ & \textbf{\textit{return}}\ m'.\end{aligned}$$

The scheme HE *is said to be* OW-CPA*-secure if the advantage of every* **PPT** *adversary* $\mathcal{A}$ *is negligible in* λ*.*

Definition 17 (Transform 1 from [BS23], simplified). *Let* $\mathsf{HE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{Dec})$ *be a homomorphic encryption scheme with message space* $\mathcal{M}$*. We define the encryption scheme* $\mathsf{HE}' = (\mathtt{KeyGen}, \mathtt{Enc}', \mathtt{Dec}')$*. Let* $\mathsf{F} : \mathcal{M} \to \mathcal{M}$ *be a random oracle, then:*

$\mathtt{Enc}'(\mathsf{pk}, m \in \mathcal{M})$ **:** *Choose randomly* $x \xleftarrow{\$} \mathcal{M}$*. Compute* $\mathsf{ct}_0 \leftarrow m + \mathsf{F}(x)$ *and* $\mathsf{ct}_1 \leftarrow \mathtt{Enc}(\mathsf{pk}, x)$*. Return* $(\mathsf{ct}_0, \mathsf{ct}_1)$*.*
$\mathtt{Dec}'(\mathsf{sk}, (\mathsf{ct}_0, \mathsf{ct}_1))$ **:** *Compute* $x \leftarrow \mathtt{Dec}(\mathsf{sk}, \mathsf{ct}_1)$ *and* $m \leftarrow \mathsf{ct}_0 + \mathsf{F}(x)$*. Return* m*.*

In [HHK17] (Theorem 3.7) there is a proof in the ROM model demonstrating that any public encryption scheme achieving OW-CPA security can be shown to be IND-CPA-secure after applying the transform from Definition 17. However, we argue that this theorem, while seemingly valid in the ROM model, does not hold in the plain model, especially when applied to FHE schemes. In fact, whenever we replace the random oracle F with any hash function H, we can prove that the

[1] Here, ”T” stands for ”Torus”, not ”Threshold”.

resulting scheme is not $\mathsf{IND\text{-}CPA}$-secure if the original scheme was not already $\mathsf{IND\text{-}CPA}$-secure.

The high-level idea behind this argument is that, since H has a circuit representation (unlike F), we can derive an encryption of a message m under the original FHE scheme from an encryption of m under the modified scheme. In particular, given $\mathtt{Enc}'(m) = (\mathsf{ct}_0, \mathsf{ct}_1)$, we can compute $\mathtt{Eval}(\mathsf{pk}, \mathsf{ct}_0 + \mathsf{H}(\cdot), \mathsf{ct}_1) = \mathtt{Enc}(\mathsf{pk}, \mathsf{ct}_0 + \mathsf{H}(x)) = \mathtt{Enc}(\mathsf{pk}, m)$. This implies that an $\mathsf{IND\text{-}CPA}$ adversary against the old FHE scheme is also effective against the new one, since we can use the distinguisher on the FHE ciphertext of m. Note that we abuse notation here for simplicity, denoting a ciphertext output by $\mathtt{Eval}$ as "equal" to a ciphertext output by $\mathtt{Enc}$.

Similar conflicts between ROM and FHE can be found in the uninstantiability result ([GKW17], Theorem 7.1) regarding Fujisaki-Okamoto transform [FO99], as well as in the uninstantiability [WZ17] of Black, Rogaway, Shrimpton transform [BRS03].

For a more extensive and formal treatment on the transform used in [BS23], refer to Appendix F.

Alternative Security Definition for Threshold HE. Now, let us shift our focus to alternative security definitions recently proposed for Threshold HE schemes in the exact setting. The substitution of statistical distance with Rényi divergence has led to a reduction in the amount of noise utilized in newly proposed Threshold HE schemes, resulting in improved parameters. Unfortunately, the most commonly used security definitions for Threshold schemes require statistical simulations that cannot be accomplished using these techniques. Both [CSS⁺22] and [DWF22] introduce new security definitions tailored to their respective schemes. Nevertheless, we raise questions about the effectiveness of these definitions in accurately characterizing the security of a Threshold HE scheme.

The security definition proposed in [CSS+22] focuses solely on the security of the secret keys of the parties and does not account for the messages encrypted by the non-corrupted parties. In fact, during the so-called "Challenge phase", there is only a test of the $\mathsf{IND\text{-}CPA}$ security of the underlying HE scheme ($\mathsf{THE}.\mathtt{Enc}$) after multiple iterations of the THE scheme in the partial decryption query phase. To highlight the unsuitability of this definition for applications in Threshold HE, the paper [BS23] provides, as a counterexample, an obviously insecure scheme that satisfies this security definition. After the first version of our work, the paper [CSS+22] was updated with a different security definition that resembles the one we use in our paper (Definition 15).

Similarly, the security definition proposed in [DWF22] focuses on the indistinguishability of initial ciphertexts after a single iteration of the THE scheme. However, this alone is not enough to guarantee the safety of the underlying encrypted message. Instead, it only indirectly ensures the protection of the secret keys of non-corrupted parties.

We defer a more extensive and formal discussion on recently proposed Threshold HE definitions to Appendix G.

5.3 Multikey Homomorphic Encryption

There are many flavors of multikey homomorphic encryption in the literature. Most of the definitions differ in syntax, but the overall concept is same. The main differences between a multikey homomorphic encryption scheme and threshold homomorphic encryption schemes are (1) in MKHE the secret keys are generated by each user separately instead of by a single setup, (2) messages are encrypted with public keys of each user instead of a master public key. Consequently, the evaluation algorithm in MKHE "combines" ciphertexts with respect to different public keys into one ciphertext, whereas in threshold HE the ciphertext is already combined. Finally, (3) the decryption process in MKHE is a special case of threshold HE where all secret keys are needed to decrypt the message.

Both primitives however, share the same interface for decryption. In particular, both primitives define a partial decryption algorithm $\texttt{PDec}$. Furthermore, to the best of our knowledge, all current realizations of these primitives use a flavor of noise flooding to realize $\texttt{PDec}$. Hence it makes sense in our paper to investigate multikey homomorphic encryption together with threshold homomorphic encryption.

Below we give the syntax for multikey homomorphic encryption.

Definition 18 (Multikey Homomorphic Encryption). *Let $d \in \mathbb{N}$ and let $\mathcal{L}_d$ be a class of circuits of multiplicative depth smaller or equal to d. A multikey homomorphic encryption scheme* MKHE *on $\mathcal{L}_d$ is a tuple of five algorithms* $\mathsf{MKHE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *with the following syntax.*

$\texttt{KeyGen}(\lambda, d) \to (\mathsf{pk}, \mathsf{sk})$**:** *Given a security parameter λ, the maximal multiplicative depth of evaluatable circuits d, the algorithm returns a public key* pk *and s secret key* sk*.*

$\texttt{Enc}(\mathsf{pk}, m) \to \mathsf{ct}$**:** *Given a public key* pk *and a message m, the algorithm returns a ciphertext* ct*.*

$\texttt{Eval}(\mathsf{pk}_1, \ldots, \mathsf{pk}_n, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_n) \to \mathsf{ct}$**:** *Given a list of public keys $\mathsf{pk}_1, \ldots, \mathsf{pk}_n$, a circuit $C \in \mathcal{L}_d$ and ciphertexts $\mathsf{ct}_1, \ldots, \mathsf{ct}_n$, returns a ciphertext* ct*.*

$\texttt{PDec}(\mathsf{sk}_i, \mathsf{ct}) \to \mu$**:** *Given a secret key sk_i and a ciphertext* ct*, returns a partial decryption μ.*

$\texttt{Combine}(\{\mu_i\}_{i\in[n]}, \mathsf{ct}) \to m$**:** *Given a set of partial decryptions $\{\mu_i\}_{i\in[n]}$ and a ciphertext* ct*, returns a message m.*

Definition 19 (Multikey Ciphertext Error). *Let* $\mathsf{MKHE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *be a multikey homomorphic encryption scheme with message space $\mathcal{M}$. Furthermore, let $\mathcal{M}$ be a normed space with norm $||\cdot|| : \mathcal{M} \mapsto \mathbb{R}_{\geq 0}$. For all public/secret key pairs $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \texttt{KeyGen}(\lambda)$ where $i \in [n]$, any ciphertext* ct *in the image of* $\texttt{Eval}$ *and message $m \in \mathcal{M}$ the ciphertext error is defined as*

$$\texttt{Error}(\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{ct}, m) = ||\texttt{Combine}([\texttt{PDec}(\mathsf{sk}_i, \mathsf{ct})]_{i\in[n]}) - m||.$$

Below we give our definition of approximate correctness for multikey homomorphic encryption. Definition 21 gives our definition for indistinguishability security of multikey homomorphic encryption. Remind that this is the first security definition for multikey approximate homomorphic encryption that gives the adversary access to partial decryptions. Previously [CDKS19], only standard semantic security was considered, and security in the presence of partial decryptions were omitted.

Definition 20 (Approximate Correctness). *Let us define* $\mathsf{MKHE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *to be a multikey homomorphic encryption scheme with message space* $\mathcal{M} \subseteq \widetilde{\mathcal{M}}$ *that is a normed space with norm* $||\cdot|| : \widetilde{\mathcal{M}} \mapsto \mathbb{R}_{\geq 0}$. *Let* $\mathcal{L}$ *be the class of circuits,* $\mathcal{L}_k \subseteq \mathcal{L}$ *be the subset of circuits with* k *input wires, and let* $\texttt{Estimate} : \bigsqcup_{k \in \mathbb{N}} \mathcal{L}_k \times \mathbb{R}^k_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ *be an efficiently computable function. We call* HE *an approximate homomorphic encryption scheme if for all* $k \in \mathbb{N}$, *for all* $C \in \mathcal{L}_k$, *for all* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \texttt{KeyGen}(\lambda)$, *if* $\mathsf{ct}_1, \ldots, \mathsf{ct}_k$ *and* $m_1, \ldots, m_k$ *are such that* $\texttt{Error}(\mathsf{sk}_i, \mathsf{ct}_i, m_i) \leq t_i$, *for some* $t_1, \ldots, t_k \in \mathbb{R}_{\geq 0}$, *and* $\mathsf{ct} \leftarrow \texttt{Eval}(\mathsf{pk}_1, \ldots, \mathsf{pk}_k, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$, *then*

$$\texttt{Error}(\mathsf{sk}_1, \ldots, \mathsf{sk}_k, \mathsf{ct}, C(m_1, \ldots, m_k)) \leq \texttt{Estimate}(C, t_1, \ldots, t_k).$$

Definition 21 (Ind-secure MKHE). *Let* $d \in \mathbb{N}$ *and let* $\mathcal{L}_d$ *be a class of circuits of multiplicative depth smaller or equal to* d. *Let* $\mathsf{MKHE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *be a multikey homomorphic encryption scheme for a class circuits in* $\mathcal{L}_d$. *We define the experiment* $\texttt{Exp}_b^{\mathsf{IND\text{-}MKHE}}[\mathcal{A}]$, *where* $b \in \{0,1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The experiment is defined as follows:*

$$
\begin{aligned}
&\texttt{Exp}_b^{\mathsf{IND\text{-}MKHE}}[\mathcal{A}](\lambda): \\
&\qquad [r'_i \xleftarrow{\$} \mathcal{U}]_{i \in [n]}, \\
&\qquad [(\mathsf{sk}_i, \mathsf{pk}_i) \leftarrow \texttt{KeyGen}(\lambda, d, r'_i)]_{i \in [n]}, \\
&\qquad i^*, \mathsf{st}_1 \leftarrow \mathcal{A}(\mathsf{pk}_1, \ldots, \mathsf{pk}_n), \\
&\qquad [r_i \xleftarrow{\$} \mathcal{U}]_{i \in [n]}, \\
&\qquad (m_1^{(0)}, \ldots, m_n^{(0)}, m_1^{(1)}, \ldots, m_n^{(1)}), \mathsf{st}_2 \leftarrow \mathcal{A}(\mathsf{st}_1, [r_i, r'_i]_{i \in [n] \setminus \{i^*\}}), \\
&\qquad [\mathsf{ct}_i \leftarrow \mathsf{MKHE}.\mathtt{Enc}(\mathsf{pk}_i, m_i^{(b)}, r_i)]_{i \in [n]}, \\
&\qquad b' \leftarrow \mathcal{A}^{\texttt{Eval}(\{\mathsf{pk}_i\}_{i \in [n]}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)}(\mathsf{st}_2, \mathsf{ct}_{i^*}), \\
&\qquad \textbf{\textit{return}}\ b'.
\end{aligned}
$$

The $\texttt{Eval}(\{\mathsf{pk}_i\}_{i \in [n]}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ *oracle takes as input a circuit* $C_i \in \mathcal{L}_d$ *such that* $C_i(m_1^{(0)}, \ldots, m_n^{(0)}) = C_i(m_1^{(1)}, \ldots, m_n^{(1)})$. *The oracle computes and outputs* $\mathsf{ct}_{\mathsf{res}} \leftarrow \texttt{Eval}(\{\mathsf{pk}_i\}_{i \in [n]}, C_i, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ *and* $\mu_j \leftarrow \texttt{PDec}(\mathsf{sk}_j, \mathsf{ct}_{\mathsf{res}})$ *for all* $j \in [n]$.

The scheme MKHE *is said to be* λ*-bit* $\mathsf{IND\text{-}MKHE}$*-secure if, for any adversary* $\mathcal{A}$, *we have that* $\lambda \leq \log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}}$, *where* $\mathsf{adv}^{\mathcal{A}}$ *is defined as in Definition 10.*

An important question when stating a new security definition is whether the definition is meaningful in any way. Intuitively it seems that our definition captures what we would expect from the multikey HE. In particular, the adversary should not be able to distinguish encryptions even when given all secret keys except one, and given partial decryptions on evaluated ciphertexts. To give a more formal argument we show a multikey homomorphic encryption scheme for two keys gives us a homomorphic encryption scheme with circuit privacy.

Theorem 9. *Let* MKHE *be a* $\mathsf{IND\text{-}MKHE}$*-secure multikey homomorphic encryption scheme for* $n = 2$ *parties. We can build a homomorphic encryption scheme* HE *that is* $\mathsf{IND\text{-}CP}$*-secure.*

Proof. Let MKHE be a multikey homomorphic encryption for $n = 2$ keys. We build the HE encryption as follows. The $\mathtt{KeyGen}$ and $\mathtt{Enc}$ algorithms are the same as in MKHE. We denote the keys output by the $\mathtt{KeyGen}$ algorithm as $(\mathsf{sk}_1, \mathsf{pk}_1)$. The evaluation algorithm $\mathsf{HE}.\mathtt{Eval}$ on input $\mathsf{ct}_1 \leftarrow \mathsf{MKHE}.\mathtt{Enc}(pk_1, m)$ first samples $(\mathsf{pk}_2, \mathsf{sk}_2) \leftarrow \mathtt{KeyGen}(\lambda, d)$, encrypts the circuit C as $\mathsf{ct}_2 \leftarrow \mathtt{Enc}(\mathsf{pk}_2, C)$, and evaluates $\mathsf{ct} \leftarrow \mathsf{MKHE}.\mathtt{Eval}((\mathsf{pk}_1, \mathsf{pk}_2), U, \mathsf{ct}_1, \mathsf{ct}_2)$, where U is a circuit that takes as input a message x and another circuit F and outputs $F(x)$. Finally, the eval algorithm outputs ct and $\mu_2 \leftarrow \mathtt{PDec}(\mathsf{sk}_2, \mathsf{ct})$.

The decryption algorithm $\mathsf{HE}.\mathtt{Dec}$ runs $\mathsf{ct} \leftarrow \mathsf{MKHE}.\mathtt{Eval}((\mathsf{pk}_1,\ \mathsf{pk}_2),\ U,\ \mathsf{ct}_1,\ \mathsf{ct}_2)$, $\mu_1 \leftarrow \mathtt{PDec}(\mathsf{sk}_1, \mathsf{ct})$, and $m' \leftarrow \mathtt{Combine}(\{\mu_i\}_{i\in[n]}, \mathsf{ct})$. Note that from approximate correctness of MKHE we have that m' is close to $C(m)$, what implies that the HE is approximately correct.

Now we proceed to show circuit privacy. We construct a solver $\mathcal{S}$ that uses an adversary $\mathcal{A}$ against $\mathsf{IND\text{-}CP}$ of HE to break $\mathsf{IND\text{-}MKHE}$. The solver $\mathcal{S}$ obtains $\mathsf{pk}_1, \mathsf{pk}_2$ from the $\mathsf{IND\text{-}MKHE}$ challenger, and sends $i^* = 2$ back. The solver $\mathcal{S}$ obtains r_1 and r'_1 and passes both to the adversary. $\mathcal{A}$ responds with $(m_1, \ldots, m_k)$ and C_0 and C_1, and sends $(m_1, \ldots, m_k, C_0)$ and $(m_1, \ldots, m_k, C_1)$ to the MKHE challenger. Consequently, $\mathcal{S}$ obtains ct_1 and ct_2, and queries the $\mathtt{Eval}$ oracle on the U circuit and both ciphertexts. Denote the response of the oracle as μ_2. The solver returns μ_2 and $\mathsf{ct} \leftarrow \mathtt{Eval}(\mathsf{pk}_1, \mathsf{pk}_2, U, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ to $\mathcal{A}$. If $\mathcal{A}$ returns a bit b' the solver outputs it as its solution to the $\mathsf{IND\text{-}MKHE}$ experiment.

Note that $\mathcal{S}$ perfectly follows the $\mathsf{IND\text{-}MKHE}$ experiment. In particular, we set $(m_1^{(b)}, m_2^{(b)}) = (m_1, \ldots, m_k, C_b)$. Note that we set $m_1^{(b)} = (m_1, \ldots, m_k)$ and $m_2^{(b)} = C_b$. From the requirement on C_0 and C_1 imposed by the $\mathsf{IND\text{-}CP}$ definition we have that $C_0(m_1, \ldots, m_k) = C_1(m_1, \ldots, m_k)$, and what follows $U(C_0, m_1, \ldots, m_k) = U(C_1, m_1, \ldots, m_k)$ as required by the $\mathsf{IND\text{-}MKHE}$ experiment. To summarize, we have that the simulator $\mathcal{S}$ has advantage $\mathtt{adv}_{\mathsf{IND\text{-}CP}}[\mathcal{A}](\lambda)$ in returning the b' such that $b' = b$ and also has a running time that is similar to the running time of $\mathcal{A}$.

Remark 2 (On threshold homomorphic encryption and circuit privacy). Remind that we proved that multikey homomorphic encryption for two keys already gives us homomorphic encryption with indistinguishability circuit privacy. Note that the definition of threshold homomorphic encryption doesn't let itself use to build circuit privacy so easily. The reasons for this are that the common key generation algorithm in Definition 14 returns just one public key and all secret keys, and we cannot give the random seed to the adversary to generate its own keys honestly. Similarly, we would need to redefine the IND-THE experiment and encrypt part of the messages using honestly sampled seeds that are then passed to the adversary. Note that this modification strengthens the security notion. However, we are still unable to provide a seed for the key generation algorithm since IND-THE would be trivially broken. In this case, we would need to introduce a relaxation of our indistinguishability circuit privacy definition such that the adversary is given a secret key instead of a seed.

5.4 Achieving IND-MKHE-security for MK-CKKS

In this subsection we analyze the scheme MK-CKKS from [CDKS19] and show how to modify it to achieve IND-MKHE-security. We stress that this construction can also be adapted to other MKHE schemes that share similarities with MK-CKKS. In particular, the relevant properties we use are: the linearity of the `Combine` algorithm and the structure of extended ciphertext in $\mathcal{R}^k$, where all elements except one are uniform random in fresh encryptions. We present the algorithms of MK-CKKS, but we refer the reader to the original paper [CDKS19] for a complete description.

MK-CKKS.$\texttt{Setup}(\lambda)$: Given the security parameter λ, set $n \in \mathbb{N}$ and $Q \in \mathbb{N}$, the ring $\mathcal{R} := \mathcal{R}_Q^n$, the key distribution χ and the noise distribution ψ. Sample $a \xleftarrow{\$} \mathcal{R}_Q^n$ uniformly. Return $\mathsf{pp} = (n, Q, \chi, \psi, a)$.

MK-CKKS.$\texttt{KeyGen}(\mathsf{pp})$: Sample $s \leftarrow \chi$. Sample an error $e \leftarrow \psi$ and compute $b = -sa + e$. Return $((b, a), s)$ as $(\mathsf{pk}, \mathsf{sk})$.

MK-CKKS.$\texttt{Enc}(\mathsf{pk}, m \in \mathcal{R}_Q)$: Sample $v \leftarrow \chi$ and $e_0, e_1 \leftarrow \psi$. Denoting $\mathsf{pk} = (b, a)$, then compute $c_0 = vb_0 + m + e_0$ and $c_1 = va_0 + e_1$. Return $(c_0, c_1) \in \mathcal{R}^2$.

MK-CKKS.$\texttt{Eval}(\{\mathsf{pk}_i\}_{i\in[k]}, C, \overline{\mathsf{ct}}_1, \ldots, \overline{\mathsf{ct}}_k)$: For given ciphertexts $\overline{\mathsf{ct}}_i \in \mathcal{R}^{k_i+1}$, we denote $k \geq \max_{i\in[k]}\{k_i\}$ the number of parties involved in at least one of the $\overline{\mathsf{ct}}_i$. Rearrange the entries of each $\overline{\mathsf{ct}}_i$ and pad zeroes in empty entries to generate some ciphertexts $\overline{\mathsf{ct}}_i^*$ sharing the same secret key $\overline{\mathsf{sk}} = (1, \mathsf{sk}_1, \ldots, \mathsf{sk}_k)$. Then, the algorithm evaluates the arithmetic circuit C by means of addition and multiplication:

- **CKKS.$\texttt{Add}(\overline{\mathsf{ct}}_0, \overline{\mathsf{ct}}_1 \in \mathcal{R}^{k+1})$:** Return the entry-by-entry addition $\overline{\mathsf{ct}_0} + \overline{\mathsf{ct}_1}$.
- **CKKS.$\texttt{Mul}(\{\mathsf{pk}_i\}_{i\in[k]}, \overline{\mathsf{ct}}_0, \overline{\mathsf{ct}}_1 \in \mathcal{R}^{k+1})$:** Compute $\overline{\mathsf{ct}} = \overline{\mathsf{ct}}_1 \otimes \overline{\mathsf{ct}}_2$ and return the ciphertext $\overline{\mathsf{ct}}' \leftarrow \texttt{Relin}(\overline{\mathsf{ct}}, \{\mathsf{pk}_i\}_{i\in[k]})$. The `Relin` algorithm returns a ciphertext $\overline{\mathsf{ct}} \in \mathcal{R}^{k+1}$ encrypting $m_0 m_1$ with an error that follows the noise growth law of Lemma 5.

MK-CKKS.$\texttt{PDec}(\mathsf{sk}, \overline{\mathsf{ct}} \in \mathcal{R}^{k+1})$: Call $\overline{\mathsf{ct}}.a_i$ the component of $\overline{\mathsf{ct}}$ associated to the secret key sk. Return $\mu = \mathsf{sk} \cdot \overline{\mathsf{ct}}.a_i$. [2]

MK-CKKS.$\texttt{Combine}(\{\mu_i\}_{i\in[k]}, \overline{\mathsf{ct}} \in \mathcal{R}^{k+1})$: Return $m = \overline{\mathsf{ct}}.b + \sum_{i=1}^{k} \mu_i$.

---

[2] In the original scheme, the partial decryption algorithm already added a smudging noise $e_{\mathsf{sm}} \leftarrow \phi$. Since ϕ is not described in detail, we decided not to include it here so as to simplify the exposition of `PDec` in Algorithm 4.

The estimate function of MK-CKKS is handled similarly to CKKS but with the noise growth rule of Lemma 5.

To simplify the notation, from now on, we are going to refer to the entries of a ciphertext $\mathsf{ct} \in \mathcal{R}^{k+1}$ as $(\mathsf{ct}.b, \mathsf{ct}.a_1, \ldots, \mathsf{ct}.a_k)$. Also, when writing $\mathsf{ct}.a$, we will be referring to $(\mathsf{ct}.a_1, \ldots, \mathsf{ct}.a_k)$. We now show how to modify the `Eval` and the `PDec` algorithm in MK-CKKS to achieve IND-MKHE-security. The main idea behind `Eval′` is to re-randomize the ciphertext by adding a fresh encryption of zero for each public key pk associated to ct and then to post-process the component $\mathsf{ct}.b$ using an appropriate differential privacy mechanism M_T.

Algorithm 3: The modified evaluation MK-CKKS.`Eval′`

Data: A set of public keys $\{\mathsf{pk}_i\}_{i\in[k]}$, circuit $C \in \mathcal{L}$, a vector of ciphertexts $\mathsf{ct}_1 \in \mathcal{R}^{k+1}, \ldots, \mathsf{ct}_N \in \mathcal{R}^{k+1}$.

begin

$\mathsf{ct}_{\mathsf{res}} \leftarrow \mathtt{Eval}(\{\mathsf{pk}_i\}_{i\in[k]}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$;

For $i = 1$ to k: $\mathsf{ct}_{\mathsf{res}} \leftarrow \mathsf{ct}_{\mathsf{res}} + \mathtt{Enc}(\mathsf{pk}_i, 0)$;

$T \leftarrow \mathsf{ct}_{\mathsf{res}}.t + t_{\mathsf{fresh}}$;

$\mathsf{ct}_{\mathsf{res}}.b \leftarrow M_T(\mathsf{ct}_{\mathsf{res}}.b)$;

return $\mathsf{ct}_{\mathsf{res}}$;

Algorithm 4: The modified partial decryption MK-CKKS.`PDec′`

Data: A secret key sk, a ciphertext $\mathsf{ct} \in \mathcal{R}^{k+1}$.

begin

$\mu \leftarrow M_{\mathsf{ct}.t}(\mathtt{PDec}(\mathsf{sk}, \mathsf{ct}))$;

return μ ;

Theorem 10. *Let* MK-CKKS $= ($`Setup`, `KeyGen`, `Enc`, `Eval`, `PDec`, `Combine`$)$ *be the* MK-CKKS *multikey homomorphic encryption scheme, with plaintext space* $\mathcal{R}$ *and estimate function* `Estimate`. *Let* $q \in \mathbb{N}$. *Let* M_t *be a* ρ*-KLDP mechanism on* $\mathcal{R}$ *where* $\rho \leq 2^{-\lambda-8}/q$. *If* MK-CKKS.`Enc` *is* $(\lambda+8)$*-bit secure in the* IND-CPA *game, then* MK-CKKS *with the modified* MK-CKKS.`Eval′` *given by Algorithm 3 and with the modified* MK-CKKS.`PDec′` *given by Algorithm 4 is* λ*-bit secure in the* IND-MKHE *game where* q *is the maximum amount of oracle queries by the adversary.*

Proof. The high-level idea is as in Theorem 5. The main difference between the two proofs is the structure of the game $\mathcal{G}_1$ that has not only to protect the message choice b but also to guarantee the protection of sk_{i^*}. Also, the output of the adversary's queries is not a rLWE ciphertext anymore but it is a couple made by an extended rLWE ciphertext and a partial decryption share. This makes the tasks of upper-bounding the KL-divergence somewhat harder.

The full version of the proof is deferred to Appendix H

Analysis of the post-processing noise. We give an analysis of the lost precision when modifying the MK-CKKS scheme as in Theorem 10. We instantiate the differential privacy mechanism from Definition 8 and $\rho = 2^{-\lambda-8}/q$. Considering the output of the `Combine` algorithm and that $\mathsf{ct}.t$ is expressed in the canonical infinity norm and not in the euclidean norm, we obtain that a Gaussian noise of standard deviation $2^{7/2}\sqrt{qn2^{\lambda}}(\mathsf{ct}.t + kt_{\mathsf{fresh}})$ and $(k-1)$ Gaussian noises of standard deviation $2^{7/2}\sqrt{qn2^{\lambda}}\mathsf{ct}.t$ are added to each coordinate. The additional bits of precision lost are approximately $\lambda/2 + \log_2 \sqrt{q} + \log_2 \sqrt{n} + 7/2 + \log_2 k + \log_2 t_{\mathsf{fresh}}$.

Parameters for MK-CKKS*.* Table 2 gives parameters for instantiating MK-CKKS with k parties and with a bound on the maximum number of queries of q. For the base CKKS scheme, we consider parameters such as ring dimension and ciphertext modulus from [MA18]. In particular, we set the ring dimension to be smaller or equal to 2^{15} and the standard deviation for fresh encryption σ_{fresh} to be 3.2.

		Number of Parties			
		$k=2$	$k=2^2$	$k=2^3$	$k=2^5$
Max Queries	$q=1$	81.13	82.13	83.13	85.13
	$q=2^5$	83.64	84.64	85.64	87.64
	$q=2^{10}$	86.14	87.14	88.14	90.14

Table 2. Bits of additional Gaussian noise added in the modified MK-CKKS of Theorem 10 to achieve 128-bits of IND-MKHE-security.

5.5 Tightness of the Differential Privacy Parameters

By Theorem 10, it is possible to achieve λ bits of IND-MKHE-security by post-processing the outputs from `Eval` and `PDec` with a differentially private algorithm. Concretely we choose the Gaussian mechanism with Gaussian noise of variance $\sigma_{\mathsf{max}} \leftarrow \frac{\mathsf{ct}.t^2}{2\rho}$, where $\rho \leq 2^{-\lambda-8}/q$ is the privacy bound for ρ-KL differential privacy (Definition 7). We show that, using an appreciably smaller variance $\sigma_{\mathsf{s}} \ll \sigma_{\mathsf{max}}$, leads to the existence of an adversary that wins the IND-MKHE schemes with a non-negligible probability. In other words, we show that the noise parameters are tight when using the Gaussian mechanism, and the added Gaussian noise must be exponential in the security parameter.

The adversary that we construct exploits the noise growth in the `Eval` algorithm. This noise growth follows the rules of the following lemma.

Lemma 5 (Appendix C.3 of [CDKS19]). *Let* $\mathsf{ct}_i = \text{MK-CKKS}.\texttt{Enc}(\mathsf{pk}, m_i)$ *for* $i \in \{0, 1\}$ *and their ciphertext error be, respectively,* $\texttt{Error}(\mathsf{sk}, ct_i, m_i) = e_i$. *The ciphertext error of the sum of both ciphertexts is equal to* $e_0 + e_1$ *and the ciphertext error of their product is equal to* $m_0 e_1 + m_1 e_0 + e_0 e_1 + e_{\mathsf{mult}} + e_{\mathsf{lin}}$, *where the term* e_{mult} *depends on the parameters of the scheme and on the two ciphertexts.*

Algorithm 5: Adversary $\mathcal{A}(\lambda)$.

Data: A security parameter λ. The adversary has oracle access to $\texttt{Eval}_{\sigma_{\mathsf{s}}}$.

begin

- $\mathsf{pp} \leftarrow \texttt{Setup}(\lambda, d)$;
- $[r'_i \xleftarrow{\$} \mathcal{U}]$;
- $[(\mathsf{sk}_i, \mathsf{pk}_i) \leftarrow \texttt{KeyGen}(\mathsf{pp}, r'_i)]_{i\in[2]}$;
- $i^* \leftarrow 1$;
- $[r_i \xleftarrow{\$} \mathcal{U}]_{i\in[2]}$;
- $(m_1^{(0)}, m_2^{(0)}), (m_1^{(1)}, m_2^{(1)}) \leftarrow (0, B), (B, B)$;
- $C \leftarrow x_1 \cdot x_2 - B \cdot x_1$;
- $\mathsf{ct} \leftarrow \texttt{Enc}(\mathsf{pk}_1, m_1^{(b)}, r_1)$;
- $\tilde{\mathsf{ct}} \leftarrow \texttt{Enc}(\mathsf{pk}_2, m_2^{(b)}, r_2)$;
- $\tilde{e} \leftarrow \texttt{Dec}(\mathsf{sk}_2, \mathsf{ct}_2) - B$;
- $\mathsf{ct}_{\mathsf{res}}, \mu_1, \mu_2 \leftarrow \mathcal{O}^{\texttt{Eval}_{\sigma_{\mathsf{s}}}}(\{\mathsf{pk}_i\}_{i\in[2]}, C, \mathsf{ct}, \tilde{\mathsf{ct}})$;
- $e_{\mathsf{res}} \leftarrow \texttt{Combine}(\mu_1, \texttt{PDec}(\mathsf{sk}_2, \mathsf{ct}_{\mathsf{res}}), \mathsf{ct}_{\mathsf{res}})$;
- Choose $I \in \{0, \ldots, n-1\}$ such that $|\tilde{e}_I|$ is maximal ;
- If $|e_{\mathsf{res},I} - B\tilde{e}_I| \geq |e_{\mathsf{res},I}|$ then return 0. Otherwise output 1 ;

Theorem 11. *Let* $\sigma_{\mathsf{s}} > 0$. *Let* $\texttt{Eval}_{\sigma_{\mathsf{s}}}$ *and* $\texttt{PDec}_{\sigma_{\mathsf{s}}}$ *be the modified* MK-CKKS *algorithms we presented as Algorithm 3 and as Algorithm 4 but where the post-processing noise are sampled from* $\mathcal{N}_{\mathbb{Z}^n}(0, \sigma_{\mathsf{s}}^2 \mathsf{ct}.t^2 I_n)$. *Let* σ *be the standard deviation of the underlying rLWE error. Then there exists an adversary* $\mathcal{A}$ *(Algorithm 5)against* $\text{MK-CKKS}_{\sigma_{\mathsf{s}}}$ *in the* IND-MKHE*-security game such that* $\mathsf{adv}^{\mathcal{A}} = \Omega\left(\frac{1}{\sigma_{\mathsf{s}}^2 \sigma^2 n^3}\right)$.

Proof. Deferred to Appendix I.

Theorem 12. *If the scheme* MK-CKKS *with the modified evaluation* $\texttt{Eval}_{\sigma_{\mathsf{s}}}$ *and the modified partial decryption* $\texttt{PDec}_{\sigma_{\mathsf{s}}}$ *is* λ*-bit* IND-MKHE*-secure, then* $\sigma_{\mathsf{s}} = \Omega(2^{\lambda/2}/\sigma n^{3/2})$, *i.e. one must add at least* $\lambda/2 - \tilde{\Omega}(\sigma n^{3/2})$ *bits of additional Gaussian noise.*

Proof. By using the definition of bit security, we know that $\lambda \leq \log_2 O(\frac{T(A)}{\mathsf{adv}^A}) \leq \log_2 O(\sigma_s^2 \sigma^2 n^3)$. This means that $\sigma_s \geq 2^{\lambda/2}/(\sigma n^{3/2})$ and $\lambda/2 - \log_2 \Omega(\sigma n^{3/2}) \leq \log_2 \sigma_s$.

6 Conclusion and Open Problems

In this paper, we introduced formal models for the study of circuit privacy in the FHE approximate setting. We included the first security analysis for approximate multikey homomorphic encryption and approximate threshold homomorphic encryption that considers the knowledge of partial decryptions.

We presented a modified version of the CKKS scheme (Theorem 5) that is able to achieve λ-bit $\mathsf{IND\text{-}CP}$-security by post-processing the ciphertext with $\lambda/2 + \tilde{\mathcal{O}}(1)$ bits of noise. Additionally, we modified the $\mathsf{MK\text{-}CKKS}$ scheme (Theorem 10) to achieve λ-bit $\mathsf{IND\text{-}MKHE}$-security. We did this by post-processing the ciphertext and the decryption shares with $\lambda/2+\tilde{\mathcal{O}}(1)$ bits of noise. We proved that these bounds are essentially tight by providing adversaries for when only $\lambda/2 - \tilde{\Omega}(1)$ bits of noise are added.

Acknowledgments. This work has been partially funded/supported by the German Ministry for Education and Research through funding for the project CISPA-Stanford Center for Cybersecurity (Funding number: 16KIS0927).

References

ABFK16. Carlos Aguilar Melchor, Joris Barrier, Laurent Fousse, and Marc-Olivier Killijian. XPIR: Private information retrieval for everyone. *Proceedings on Privacy Enhancing Technologies*, 2016(2):155–174, April 2016.

ABSdV19. Mark Abspoel, Niek J. Bouman, Berry Schoenmakers, and Niels de Vreede. Fast secure comparison for medium-sized integers and its application in binarized neural networks. In Mitsuru Matsui, editor, *Topics in Cryptology – CT-RSA 2019*, volume 11405 of *Lecture Notes in Computer Science*, pages 453–472. Springer, Heidelberg, March 2019.

ACLS18. Sebastian Angel, Hao Chen, Kim Laine, and Srinath T. V. Setty. PIR with compressed queries and amortized query processing. In *2018 IEEE Symposium on Security and Privacy*, pages 962–979. IEEE Computer Society Press, May 2018.

AJJM20. Prabhanjan Ananth, Abhishek Jain, Zhengzhong Jin, and Giulio Malavolta. Multi-key fully-homomorphic encryption in the plain model. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020: 18th Theory of Cryptography Conference, Part I*, volume 12550 of *Lecture Notes in Computer Science*, pages 28–57. Springer, Heidelberg, November 2020.

ALP+21. Asra Ali, Tancrède Lepoint, Sarvar Patel, Mariana Raykova, Phillipp Schoppmann, Karn Seth, and Kevin Yeo. Communication-computation trade-offs in PIR. In Michael Bailey and Rachel Greenstadt, editors, *USENIX Security 2021: 30th USENIX Security Symposium*, pages 1811–1828. USENIX Association, August 2021.

BDGM20. Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Candidate iO from homomorphic encryption schemes. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology – EUROCRYPT 2020, Part I*, volume 12105 of *Lecture Notes in Computer Science*, pages 79–109. Springer, Heidelberg, May 2020.

BDPMW16. Florian Bourse, Rafaël Del Pino, Michele Minelli, and Hoeteck Wee. Fhe circuit privacy almost for free. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 62–89, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

BGG+18. Dan Boneh, Rosario Gennaro, Steven Goldfeder, Aayush Jain, Sam Kim, Peter M. R. Rasmussen, and Amit Sahai. Threshold cryptosystems from threshold fully homomorphic encryption. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018*, pages 565–596, Cham, 2018. Springer International Publishing.

BGGJ18. Christina Boura, Nicolas Gama, Mariya Georgieva, and Dimitar Jetchev. CHIMERA: Combining ring-LWE-based fully homomorphic encryption schemes. Cryptology ePrint Archive, Report 2018/758, 2018. https://eprint.iacr.org/2018/758.

BGPG20. Marcelo Blatt, Alexander Gusev, Yuriy Polyakov, and Shafi Goldwasser. Secure large-scale genome-wide association studies using homomorphic encryption. *Proceedings of the National Academy of Sciences*, 117(21):11608–11613, 2020.

BLR+18. Shi Bai, Tancrède Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance. *Journal of Cryptology*, 31(2):610–640, April 2018.

BP16. Zvika Brakerski and Renen Perlman. Lattice-based fully dynamic multi-key FHE with short ciphertexts. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016, Part I*, volume 9814 of *Lecture Notes in Computer Science*, pages 190–213. Springer, Heidelberg, August 2016.

BRS03. John Black, Phillip Rogaway, and Thomas Shrimpton. Encryption-scheme security in the presence of key-dependent messages. In Kaisa Nyberg and Howard M. Heys, editors, *SAC 2002: 9th Annual International Workshop on Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 62–75. Springer, Heidelberg, August 2003.

BS23. Katharina Boudgoust and Peter Scholl. Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus. Cryptology ePrint Archive, Report 2023/016, 2023. https://eprint.iacr.org/2023/016.

CCH+22. 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.

CCS19. Hao Chen, Ilaria Chillotti, and Yongsoo Song. Multi-key homomorphic encryption from TFHE. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology – ASIACRYPT 2019, Part II*, volume 11922 of *Lecture Notes in Computer Science*, pages 446–472. Springer, Heidelberg, December 2019.

CDKS19. Hao Chen, Wei Dai, Miran Kim, and Yongsoo Song. Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019: 26th Conference on Computer and Communications Security*, pages 395–412. ACM Press, November 2019.

CdWM+17. Hervé Chabanne, Amaury de Wargny, Jonathan Milgram, Constance Morel, and Emmanuel Prouff. Privacy-preserving classification on deep neural network. Cryptology ePrint Archive, Report 2017/035, 2017. https://eprint.iacr.org/2017/035.

CHK22. Henry Corrigan-Gibbs, Alexandra Henzinger, and Dmitry Kogan. Single-server private information retrieval with sublinear amortized time. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022, Part II*, volume 13276 of *Lecture Notes in Computer Science*, pages 3–33. Springer, Heidelberg, May / June 2022.

CKKS17. 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. Springer, Heidelberg, December 2017.

CLR17. Hao Chen, Kim Laine, and Peter Rindal. Fast private set intersection from homomorphic encryption. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017: 24th Conference on Computer and Communications Security*, pages 1243–1255. ACM Press, October / November 2017.

CM15. Michael Clear and Ciaran McGoldrick. Multi-identity and multi-key leveled FHE from learning with errors. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *Advances in Cryptology – CRYPTO 2015, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 630–656. Springer, Heidelberg, August 2015.

CO17. Wutichai Chongchitmate and Rafail Ostrovsky. Circuit-private multi-key FHE. In Serge Fehr, editor, *PKC 2017: 20th International Conference on Theory and Practice of Public Key Cryptography, Part II*, volume 10175 of *Lecture Notes in Computer Science*, pages 241–270. Springer, Heidelberg, March 2017.

CSS+22. Siddhartha Chowdhury, Sayani Sinha, Animesh Singh, Shubham Mishra, Chandan Chaudhary, Sikhar Patranabis, Pratyay Mukherjee, Ayantika Chatterjee, and Debdeep Mukhopadhyay. Efficient threshold FHE with application to real-time systems. Cryptology ePrint Archive, Report 2022/1625, 2022. https://eprint.iacr.org/2022/1625.

CZW17. Long Chen, Zhenfeng Zhang, and Xueqing Wang. Batched multi-hop multi-key FHE from ring-LWE with compact ciphertext extension. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017: 15th Theory of Cryptography Conference, Part II*, volume 10678 of *Lecture Notes in Computer Science*, pages 597–627. Springer, Heidelberg, November 2017.

DGBL+16. Nathan Dowlin, Ran Gilad-Bachrach, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. ICML'16, page 201–210. JMLR.org, 2016.

DMR18. Luc Devroye, Abbas Mehrabian, and Tommy Reddad. The total variation distance between high-dimensional gaussians with the same mean, 2018.

DS16. Léo Ducas and Damien Stehlé. Sanitization of FHE ciphertexts. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 294–310. Springer, Heidelberg, May 2016.

DWF22. Xiaokang Dai, Wenyuan Wu, and Yong Feng. Summation rather than concatenation: a more efficient MKFHE scheme in the plain model. Cryptology ePrint Archive, Report 2022/055, 2022. https://eprint.iacr.org/2022/055.

FO99. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Michael J. Wiener, editor, *Advances in Cryptology – CRYPTO'99*, volume 1666 of *Lecture Notes in Computer Science*, pages 537–554. Springer, Heidelberg, August 1999.

Gen09a. Craig Gentry. *A Fully Homomorphic Encryption Scheme*. PhD thesis, Stanford, CA, USA, 2009.

Gen09b. Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *41st Annual ACM Symposium on Theory of Computing*, pages 169–178. ACM Press, May / June 2009.

GH19. Craig Gentry and Shai Halevi. Compressible FHE with applications to PIR. In Dennis Hofheinz and Alon Rosen, editors, *TCC 2019: 17th Theory of Cryptography Conference, Part II*, volume 11892 of *Lecture Notes in Computer Science*, pages 438–464. Springer, Heidelberg, December 2019.

GHS12. Craig Gentry, Shai Halevi, and Nigel P. Smart. Homomorphic evaluation of the AES circuit. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology – CRYPTO 2012*, volume 7417 of *Lecture Notes in Computer Science*, pages 850–867. Springer, Heidelberg, August 2012.

GHV10. Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. i-Hop homomorphic encryption and rerandomizable Yao circuits. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 155–172. Springer, Heidelberg, August 2010.

GKW17. Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In Chris Umans, editor, *58th Annual Symposium on Foundations of Computer Science*, pages 612–621. IEEE Computer Society Press, October 2017.

GP21. Romain Gay and Rafael Pass. Indistinguishability obfuscation from circular security. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd Annual ACM Symposium on Theory of Computing*, pages 736–749. ACM Press, June 2021.

GSW13. Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 75–92. Springer, Heidelberg, August 2013.

HFH99. Bernardo A. Huberman, Matt Franklin, and Tad Hogg. Enhancing privacy and trust in electronic communities. In *Proceedings of the 1st ACM Conference on Electronic Commerce*, EC '99, page 78–86, New York, NY, USA, 1999. Association for Computing Machinery.

HHC+22. Alexandra Henzinger, Matthew M. Hong, Henry Corrigan-Gibbs, Sarah Meiklejohn, and Vinod Vaikuntanathan. One server for the price of two:

Simple and fast single-server private information retrieval. Cryptology ePrint Archive, Report 2022/949, 2022. https://eprint.iacr.org/2022/949.

HHK17. Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017: 15th Theory of Cryptography Conference, Part I*, volume 10677 of *Lecture Notes in Computer Science*, pages 341–371. Springer, Heidelberg, November 2017.

IP07. Yuval Ishai and Anat Paskin. Evaluating branching programs on encrypted data. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 575–594. Springer, Heidelberg, February 2007.

JKLS18. Xiaoqian Jiang, Miran Kim, Kristin E. Lauter, and Yongsoo Song. Secure outsourced matrix computation and application to neural networks. In David Lie, Mohammad Mannan, Michael Backes, and XiaoFeng Wang, editors, *ACM CCS 2018: 25th Conference on Computer and Communications Security*, pages 1209–1222. ACM Press, October 2018.

JVC18. Chiraag Juvekar, Vinod Vaikuntanathan, and Anantha Chandrakasan. GAZELLE: A low latency framework for secure neural network inference. In William Enck and Adrienne Porter Felt, editors, *USENIX Security 2018: 27th USENIX Security Symposium*, pages 1651–1669. USENIX Association, August 2018.

KKL+22. Taechan Kim, Hyesun Kwak, Dongwon Lee, Jinyeong Seo, and Yongsoo Song. Asymptotically faster multi-key homomorphic encryption from homomorphic gadget decomposition. Cryptology ePrint Archive, Report 2022/347, 2022. https://eprint.iacr.org/2022/347.

Klu22. Kamil Kluczniak. NTRU-ν-um: Secure fully homomorphic encryption from NTRU with small modulus. Cryptology ePrint Archive, Report 2022/089, 2022. https://eprint.iacr.org/2022/089.

KS22. Kamil Kluczniak and Leonard Schild. Fdfb: Full domain functional bootstrapping towards practical fully homomorphic encryption. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2023(1):501–537, Nov. 2022.

KSK+18. Andrey Kim, Yongsoo Song, Miran Kim, Keewoo Lee, and Jung Hee Cheon. Logistic regression model training based on the approximate homomorphic encryption. *BMC medical genomics*, 11(4):23–31, 2018.

KSK+20. Duhyeong Kim, Yongha Son, Dongwoo Kim, Andrey Kim, Seungwan Hong, and Jung Hee Cheon. Privacy-preserving approximate gwas computation based on homomorphic encryption. *BMC Medical Genomics*, 13(7):1–12, 2020.

LJLA17. Jian Liu, Mika Juuti, Yao Lu, and N. Asokan. Oblivious neural network predictions via MiniONN transformations. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017: 24th Conference on Computer and Communications Security*, pages 619–631. ACM Press, October / November 2017.

LM21. 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. Springer, Heidelberg, October 2021.

LMSS22. 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. Springer, Heidelberg, August 2022.

LTV12. Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In Howard J. Karloff and Toniann Pitassi, editors, *44th Annual ACM Symposium on Theory of Computing*, pages 1219–1234. ACM Press, May 2012.

MA18. et al. Martin Albrecht. Homomorphic encryption security standard, November 2018.

Mea86. Catherine Meadows. A more efficient cryptographic matchmaking protocol for use in the absence of a continuously available third party. In *1986 IEEE Symposium on Security and Privacy*, pages 134–134, 1986.

Mir17. Ilya Mironov. Rényi differential privacy. In Boris Köpf and Steve Chong, editors, *CSF 2017: IEEE 30st Computer Security Foundations Symposium*, pages 263–275. IEEE Computer Society Press, 2017.

MW16. Pratyay Mukherjee and Daniel Wichs. Two round multiparty computation via multi-key FHE. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016, Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 735–763. Springer, Heidelberg, May 2016.

MW18. Daniele Micciancio and Michael Walter. On the bit security of cryptographic primitives. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018, Part I*, volume 10820 of *Lecture Notes in Computer Science*, pages 3–28. Springer, Heidelberg, April / May 2018.

MW22. Samir Jordan Menon and David J. Wu. SPIRAL: Fast, high-rate single-server PIR via FHE composition. In *2022 IEEE Symposium on Security and Privacy*, pages 930–947. IEEE Computer Society Press, May 2022.

PW14. Yury Polyanskiy and Yihong Wu. *Lecture notes on information theory*. 2014.

QWW18. Willy Quach, Hoeteck Wee, and Daniel Wichs. Laconic function evaluation and applications. Cryptology ePrint Archive, Report 2018/409, 2018. https://eprint.iacr.org/2018/409.

RAD78. RL Rivest, L Adleman, and ML Dertouzos. On data banks and privacy homomorphisms. foundations of secure computation (1978), 169–180. *Search in*, 1978.

RSC+19. M. Sadegh Riazi, Mohammad Samragh, Hao Chen, Kim Laine, Kristin E. Lauter, and Farinaz Koushanfar. XONN: XNOR-based oblivious deep neural network inference. In Nadia Heninger and Patrick Traynor, editors, *USENIX Security 2019: 28th USENIX Security Symposium*, pages 1501–1518. USENIX Association, August 2019.

WZ17. Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under LWE. In Chris Umans, editor, *58th Annual Symposium on Foundations of Computer Science*, pages 600–611. IEEE Computer Society Press, October 2017.

A Improving parameters with Relaxed Bit Security

In [LMSS22], Li et al. introduced a relaxation of the bit security definition and showed how relaxed $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$ can be achieved in approximate HE schemes with less demanding amounts of noise.

Informally, a primitive is (c, s)-bit secure if, for any adversary $\mathcal{A}$, either $\mathcal{A}$ has less than 2^{-s} statistical advantage, or the running time of the attack is at least 2^c times greater than the advantage achieved.

We recall the formal definition of Relaxed Bit Security.

Definition 22 (Relaxed Bit Security, Definition 19 of [LMSS22]). *Let $\mathcal{G}$ be an indistinguishability game. Let $\mathsf{adv}^{\mathcal{A}}$ be the advantage of an adversary $\mathcal{A}$ against $\mathcal{G}$, as in Definition 10. We say that the indistinguishability game $\mathcal{G}$ is (c, s)-bit secure if, for any adversary $\mathcal{A}$, either*

$$\log_2 \frac{T(\mathcal{A})}{\mathsf{adv}^{\mathcal{A}}} \geq c \quad \textsf{or} \quad \log_2 \frac{1}{\mathsf{adv}^{\mathcal{A}}} \geq s.$$

This definition expresses two different security parameters: a computational one (c) and a statistical one (s). When choosing $s < c$, the notion of security becomes more permissive than standard bit security (Definition 10); however, this relaxation and the additional allowed statistical attacks can be accurately described and analyzed.

When using statistical techniques on a computational primitive, this finer grained definition allows to tailor the desired achieved security depending on the application. In our case, to achieve (c, s)-bits of $\mathsf{IND\text{-}MKHE}$-security, the amount of added noise depends on the statistical parameter s and not on the computational parameter c. This allows us to decrease the cost of our post-processing phase in Algorithm 3 and 4, saving around $(c - s)/2$ bits of Gaussian noise.

Theorem 13. *Let $\mathsf{MK\text{-}CKKS} = (\mathtt{Setup}, \mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{PDec}, \mathtt{Combine})$ be the $\mathsf{MK\text{-}CKKS}$ multikey homomorphic encryption scheme, with plaintext space $\mathcal{R}$ and estimate function $\mathtt{Estimate}$. Let $q \in \mathbb{N}$. Let M_t be a ρ-KLDP mechanism on $\mathcal{R}$. If $\mathsf{MK\text{-}CKKS}.\mathtt{Enc}$ is λ-bit secure in the $\mathsf{IND\text{-}CPA}$ game, then $\mathsf{MK\text{-}CKKS}$ with the modified $\mathsf{MK\text{-}CKKS}.\mathtt{Eval}'$ given by Algorithm 3 and with the modified $\mathsf{MK\text{-}CKKS}.\mathtt{PDec}'$ given by Algorithm 4 is $(\lambda - \log_2 24, \log_2(1/\rho) - \log_2 q - \log_2 24)$-bit secure in the $\mathsf{IND\text{-}MKHE}$ game where q is the maximum amount of oracle queries by the adversary.*

Proof. The proof in ([LMSS22], Appendix F) can be easily adapted to this theorem just by considering, as games $\mathcal{G}_0$ and $\mathcal{G}_1$, the games that we used in the proof of Theorem 10.

Parameters for $\mathsf{MK\text{-}CKKS}$ *with relaxed bit security.* We provide concrete parameters for instantiating $\mathsf{MK\text{-}CKKS}$ with k parties and a statistical security parameter λ_{s}. For the base CKKS scheme, we consider parameters such as ring dimension and ciphertext modulus from [MA18]. In particular, we set the ring dimension to be smaller or equal to 2^{15} and the standard deviation for fresh encryption σ_{fresh} to be 3.2.

λ_{s} \ k	2	2^2	2^3	2^5
128	86.14	87.14	88.14	90.14
112	78.14	79.14	80.14	82.14
96	70.14	71.14	72.14	74.14
80	62.14	63.14	64.14	66.14

Table 3. Bits of additional Gaussian noise added in the modified MK-CKKS of Theorem 10 to achieve $(128,\lambda_{\mathsf{s}})$-bits of IND-MKHE-security, with a bound on the maximum number of queries of 2^{10}.

The choice of the appropriate statistical security parameter strongly depends from the desired application and we refer to ([LMSS22], Subsections 4.4 and 4.5) for a more in-depth discussion on parameters choice and on Definition 22.

B Proof of Theorem 5

Theorem 5. *Let* CKKS $=$ ($\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{Dec}$) *be the CKKS approximate encryption scheme, with the normed plaintext space* $\mathcal{R}$ *and estimate function* $\texttt{Estimate}$*. Let* M_t *be a* ρ*-KLDP mechanism on* $\mathcal{R}$ *where* $\rho \leq 2^{-\lambda-7}$*. Then,* CKKS *with the modified* $\texttt{Eval}_{\mathcal{L}}$ *given by Algorithm 1 is* λ*-bit secure in the* IND-CP *game for the circuit space* $\mathcal{L}$*.*

Proof. We start by describing the two indistinguishability games.

1. $\mathcal{G}_0$: the CKKS scheme with the evaluation algorithm given by Algorithm 1 in the IND-CP game with circuit space $\mathcal{L}$.
2. $\mathcal{G}_1$: the original CKKS scheme in a variant of the IND-CP game where the challenger returns a fresh noiseless encryption (that we denote as $\texttt{Enc}_{\mathsf{n}}$) of the result $m_{\mathsf{res}} = C_0(m_1, \ldots, m_k) = C_1(m_1, \ldots, m_k)$. Furthermore, $\mathsf{ct}.b$ is post-processed with a differential privacy mechanism that uses the same noise tag obtained in the game $\mathcal{G}_0$. More formally, we consider the following

experiment:

$$\begin{aligned}\mathtt{Exp}_b^{\mathcal{G}_1}[\mathcal{A}](\lambda) : & \ r \xleftarrow{\$} \mathcal{U}, \\ & (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda; r), \\ & m_1, \ldots, m_k, C_0, C_1, \mathsf{st} \leftarrow \mathcal{A}(\lambda; r), \\ & m_{\mathsf{res}} \leftarrow C_0(m_1, \ldots, m_n), \\ & \mathsf{ct} \leftarrow \mathtt{Enc}_{\mathsf{n}}(\mathsf{pk}, m_{\mathsf{res}}), \\ & \mathsf{ct}.t \leftarrow \max_{D \in \mathcal{L}} \{\mathtt{Estimate}(D, t_{\mathsf{fresh}}, \ldots, t_{\mathsf{fresh}})\} + t_{\mathsf{fresh}}, \\ & \mathsf{ct} \leftarrow (\mathsf{ct}.a, M_{\mathsf{ct}.t}(\mathsf{ct}.b)), \\ & b' \leftarrow \mathcal{A}(\mathsf{st}, \mathsf{ct}), \\ & \textbf{return } b'.\end{aligned}$$

We want to compare these two games and, in particular, analyze the ciphertext the adversary receives from the challenger in each game. In $\mathcal{G}_0$, the ciphertext is obtained by actually homomorphically evaluating the chosen circuit and then by post-processing it with the re-randomization and with a differential privacy mechanism on the second component. In $\mathcal{G}_1$, the ciphertext is simulated by encrypting the plaintext result of the evaluation, without performing any homomorphic evaluation. We will refer to the ciphertexts returned by $\mathcal{G}_0$ and $\mathcal{G}_1$, respectively, as ct_0 and ct_1.

While assuming that $\mathsf{ct}_0.a = \mathsf{ct}_1.a = a$, we compute the norm of the difference between $\mathsf{ct}_0.b$ and $\mathsf{ct}_1.b$, which are the first components of the ciphertexts before applying the differential privacy mechanism.

$$\begin{aligned}\|\mathsf{ct}_0.b - \mathsf{ct}_1.b\| &= \|(\mathsf{ct}_0.b + a \cdot \mathsf{sk}) - (\mathsf{ct}_1.b + a \cdot \mathsf{sk})\| \\ &= \|(m + e_0) - (m)\| = \|e_0\|,\end{aligned}$$

where e_0 is the real error of the ciphertext ct_0. By definition of approximate correctness of CKKS we know that the error e_0 is smaller than the ciphertext noise tag $\mathsf{ct}_0.t$. Therefore,

$$\|\mathsf{ct}_0.b - \mathsf{ct}_1.b\| = \|e_0\| \leq \mathsf{ct}.t$$

Since we were able to bound $\|\mathsf{ct}_0.b - \mathsf{ct}_1.b\|$ with $\mathsf{ct}.t$ we can now use Definition 7 to bound their KL divergence after post-processing

$$D\left((M_{\mathsf{ct}.t}(\mathsf{ct}_0.b)|\mathsf{ct}_0.a = a) \ || \ (M_{\mathsf{ct}.t}(\mathsf{ct}_1.b)|\mathsf{ct}_1.a = a)\right) \leq \rho.$$

We now use Lemma 2 to obtain the following inequality.

$$\begin{aligned}& D(M_{\mathsf{ct}.t}(\mathsf{ct}_0.b), \mathsf{ct}_0.a || M_{\mathsf{ct}.t}(\mathsf{ct}_1.b), \mathsf{ct}_1.a) \\ & \leq \max_a D(M_{\mathsf{ct}.t}(\mathsf{ct}_0.b)|\mathsf{ct}_0.a = a || M_{\mathsf{ct}.t}(\mathsf{ct}_1.b)|\mathsf{ct}_1.a = a) + D(\mathsf{ct}_0.a || \mathsf{ct}_1.a).\end{aligned}$$

It is easy to show that $\mathsf{ct}_0.a$ is uniform random in $\mathcal{R}$ because we re-randomized it by adding $\mathtt{Enc}(\mathsf{pk}, 0)$ to ct. Also $\mathsf{ct}_1.a$ is uniform random in $\mathcal{R}$ because it is obtained as a fresh encryption. This implies that the KL divergence $D(\mathsf{ct}_0.a || \mathsf{ct}_1.a) =$

0. We have already shown that ρ is an upper bound for the remaining term, for every a. This means that the upper bound can be rewritten as follows.

$$D(M_{\mathsf{ct}.t}(\mathsf{ct}_0.b), \mathsf{ct}_0.a || M_{\mathsf{ct}.t}(\mathsf{ct}_1.b), \mathsf{ct}_1.a) \leq \rho.$$

Then, since the KL-divergence between these two indistinguishability games is smaller than a fixed value ρ and provided that $\rho/2 \leq 2^{-\lambda-8}$, we can use Theorem 4 to relate the bit security of $\mathcal{G}_0$ with the bit security of $\mathcal{G}_1$ and we obtain that $\mathcal{G}_0$ is λ-bit IND-CP-secure.

C Proof of Theorem 6

Theorem 6. *Let $k, d \in \mathbb{N}$. Let $C(x_1, \ldots, x_k)$ be a multivariate polynomial of degree smaller or equal to d. Let $B \in \mathbb{N}$ such that $\|m_i\|_{\mathsf{can}} \leq B$ for $i \in [k]$, then*

$$\mathtt{Estimate}(\mathsf{sk}, \mathsf{CKKS}.\mathtt{Eval}(\mathsf{pk}, C, [\mathsf{ct}_i]_{i\in[k]}), C([m_i]_{i\in[k]})) = d\binom{k+d}{d} O(B^d t_{\mathsf{fresh}})$$

where $\mathsf{ct}_i \leftarrow \mathtt{Enc}(\mathsf{pk}, m_i)$ for $i \in [k]$.

To prove Theorem 6 we need to recall an heuristic on e_{mult}. More accurate noise analysis on $\mathsf{CKKS}.\mathtt{Eval}$ (like [CCH+22], Heuristic 8) can be found in the literature; although, for the scope of this paper, using the following result will be enough.

Heuristic 1 (Appendix A.5 of [GHS12]) *Let w be the hamming weight of the secret key sk (i.e., the number of non-zero coordinates of sk) and n be the plaintext ring dimension. Then e_{mult} behaves like a random variable with mean zero and variance $O(wn)$.*

Proof (of Theorem 6). In this proof we denote $\mathtt{Estimate}(f(x), t_{\mathsf{fresh}})$ as $\mathtt{Est}(f(x))$. Also we omit the subscript can when using the canonical norm since it is the only norm used in this proof.

First, we want to prove that $\mathtt{Est}(x^d) = O(dB^{d-1}t_{\mathsf{fresh}})$ by strong induction. This is trivially true for $d = 1$. We now study the statement for $d > 1$. $\mathtt{Est}(x^d) = \mathtt{Estimate}(x^a \cdot x^b) = \|m^a e_b + m^b e_a + e_a e_b + e_{\mathsf{mult}}\|$ where e_a and e_b are, respectively, the resulting errors from the evaluation of the polynomials x^a and x^b, with $a + b = d$. We can bound this quantity from above by using the triangular inequality $\mathtt{Est}(x^d) \leq B^a\|e_b\| + B^b\|e_a\| + \|e_a e_b + e_{\mathsf{mult}}\|$. Using the strong inductive hypothesis $\|e_a\| = O(aB^{a-1}t_{\mathsf{fresh}})$ and $\|e_b\| = O(bB^{b-1}t_{\mathsf{fresh}})$, we can rewrite this quantity as $\mathtt{Est}(x^d) = O(B^a bB^{b-1}t_{\mathsf{fresh}} + B^b aB^{a-1}t_{\mathsf{fresh}}) + \|e_a e_b + e_{\mathsf{mult}}\|$. Since $\|e_a e_b + e_{\mathsf{mult}}\| \ll B^{d-1}$ we can just conclude that $\mathtt{Est}(x^d) = O(dB^{d-1}t_{\mathsf{fresh}})$.

We can now extend our study to monomials $x_1^{i_1} \ldots x_k^{i_k}$. We prove by induction on k that $\mathtt{Est}(x_1^{i_1} \ldots x_k^{i_k}) = O(dB^{d-1}t_{\mathsf{fresh}})$, where $d = i_1 + \cdots + i_k$. We already showed that it is true for $k = 1$. We now study the statement for $k > 1$. $\mathtt{Est}(x_1^{i_1} \ldots x_{k-1}^{i_{k-1}} \cdot x_k^{i_k}) = \|(m_1^{i_1} \ldots m_{k-1}^{i_{k-1}})e_k + m_k^{i_k} e_{k-1} + e_{k-1}e_k + e_{\mathsf{mult}}\|$, where

e_{k-1} and e_k are, respectively, the resulting error from the evaluations of the monomials $x_1^{i_1} \dots x_{k-1}^{i_{k-1}}$ and $x_k^{i_k}$. We can bound this quantity from above by using the triangular inequality $\texttt{Est}(x_1^{i_1} \dots x_k^{i_k}) \leq B^{i_1+\cdots+i_{k-1}} \|e_k\| + B^{i_k} \|e_{k-1}\| + \|e_{k-1}e_k + e_{\mathsf{mult}}\|$. Using the inductive hypothesis on e_{k-1} and e_k, we can rewrite this quantity as $\texttt{Est}(x_1^{i_1} \dots x_k^{i_k}) = O(B^{i_1+\cdots+i_{k-1}} i_k B^{i_k-1} t_{\mathsf{fresh}} + B^{i_k}(i_1 + \cdots + i_{k-1}) B^{i_1+\cdots+i_{k-1}-1} t_{\mathsf{fresh}}) + \|e_{k-1}e_k + e_{\mathsf{mult}}\|$. Since $\|e_{k-1}e_k + e_{\mathsf{mult}}\| \ll B^d$ we can just conclude that $\texttt{Est}(x_1^{i_1} \dots x_k^{i_k}) = O(dB^{d-1} t_{\mathsf{fresh}})$ where $d = i_1 + \cdots + i_k$. Finally, we analyze a generic multivariate polynomial with k variables and degree smaller or equal to d.

$$
\begin{aligned}
\texttt{Est}\Big(\sum_{\substack{0 \leq i_1 + \cdots + i_k \leq d \\ 0 < i_1, \dots, i_k \leq d}} a_{i_1,\dots,i_k} x_1^{i_1} \cdot \dots \cdot x_k^{i_k}\Big) &\leq B \binom{k+d}{d} \texttt{Est}(x_1^{i_1} \cdot \dots \cdot x_k^{i_k}) \\
&= B \binom{k+d}{d} O(dB^{d-1} t_{\mathsf{fresh}}) \\
&= d \binom{k+d}{d} O(B^d t_{\mathsf{fresh}}).
\end{aligned}
$$

D Proof of Lemma 4

Lemma 4. *Let $d \in \mathbb{N}$. Let B be the plaintext modulus and* $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, B)$*, then*

$$\mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(x^d, \mathsf{ct})) - B^d = dB^{d-1} \mathsf{ct}.e + f$$

where $\|f\|_{\mathsf{can}} = O(B^{d-1})$.

Remark 3 (On the order of operations when evaluating a polynomial). When homomorphically evaluating a polynomial, the associated noise growth does not only depend from the noise of the starting ciphertexts and the polynomial itself. In particular, in CKKS, another relevant factor is how we write the polynomial as a sequence of CKKS.Add and CKKS.Mul. For example, computing a polynomial with the *double-and-add* technique or computing it directly as $x \cdot x \cdot \dots \cdot x$ results in two different error growths. In this theorem, we analyze the direct method. Our focus on this method simplifies the derivation of the lower bound on the noise growth. We specifically consider this case because the primary objective of this theorem in our paper is to estimate the advantage of Adversary 2 who can freely choose the order of operations for the homomorphic evaluation of the polynomial.

Proof. We define e_d as the left-hand term of the equation, therefore as

$$e_d := \mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(x^d, \mathsf{ct})) - B^d.$$

In the special case of $d = 1$ we have that $e_1 = \mathsf{ct}.e$

We now prove the result by performing an induction on the degree d. This is trivially true for $d = 2$, since

$$\mathtt{Dec}(\mathsf{sk}, \mathtt{Eval}(x^2, \mathtt{Enc}(\mathsf{pk}, B))) - B^2 = 2B\mathsf{ct}.e + \mathsf{ct}.e^2 + e_{\mathsf{mult}},$$

and $f := \mathsf{ct}.e^2 + e_{\mathsf{mult}}$ is such that $\|f\|_{\mathsf{can}} = O(B)$.

We now study the statement for $d > 2$. By computing x^d as $x^{d-1} \cdot x$, and by using CKKS noise growth rule (Lemma 1), we obtain that

$$e_d = e_{d-1}B + e_1 B^{d-1} + e_{d-1}e_1 + e_{\mathsf{mult}}.$$

Using inductive hypothesis we obtain that

$$\begin{aligned} e_d &= \left((d-1)B^{d-2}\mathsf{ct}.e + f_{d-1}\right) \cdot B + \mathsf{ct}.eB^{d-1} + \left((d-1)B^{d-1}\mathsf{ct}.e + f_{d-1}\right) \mathsf{ct}.e + e_{\mathsf{mult}} \\ &= (d-1)B^{d-1}\mathsf{ct}.e + B^{d-1}\mathsf{ct}.e + f_{d-1}B + \left((d-1)B^{d-1}\mathsf{ct}.e + f_{d-1}\right) \mathsf{ct}.e + e_{\mathsf{mult}} \\ &= dB^{d-1}\mathsf{ct}.e + f_d, \end{aligned}$$

where $f_d := f_{d-1}B + \left((d-1)B^{d-1}\mathsf{ct}.e + f_{d-1}\right) \mathsf{ct}.e + e_{\mathsf{mult}}$ and $\|f_d\|_{\mathsf{can}} = O(B^{d-1})$.

E Proof of Theorem 7

Theorem 7. *Let $\sigma_{\mathsf{s}} > 0$. Let $\mathtt{Eval}^{\sigma_{\mathsf{s}}}_{\mathcal{L}_d}$ be the modified* CKKS *evaluation given by Algorithm 1 but where the post-processing noise is sampled from the discrete Gaussian $\mathcal{N}_{\mathbb{Z}^n}(0, \sigma_{\mathsf{s}}^2 T_{\max}^2 I_n)$. Then there exists an adversary $\mathcal{A}$ (Algorithm 2) against $\mathsf{CKKS}^{\sigma_{\mathsf{s}}}_{\mathcal{L}_d}$ in the* IND-CP*-game such that $\mathsf{adv}^{\mathcal{A}} = \Omega(\frac{1}{\sigma_{\mathsf{s}}^2 B^2 t_{\mathsf{fresh}}^2})$, where B is an upper bound on the messages norm modulus and t_{fresh} is the noise tag associated to freshly encrypted messages.*

Proof. We give a brief description of the high-level idea of this proof. First, the adversary computes the ciphertext errors after the homomorphic evaluation of each circuit but before the post-processing phase of the challenger. Then, we rewrite each ciphertext error after the post-processing as a sample of a Gaussian distribution, where mean and variance only depend from the chosen circuit and variables known by the challenger. Finally, we compute the statistical distance between the two Gaussian distributions linked to the two possible circuits and use this distance to obtain a lower bound on the adversary's advantage.

The adversary knows $e := \mathsf{ct}.e$, receives the resulting error e_{res} after decrypting the oracle output and can compute the errors e_0 and e_1 obtained after the standard CKKS evaluation of C_0 and C_1 on ct. The oracle computes $\mathsf{ct}_{\mathsf{res}}$ as $\mathsf{CKKS}.\mathtt{Eval}(C_b, \mathsf{ct}) + e_{\mathsf{sm}}$, where e_{sm} is sampled from $\mathcal{N}_{\mathbb{Z}^n}(0, \sigma_{\mathsf{s}}^2 T_{\max}^2 I_n)$. This means that the adversary sees e_{res} that is a sample of $\mathcal{N}_{\mathbb{Z}^n}(e_b, \sigma_{\mathsf{s}}^2 T_{\max}^2 I_n)$. Then, the adversary analyzes the polynomial $e_0 - e_1$ and chooses i as the component where the difference of the i-th coefficients of the polynomials e_0 and e_1 is maximal in absolute value. After this, the adversary focuses on the i-th coefficient of e_{res}. This is a sample of $\mathcal{N}_{\mathbb{Z}}(e_{b,i}, \sigma_{\mathsf{s}}^2 T_{\max}^2)$. Obtaining that $|e_{\mathsf{res},i} - e_{0,i}| < |e_{\mathsf{res},i} - e_{1,i}|$ is more likely when $b = 0$ while if $|e_{\mathsf{res},i} - e_{0,i}| \geq |e_{\mathsf{res},i} - e_{1,i}|$ it is at least more likely that $b = 1$ rather then $b = 0$. To analyze the adversary's advantage in distinguishing these distributions, we first study the total variation distance between them. Computing this quantity for discrete Gaussians is not an easy task, therefore we will approximate it by considering their counterparts on the real numbers. By Lemma 3 and Lemma 4 we have that

$$\Delta(\mathcal{N}(e_{0,i}, \sigma_{\mathsf{s}}^2 T_{\max}^2), \mathcal{N}(e_{1,i}, \sigma_{\mathsf{s}}^2 T_{\max}^2)) \geq \frac{1}{50} \frac{|e_{0,i} - e_{1,i}|}{\sigma_{\mathsf{s}} T_{\max}} = \Theta\left(\frac{B^{d-1}|e_i|}{\sigma_{\mathsf{s}} T_{\max}}\right).$$

Theorem 6 gives us that $T_{\max} = d(d-1)O(B^d t_{\mathsf{fresh}})$ and $|e_i| \geq 1$ with high probability. We can now rewrite the right hand term of the past equation as $\Omega(\frac{1}{\sigma_{\mathsf{s}} B t_{\mathsf{fresh}}})$. The adversary's advantage in the IND-CP game for this scheme is the square of the total variation distance we just estimated, therefore $\Omega(\frac{1}{\sigma_{\mathsf{s}}^2 B^2 t_{\mathsf{fresh}}^2})$.

F Extended analysis of the transform from [BS23]

We recall the full transform from [BS23].

Definition 23 (Transform 1 from [BS23]). *The transform is parameterized by $\delta \in \mathbb{N}$. Let* THE=($\texttt{KeyGen}$,$\texttt{Enc}$, $\texttt{Eval}$,$\texttt{PDec}$,$\texttt{Combine}$) *be a threshold homomorphic encryption scheme with message space $\mathcal{M}$. We define the threshold encryption scheme* $\mathsf{THE}' = (\texttt{KeyGen}, \texttt{Enc}', \texttt{PDec}', \texttt{Combine}')$, *with the abelian group $(\mathcal{M}', +)$ as message space. Let $\mathsf{F} : \mathcal{M}^\delta \to \mathcal{M}'$ and $\mathsf{G} : \mathcal{M}^\delta \to \{0,1\}^{2\lambda}$ be two random oracles, then*

$\texttt{Enc}'(\mathsf{pk}, m \in \mathcal{M})$ **:** *Choose randomly $x := (x_1, \ldots, x_\delta) \xleftarrow{\$} \mathcal{M}^\delta$. Compute $\mathsf{ct}_0 \leftarrow m + \mathsf{F}(x)$ and $\mathsf{ct}_j \leftarrow \texttt{Enc}(\mathsf{pk}, x_j)$ for $j \in [\delta] \setminus \{0\}$. Then compute $\mathsf{ct}_{\delta+1} \leftarrow \mathsf{G}(x)$. Return $(\mathsf{ct}_0, \ldots, \mathsf{ct}_{\delta+1})$.*

$\texttt{PDec}'(\mathsf{sk}, (\mathsf{ct}_0, \ldots, \mathsf{ct}_{\delta+1}))$ **:** *Compute $\mu := (\mu_j)$ for $j \in [\delta] \setminus \{0\}$ where $\mu_j \leftarrow \texttt{PDec}(\mathsf{sk}, \mathsf{ct}_j)$. Return μ.*

$\texttt{Combine}'((\mu_i)_{i \in \mathbb{A}}, (\mathsf{ct})_{0 \leq j \leq \delta+1})$ **:** *Compute $x'_j \leftarrow \texttt{Combine}((\mu_{ij})_{i \in \mathbb{A}}, \mathsf{ct}_j)$ for $j \in [\delta] \setminus \{0\}$. Set $x' \leftarrow (x'_1, \ldots, x'_\delta)$. If $x'_{\delta+1} = \mathsf{G}(x')$ return $\mathsf{ct}_0 - \mathsf{F}(x')$.*

As we already mentioned in Subsection 5.2, the paper ([BS23], Theorem 2) proves in the random oracle model that, given a THE scheme that is $(l, \delta) -$ OW-CPA-secure, then THE' achieves $l -$ IND-CPA security.[3] We argue that this theorem, while seems valid in the ROM oracle, does not hold in the plain model when applied to FHE schemes. In fact, whenever we replace the random oracle F and G with generic hash functions H and H' we can show that the resulting scheme THE' is not IND-CPA-secure if the original THE was not. We can use the circuit representation of H and H' to obtain ciphertexts of the encrypted message m under the original THE scheme.

In particular, given $\texttt{Enc}'(\mathsf{pk}, m) = (\mathsf{ct}_0, \ldots, \mathsf{ct}_{\delta+1})$, we can compute $\texttt{Eval}(\mathsf{ct}_0 - \mathsf{H}, \mathsf{ct}_1 \ldots, \mathsf{ct}_\delta) = \texttt{Enc}(\mathsf{pk}, m)$. This means that any adversary $\mathcal{A}$ against IND-CPA-security of THE can just be efficiently adapted to an adversary against THE' with the same advantage.

[3] The formal definition of $(l, \delta) -$ OW-CPA security can be found in Appendix G.

In [BS23] there is also another transform (Transform 2) that is applied to the OW-CPA-secure starting scheme. This second transform does not improve the security of the scheme against IND-CPA adversaries: any adversary $\mathcal{A}$ can still recover an encryption of the message m under the original scheme THE. Similarly to our previous construction, $\mathcal{A}$ can just homomorphically reverse all the steps of this second transform.

G Extended analysis of recently proposed Threshold HE definitions

As we have previously discussed in Subsection 5.2, numerous security definitions for Threshold HE have been proposed in the literature. One of the primary driving forces behind this trend is the substitution of statistical distance with Rényi divergence in the security analysis of many schemes. This transition has resulted in improved parameter choices and reduced noise levels in these schemes. Unfortunately, the most commonly used security definitions for Threshold HE schemes rely on statistical simulation and, consequently, cannot be satisfied by these new constructions.

Creating new security definitions that can be achieved with Rényi divergence and that are able to properly describe the security of a Threshold HE scheme is a delicate task. Dependencies in error propagation can be subtle, and security definitions may not comprehensively account the influence of partial decryptions on the confidentiality of the messages encrypted by the parties. As an example of this we scrutinize the security definitions provided in a previous version of [CSS+22] and in [DWF22].

The security definition used in [CSS+22] is the following. In the definition we highlight the phases defined in the original paper.

Definition 24 (Security definition for THE schemes from [CSS+22]). *Let $d, n \in \mathbb{N}$ and let $\mathcal{L}_d$ be a class of circuits of multiplicative depth smaller or equal to d. Let* THE $=$ (`KeyGen`, `Enc`, `Eval`, `PDec`, `Combine`) *be a threshold fully homomorphic encryption scheme for a class of access structures $\mathbb{S}$ and circuits in $\mathcal{L}_d$. We define the experiment $\mathtt{Exp}'_b[\mathcal{A}]$, where $b \in \{0, 1\}$ is a bit and $\mathcal{A}$ is an* *adversary. The experiment is defined as follows:*

$$
\begin{aligned}
\texttt{Exp}'_b[\mathcal{A}](\lambda):&\\
&\#\ \textit{Initialization phase}\\
&\mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}),\\
&(\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \texttt{KeyGen}(\lambda, \mathbb{A}),\\
&S \leftarrow \mathcal{A}(\mathsf{pk}) \ \textit{s.t.}\ S \notin \mathbb{A} \ \textit{and } S \textit{ is a maximal invalid set},\\
&\#\ \textit{Honest Encryption query phase}\\
&(m_0, \ldots, m_k), \mathsf{st} \leftarrow \mathcal{A}([\mathsf{sk}_i]_{i \in S}),\\
&[\mathsf{ct}_i \leftarrow \mathsf{THE}.\texttt{Enc}(\mathsf{pk}, m_i)]_{i=1}^{k},\\
&\#\ \textit{Partial Decryption query phase}\\
&(\mathsf{ct}_{\mathsf{res}}^{(i)}, \{\mu_j^{(i)}\}_{j \in [n]})_{i \in [q]} \leftarrow \mathcal{A}^{\mathcal{O}(\mathsf{pk}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)}(\mathsf{st}, \mathsf{ct}_1, \ldots, \mathsf{ct}_n),\\
&\#\ \textit{Challenge Phase}\\
&m_0, m_1 \leftarrow \mathcal{A}(\mathsf{st}, \{\mathsf{ct}_{\mathsf{res}}^{(i)}, \mu_j^{(i)}\}_{j \in [n], i \in [q]}),\\
&\mathsf{ct} \leftarrow \mathsf{THE}.\texttt{Enc}(\mathsf{pk}, m_b),\\
&b' \leftarrow \mathcal{A}(\mathsf{st}, \mathsf{ct}),\\
&\textbf{\textit{return}}\ b'.
\end{aligned}
$$

The $\mathcal{O}(\mathsf{pk}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *oracle takes as input circuits in* $C_i \in \mathcal{L}_d$ *. The oracle computes and outputs* $\mathsf{ct}_{\mathsf{res}} \leftarrow \texttt{Eval}(\mathsf{pk}, C_i, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *and* $\mu_j \leftarrow \texttt{PDec}(\mathsf{sk}_j, \mathsf{ct}_{\mathsf{res}})$ *for all* $j \in [n]$*. The scheme is secure under this definition if the advantage of all PPT adversary is negligible in* λ*.*

This security definition exclusively focuses on the safety of the secret keys of non-corrupted parties and does not address the confidentiality of the message encrypted by them. This limitation becomes evident during the challenge phase, where the game assesses only the IND-CPA security of the underlying HE scheme (THE.Enc) after multiple iterations of the THE scheme in the partial decryption query phase. To illustrate the inadequacy of this definition for applications in Threshold HE, the paper [BS23] provides an example of an obviously insecure scheme that satisfies this security definition. In this counterexample, the Eval algorithm is modified to also output an encryption of the first message, i.e. $\texttt{Eval}'(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k) := (\texttt{Eval}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_n), \mathsf{ct}_1)$. As a result, after executing the Combine algorithm, all the parties learn m_1, the message originally encrypted from the first party.

We now analyze the security definition utilized in [DWF22]. The paper focuses mainly on achieving "indistinguishability on the initial ciphertexts". Although the security definition is originally designed for Multikey HE schemes, it can be easily adapted to Threshold HE schemes.

Definition 25 (Security definition for THE schemes from [DWF22], adapted). *Let* $d, n \in \mathbb{N}$ *and let* $\mathcal{L}_d$ *be a class of circuits of multiplicative depth smaller or equal to* d*. Let* THE = (KeyGen*,* Enc*,* Eval*,* PDec*,* Combine) *be a* *threshold fully homomorphic encryption scheme for a class of access structures* $\mathbb{S}$*, with ciphertext space* $\mathcal{C}$ *and circuits in* $\mathcal{L}_d$*. We define the indistinguishability game* $\texttt{Exp}'_b[\mathcal{A}]$*, where* $b \in \{0,1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The experiment is defined as follows:*

$$
\begin{aligned}
\texttt{Exp}'_b[\mathcal{A}](\lambda): \quad & \mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}),\\
& (\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \texttt{KeyGen}(\lambda, \mathbb{A}),\\
& S \leftarrow \mathcal{A}(\mathsf{pk}) \textit{ s.t. } S \notin \mathbb{A} \textit{ and } S \textit{ is a maximal invalid set},\\
& (m, \mathsf{st}) \leftarrow \mathcal{A}([\mathsf{sk}_i]_{i \in S}, \mathsf{pk}),\\
& \textit{if b=0 then: } \mathsf{ct} \leftarrow \texttt{Enc}(\mathsf{pk}, m),\\
& \textit{if b=1 then: } \mathsf{ct} \xleftarrow{\$} \mathcal{C},\\
& b' \leftarrow \mathcal{A}(\mathsf{ct}, \mathsf{st}),\\
& \textbf{\textit{return}}\ b'.
\end{aligned}
$$

This scheme is secure under this definition if the advantage of all **PPT** *adversaries is negligible in* λ*.*

This definition does not consider the security of the encrypted message m within the scheme. An analogous counterexample to the one we mentioned earlier for the scheme in [CSS+22] shows how obviously insecure schemes can still satisfy this security definition.

In the latter part of this appendix, we aim to describe other recently proposed security definitions and compare them with the ones employed in this paper. Firstly, we describe the security definition from the updated version of [CSS+22].

Definition 26 (Updated security definition from [CSS+22]). *Let* $d, n \in \mathbb{N}$ *and let* $\mathcal{L}_d$ *be a class of circuits of multiplicative depth smaller or equal to* d*. Let* $\mathsf{THE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *be a threshold fully homomorphic encryption scheme for a class of access structures* $\mathbb{S}$ *and circuits in* $\mathcal{L}_d$*. We define the experiment* $\texttt{Exp}'_b[\mathcal{A}]$*, where* $b \in \{0,1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The* *experiment is defined as follows:*

$$
\begin{aligned}
\texttt{Exp}'_b[\mathcal{A}](\lambda):\quad & \\
& \textsf{\# Initialization phase} \\
& \mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}), \\
& (\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \texttt{KeyGen}(\lambda, \mathbb{A}), \\
& S \leftarrow \mathcal{A}(\mathsf{pk}) \textit{ s.t. } S \notin \mathbb{A} \textit{ and } S \textit{ is a maximal invalid set}, \\
& \textsf{\# Honest Encryption query phase} \\
& (m_1^{(0)}, \ldots, m_k^{(0)}, m_1^{(1)}, \ldots, m_k^{(1)}), \mathsf{st} \leftarrow \mathcal{A}([\mathsf{sk}_i]_{i\in S}), \\
& [\mathsf{ct}_i \leftarrow \mathsf{THE}.\texttt{Enc}(\mathsf{pk}, m_i^{(b)})]_{i=1}^{k}, \\
& \textsf{\# Partial Decryption query phase} \\
& \{C_i\}_{i\in[Q]} \leftarrow \mathcal{A}(\mathsf{st}, \mathsf{ct}_1, \ldots, \mathsf{ct}_k), \\
& [\mathsf{ct}_{\mathsf{res},i} \leftarrow \mathsf{THE}.\texttt{Eval}(\mathsf{pk}, C_i, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)]_{i\in[Q]}, \\
& [\mu_{i,j} \leftarrow \mathsf{THE}.\texttt{PDec}(\mathsf{sk}_j, \mathsf{ct}_{\mathsf{res},i})]_{i\in[Q], j\notin S}, \\
& b' \leftarrow \mathcal{A}(\mathsf{st}, \{\mathsf{ct}_{\mathsf{res},i}, \mu_{i,j}\}_{i\in[Q], j\notin S}), \\
& \textbf{\textit{return}}\ b'.
\end{aligned}
$$

The adversary can send to the challenger a polynomial number $Q = \texttt{poly}(\lambda)$ *of circuits such that* $C_i(m_1^{(0)}, \ldots, m_k^{(0)}) = C_i(m_1^{(1)}, \ldots, m_k^{(1)})$*, for every* $i \in [Q]$*.*

The definition used in the updated version of [CSS+22] resembles Definition 15. The primary distinction is that, in the former, the adversary must select all the circuits that the challenger evaluates at the same time, prior to gaining any information on the results. In the latter, however, the adversary can adaptively choose the circuits to submit to the challenger.

We now describe the IND-CPA-style security definition from [BS23].

Definition 27 **(l – IND-CPA security for THE schemes from [BS23]).** *Let* $d, n \in \mathbb{N}$ *and let* $\mathcal{L}_d$ *be a class of circuits of multiplicative depth smaller or equal to* d*. Let* $\mathsf{THE} = (\texttt{KeyGen}, \texttt{Enc}, \texttt{Eval}, \texttt{PDec}, \texttt{Combine})$ *be a threshold fully homomorphic encryption scheme for a class of access structures* $\mathbb{S}$ *and circuits in* $\mathcal{L}_d$*. We define the experiment* $\texttt{Exp}'_b[\mathcal{A}]$*, where* $b \in \{0, 1\}$ *is a bit and* $\mathcal{A}$ *is an adversary. The experiment is defined as follows:*

$$
\begin{aligned}
\texttt{Exp}'_b[\mathcal{A}](\lambda):\quad & \mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}), \\
& (\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \texttt{KeyGen}(\lambda, \mathbb{A}), \\
& S \leftarrow \mathcal{A}(\mathsf{pk}) \textit{ s.t. } S \notin \mathbb{A} \textit{ and } S \textit{ is a maximal invalid set}, \\
& b' \leftarrow \mathcal{A}^{\texttt{OEnc},\texttt{OChallEnc},\texttt{OPDec}}(\mathsf{st}, \mathsf{pk}, [\mathsf{sk}_i]_{i\in S}), \\
& \textbf{\textit{return}}\ b'.
\end{aligned}
$$

In this definition, the oracle $\texttt{OEnc}(m)$ *can be used to obtain a fresh encryption of a message* m*, the oracle* $\texttt{OChallEnc}(m_0, m_1)$ *serves to acquire an encryption of*

m_b, and the oracle $\mathtt{OPDec}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *can be used to obtain partial decryption shares from non-corrupted parties in the form of* $\mathtt{PDec}(\mathsf{sk}_j, \mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k))$*. The total calls to the oracles are bounded by l and the last oracle returns an output only if it is queried by using ciphertexts obtained from the other oracles and if $C(m_1^{(0)}, \ldots, m_k^{(0)}) = C(m_1^{(1)}, \ldots, m_k^{(1)})$, where $m_i^{(0)}$ and $m_i^{(1)}$ are the possible encrypted messages, depending on the bit, in* ct_i*.*

The main difference with the previous definitions is that the adversary can query these oracles in any desired order without necessarily follow the standard sequential execution of the threshold scheme. Another meaningful difference is that the $\mathtt{OPDec}$ oracle does not return the ciphertext $\mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$.

Finally, another noteworthy definition is the OW-CPA security for THE schemes from [BS23].

Definition 28 ((l, v) – OW-CPA security for THE schemes from [BS23]). *Let $d, n \in \mathbb{N}$ and let $\mathcal{L}_d$ be a class of circuits of multiplicative depth smaller or equal to d. Let* $\mathsf{THE} = (\mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{PDec}, \mathtt{Combine})$ *be a threshold fully homomorphic encryption scheme for a class of access structures $\mathbb{S}$ and circuits in $\mathcal{L}_d$. We define the experiment* $\mathtt{Exp}'[\mathcal{A}]$*, where $\mathcal{A}$ is an adversary. The experiment is defined as follows:*

$$\begin{aligned}
\mathtt{Exp}'[\mathcal{A}](\lambda): \quad & \mathbb{A} \leftarrow \mathcal{A}(\lambda, d, n, \mathbb{S}), \\
& (\mathsf{sk}_1, \ldots, \mathsf{sk}_n, \mathsf{pk}) \leftarrow \mathtt{KeyGen}(\lambda, \mathbb{A}), \\
& S \leftarrow \mathcal{A}(\mathsf{pk}) \textit{ s.t. } S \notin \mathbb{A} \textit{ and } S \textit{ is a maximal invalid set}, \\
& (m', j) \leftarrow \mathcal{A}^{\mathtt{OEnc}, \mathtt{OChallEnc}, \mathtt{OPDec}}(\mathsf{st}, \mathsf{pk}, [\mathsf{sk}_i]_{i \in S}), \\
& \textbf{\textit{return }} m_j = m'.
\end{aligned}$$

In this definition, the oracle $\mathtt{OEnc}(m)$ *can be used to obtain a fresh encryption of a message m, the oracle* $\mathtt{OChallEnc}()$ *serves to acquire an encryption of a random $m \xleftarrow{\$} \mathcal{M}$, and the oracle* $\mathtt{OPDec}(C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k)$ *can be used to obtain partial decryption shares from non-corrupted parties as* $\mathtt{PDec}(\mathsf{sk}_j, \mathtt{Eval}(\mathsf{pk}, C, \mathsf{ct}_1, \ldots, \mathsf{ct}_k))$*. In this definition m_j is the message encrypted in the ciphertext obtained from the j-th call to the* $\mathtt{OChallEnc}$ *oracle. The total calls to the oracles are bounded by l and the last oracle returns an output only if it is queried by using ciphertexts obtained from the other oracles and if for all the challenge messages, the conditional min-entropy has always decreased for an amount smaller than v after having learned the circuit evaluation.*

H Proof of Theorem 10

Theorem 10. *Let* $\mathsf{MK\text{-}CKKS} = (\mathtt{Setup}, \mathtt{KeyGen}, \mathtt{Enc}, \mathtt{Eval}, \mathtt{PDec}, \mathtt{Combine})$ *be the* MK-CKKS *multikey homomorphic encryption scheme, with plaintext space $\mathcal{R}$ and estimate function* $\mathtt{Estimate}$*. Let $q \in \mathbb{N}$. Let M_t be a ρ-KLDP mechanism on $\mathcal{R}$ where $\rho \leq 2^{-\lambda-8}/q$. If* $\mathsf{MK\text{-}CKKS}.\mathtt{Enc}$ *is $(\lambda+8)$-bit secure in the* IND-CPA *game, then* MK-CKKS *with the modified* $\mathsf{MK\text{-}CKKS}.\mathtt{Eval}'$ *given by Algorithm 3* *and with the modified* $\mathsf{MK\text{-}CKKS}.\mathtt{PDec}'$ *given by Algorithm 4 is* λ*-bit secure in the* $\mathsf{IND\text{-}MKHE}$ *game where* q *is the maximum amount of oracle queries by the adversary.*

Proof. We start by describing the two indistinguishability games.

1. $\mathcal{G}_0$: the $\mathsf{MK\text{-}CKKS}$ scheme with the modified algorithms given by Algorithm 3 and Algorithm 4 in the $\mathsf{IND\text{-}MKHE}$-security game with a bound of maximum q queries.
2. $\mathcal{G}_1$: the original $\mathsf{MK\text{-}CKKS}$ scheme in a variant of the $\mathsf{IND\text{-}MKHE}$-security game with a bound of maximum q queries and the modified oracle $\mathtt{Eval}'$. The oracle $\mathtt{Eval}'(\{\mathsf{pk}_i\}_{i=1}^n, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ takes as input a circuit $C_i \in \mathcal{L}_d$ such that $C_i(m_1^{(0)}, \ldots, m_n^{(0)}) = C_i(m_1^{(1)}, \ldots, m_n^{(1)})$, and behaves in the following way. When writing $\mathtt{Enc}_\mathsf{n}(\mathsf{pk}, m)$ we denote a noiseless encryption of m.

$$
\begin{aligned}
&\mathtt{Eval}'(\{\mathsf{pk}_i\}_{i\in[n]}, \cdot, \mathsf{ct}_1, \ldots, \mathsf{ct}_n) : \\
&\qquad m_\mathsf{res} \leftarrow C(m_1^{(0)}, \ldots, m_n^{(0)}), \\
&\qquad \mathsf{ct}_\mathsf{res} \leftarrow \mathtt{Enc}(\mathsf{pk}_{i^*}, 0) + \sum_{j\in[n]\smallsetminus\{i^*\}} \mathtt{Enc}_\mathsf{n}(\mathsf{pk}_j, 0), \\
&\qquad \mathsf{ct}_\mathsf{res}.t \leftarrow \mathtt{Estimate}(C, \mathsf{ct}_1.t, \ldots, \mathsf{ct}_n.t) + (k+1)t_\mathsf{fresh}, \\
&\qquad \mu_{i^*} \leftarrow M_{\mathsf{ct}_\mathsf{res}.t}(\mathsf{ct}_\mathsf{res}.b - \sum_{j\neq i^*} \mathsf{sk}_j \cdot \mathsf{ct}_\mathsf{res}.a_j), \\
&\qquad [\mu_i \leftarrow \mathsf{sk}_i \cdot \mathsf{ct}_\mathsf{res}.a_i]_{i\neq i^*}, \\
&\qquad \mathsf{ct}_\mathsf{res}.b \leftarrow M_{\mathsf{ct}_\mathsf{res}.t}(\mathsf{ct}_\mathsf{res}.b + m_\mathsf{res}), \\
&\qquad \mathbf{return}(\mathsf{ct}_\mathsf{res}, [\mu_i]_{i\in[n]}).
\end{aligned}
$$

In $\mathcal{G}_0$, the ciphertext ct_res and the decryption shares μ_i are obtained by homomorphically evaluating the circuit C on the input ciphertexts and partially decrypting the resulting ciphertext. After computing them, we perform some post-processing with a re-randomization on ct_res and with a differential privacy mechanism on both. In $\mathcal{G}_1$, the ciphertext ct_res and the decryption shares μ_i are simulated, and they do not depend from the input ciphertexts, from b or from the secret key of the non-corrupted party i^*. ct_res is a fresh, random encryption of m_res, and the share μ_{i^*} is obtained without using sk_{i^*}.

To simplify the notation in this proof, we will denote $\mathsf{ct}_\mathsf{res}^{\mathcal{G}_0}$ as ct_0, $\mathsf{ct}_\mathsf{res}^{\mathcal{G}_1}$ as ct_1 and $\mathsf{ct}_\mathsf{res}^{\mathcal{G}_0}.t$ as t.

While assuming that $\mathsf{ct}_0.a = \mathsf{ct}_1.a = a$, we compute the norm of the difference between $\mathsf{ct}_0.b$ and $\mathsf{ct}_1.b$, which are the first components of the ciphertexts before applying the differential privacy mechanism.

$$
\begin{aligned}
\|\mathsf{ct}_0.b - \mathsf{ct}_1.b\| &= \|(\mathsf{ct}_0.b + a\cdot(\mathsf{sk}_1, \ldots, \mathsf{sk}_k)) - (\mathsf{ct}_1.b + a\cdot(\mathsf{sk}_1, \ldots, \mathsf{sk}_k))\| \\
&= \|(m+e_0) - (m+e_1)\| = \|e_0 - e_1\| \leq t + t_\mathsf{fresh},
\end{aligned}
$$

We will denote $t + t_\mathsf{fresh}$ as T for the rest of the proof. Since we were able to bound $\|\mathsf{ct}_0.b - \mathsf{ct}_1.b\|$ with T, we can now use Definition 7 to bound their KL divergence after post-processing.

$$D(M_T(\mathsf{ct}_0.b)|\mathsf{ct}_0.a = a||M_T(\mathsf{ct}_1.b)|\mathsf{ct}_1.a = a) \le \rho.$$

We repeat the same reasoning with decryption shares. To simplify the notation in this proof, we will denote $\mu_j^{\mathcal{G}_b}$ with $\mu_{j,b}$. While assuming that $\mathsf{ct}_0.b = \mathsf{ct}_1.b = b$ and $\mathsf{ct}_0.a = \mathsf{ct}_1.a = a$ are chosen, we compute the norm of the difference between μ_{0,i^*} and μ_{1,i^*}, which are the decryption shares before applying the differential privacy mechanism.

$$\|\mu_{0,i^*} - \mu_{1,i^*}\| = \|(a_{i^*} \cdot \mathsf{sk}_{i^*}) - (b - \sum_{j \ne i^*} a_j \cdot \mathsf{sk}_j)\| = \|e_0\| \le t.$$

This implies, thanks to Definition 7, that

$$D(M_t(\mu_{0,i^*})|(\mathsf{ct}_0.b = b, \mathsf{ct}_0.a = a)||M_t(\mu_{1,i^*})|(\mathsf{ct}_1.b = b, \mathsf{ct}_1.a = a)) \le \rho$$

From this point forward, we often use the notation $D_a(\mathcal{X}||\mathcal{Y})$ when referring to $D(\mathcal{X}|(\mathsf{ct}.a = a)||\mathcal{Y}|(\mathsf{ct}.a = a))$. We now use Lemma 2 to obtain the following inequality.

$$\begin{aligned} &D(M_t(\mu_{0,i^*}), M_T(\mathsf{ct}_0.b), \mathsf{ct}_0.a||M_t(\mu_{1,i^*}), M_T(\mathsf{ct}_1.b), \mathsf{ct}_1.a) \\ &\le \max_a D_a(M_t(\mu_{0,i^*}), M_T(\mathsf{ct}_0.b)||M_t(\mu_{1,i^*}), M_T(\mathsf{ct}_1.b)) + D(\mathsf{ct}_0.a||\mathsf{ct}_1.a) \end{aligned}$$

It is easy to show that $\mathsf{ct}_0.a_i$ are uniform random in $\mathcal{R}$ for each $i \in [k]$ because we re-randomized each entry by adding $\mathtt{Enc}(\mathsf{pk}_i, 0)$ to ct_0. This is also true for $\mathsf{ct}_1.a_i$ for each $i \ne i^*$. We can also say that $\mathsf{ct}_1.a_{i^*}$ is uniform random in $\mathcal{R}$ because it is obtained as a fresh encryption of 0. This implies that the KL divergence $D(\mathsf{ct}_0.a||\mathsf{ct}_1.a) = 0$. We can now apply Lemma 2 and obtain the following inequality.

$$\begin{aligned} &D(M_t(\mu_{0,i^*}), M_T(\mathsf{ct}_0.b), \mathsf{ct}_0.a||M_t(\mu_{1,i^*}), M_T(\mathsf{ct}_1.b), \mathsf{ct}_1.a) \\ &\le \max_{b,a} D_{b,a}(M_t(\mu_{0,i^*})||M_t(\mu_{1,i^*})) + \max_a D_a(M_T(\mathsf{ct}_0.b)||M_T(\mathsf{ct}_1.b)) \end{aligned}$$

We have already shown that ρ is an upper bound for each of these two terms, for every a and b. This means that the upper bound can be rewritten as follows.

$$D(M_t(\mu_{0,i^*}), M_T(\mathsf{ct}_0.b), \mathsf{ct}_0.a||M_t(\mu_{1,i^*}), M_T(\mathsf{ct}_1.b), \mathsf{ct}_1.a) \le 2\rho$$

Then, we use Theorem 4 with $\mathcal{X}_\theta$ defined as a query to the oracle $\mathtt{Eval}$ of $\mathcal{G}_0$ and $\mathcal{Y}_\theta$ as a query to the oracle $\mathtt{Eval}'$.

$$\mathsf{adv}^{\mathcal{A}} \le \frac{q}{2} \max_{\theta \in [q]} D(\mathcal{X}_\theta||\mathcal{Y}_\theta) \le \frac{q}{2}(2\rho) = q\rho.$$

We conclude the proof by studying the bit security of $\mathcal{G}_1$. In the first phase of the game the adversary receives a rLWE encryption of $m_{i^*}^{(b)}$ under sk_{i^*} and then receives a fresh encryption of zero under sk_{i^*} for a polynomial number of times q. This implies that, if MK-CKKS is $(\lambda+8)$-bit secure in the IND-CPA game, then $\mathcal{G}_1$ is also $(\lambda + 8)$-bit secure. Provided that $q\rho \le 2^{-(\lambda+8)}$, we can finally relate the bit security of $\mathcal{G}_0$ with the bit security of $\mathcal{G}_1$, using Lemma 3 and obtain that $\mathcal{G}_0$ is λ-bit secure in the IND-MKHE security game with maximum q oracle queries.

I Proof of Theorem 11

Theorem 11. *Let* $\sigma_{\mathsf{s}} > 0$*. Let* $\mathtt{Eval}_{\sigma_{\mathsf{s}}}$ *and* $\mathtt{PDec}_{\sigma_{\mathsf{s}}}$ *be the modified* MK-CKKS *algorithms we presented as Algorithm 3 and as Algorithm 4 but where the post-processing noise are sampled from* $\mathcal{N}_{\mathbb{Z}^n}(0, \sigma_{\mathsf{s}}^2 \mathsf{ct}.t^2 I_n)$*. Let* σ *be the standard deviation of the underlying rLWE error. Then there exists an adversary* $\mathcal{A}$ *(Algorithm 5) against* $\mathsf{MK\text{-}CKKS}_{\sigma_{\mathsf{s}}}$ *in the* IND-MKHE*-security game such that* $\mathsf{adv}^{\mathcal{A}} = \Omega\left(\frac{1}{\sigma_{\mathsf{s}}^2 \sigma^2 n^3}\right)$.

Proof. The high-level idea is as in the proof of Theorem 7. The main difference between the two proofs is that the adversary cannot compute the error after the homomorphic evaluation of the circuit because it depends from the encrypted message of the non-corrupted party. Nonetheless, using the ring structure of $\mathcal{R}$ and the circuit $x_1 x_2 - B x_2$, we are still able to rewrite the error as a sample of a Gaussian distribution where mean and variance only depend from the encrypted message and variables known by the challenger. Finally, we compute the statistical distance between the two Gaussian distributions linked to the two possible messages and use this distance to obtain a lower bound on the adversary's advantage.

The adversary knows the exact error $\tilde{e} := \tilde{\mathsf{ct}}.e$ and obtains the resulting error e_{res} after post-processing. We denote as $e \leftarrow \mathcal{N}_{\mathbb{Z}^n}(0, \sigma^2 I_n)$ the exact error of ct. Recalling the error growth rule of MK-CKKS, we can estimate the two possible outputs for $b \in \{0,1\}$. The resulting error after computing $x \cdot y$ is equal to $e\tilde{e} + m_b \tilde{e} + Be + e_{\mathsf{mult}}$. When subtracting $B \cdot x$ in the evaluation, we also subtract Be from the error and we obtain that the error in the output of the oracle $\mathsf{ct}_{\mathsf{res}}$ is $e\tilde{e} + m_b\tilde{e} + e_{\mathsf{mult}} + e_{\mathsf{sm}}^{(1)}$ where the $e_{\mathsf{sm}}^{(1)}$ is the post-processing noise of $\mathtt{Eval}_{\sigma_{\mathsf{s}}}$. When we compute the decryption of $\mathsf{ct}_{\mathsf{res}}$ using the $\mathtt{Combine}$ algorithm, we obtain that the result is

$$e_{\mathsf{res}} = e\tilde{e} + m_b \tilde{e} + e_{\mathsf{mult}} + e_{\mathsf{sm}}^{(1)} + e_{\mathsf{sm}}^{(2)},$$

where $e_{\mathsf{sm}}^{(2)}$ is the post-processing noise of $\mathtt{PDec}_{\sigma_{\mathsf{s}}}$. Referring to the i-th coefficient of e and $\tilde{e}$ as e_i and as $\tilde{e}_i$, we can rewrite e_{res} as follows.

$$\begin{aligned} e_{\mathsf{res}} &= \sum_{i=0}^{n-1} \left(\sum_{j=0}^{i} \tilde{e}_j e_{i-j} - \sum_{j=i}^{n-1} \tilde{e}_j e_{n+i-j} + m_b \tilde{e}_i \right) x^i + e_{\mathsf{mult}} + e_{\mathsf{sm}}^{(1)} + e_{\mathsf{sm}}^{(2)} \\ &:= \sum_{i=0}^{n-1} E_i x^i + e_{\mathsf{mult}} + e_{\mathsf{sm}}^{(1)} + e_{\mathsf{sm}}^{(2)} \end{aligned}$$

The adversary analyzes the polynomial $\tilde{e}$ and chooses I as the component where the absolute value $|\tilde{e}_I|$ is maximal. We now focus on the I-th coefficient of e_{res} and, in particular, on E_I. The term E_I is an affine combination of $\{e_i\}_{i=0}^{n-1}$ that are independently sampled from $\mathcal{N}_{\mathbb{Z}}(0, \sigma^2)$ with coefficients that are known to the adversary. This implies that E_I is a sample from the Gaussian $\mathcal{N}_{\mathbb{Z}}(m_b \tilde{e}_I,$

$\sum_{i=0}^{n-1} \tilde{e_i}^2\sigma^2$). To estimate the total variation distance, we assume that e_{mult} and e_{lin} are significantly smaller than the other terms (Heuristic 1) and that we can omit them; this approximation allows us to express $e_{\mathsf{res},I}$ as a sample from the following Gaussian distribution.

$$\mathcal{N}_{\mathbb{Z}}(m_b\tilde{e_I}, \sum_{i=0}^{n-1} \tilde{e_i}^2\sigma^2 + 2\sigma_{\mathsf{s}}^2\mathsf{ct}.t^2).$$

Obtaining that $|e_{\mathsf{res},I} - B\tilde{e_I}| < |e_{\mathsf{res},I}|$ is more likely when $b = 1$ while, if $|e_{\mathsf{res},I} - B\tilde{e_I}| \geq |e_{\mathsf{res},I}|$, it is at least more likely that $b = 0$ rather than $b = 1$. To compute the advantage of this adversary in distinguishing these distributions, we need to compute the total variation distance between them. Computing this quantity for discrete Gaussian is not easy; therefore, we will approximate it by considering their counterparts on the real numbers. We define $V := \sqrt{\|\tilde{e}\|_2^2\sigma^2 + 2\sigma_{\mathsf{s}}^2\mathsf{ct}.t^2}$ and use Lemma 3 to obtain the following lower bound.

$$\Delta(\mathcal{N}(0,V), \mathcal{N}(B\tilde{e_I}, V)) \geq \frac{1}{50}\frac{B|\tilde{e}_I|}{\sqrt{V}} = \Theta\left(\frac{B|\tilde{e_I}|}{\sqrt{\|\tilde{e}\|_2^2 + 2\sigma_{\mathsf{s}}^2\mathsf{ct}.t^2}}\right)$$

The advantage of the adversary in the $\mathsf{IND\text{-}MKHE}$ game is the square of the total variation distance we just estimated which is $\Theta\left(\frac{B^2|\tilde{e_I}|^2}{\|\tilde{e}\|_2^2 + 2\sigma_{\mathsf{s}}^2\mathsf{ct}.t^2}\right)$.

With high probability $|\tilde{e}_I| \geq 1$ and $\|\tilde{e}\|_{\mathsf{can}} \leq \sigma n$. This implies that $\|\tilde{e}\|_2^2 \leq \sigma^2 n^3$ and also that $\mathsf{ct}.t \leq O(B\sigma n^{3/2})$. Putting together all these bounds, we obtain that the advantage of the adversary is $\Omega\left(\frac{B^2}{\sigma^4 n^3 + 2\sigma_{\mathsf{s}}^2 B^2\sigma^2 n^3}\right) = \Omega\left(\frac{1}{\sigma_{\mathsf{s}}^2\sigma^2 n^3}\right)$.