Combinatorially Homomorphic Encryption

Yuval Ishai, Eyal Kushnir, and Ron D. Rothblum

Technion, Israel
Emails: {yuvali,eyal.kushnir,rothblum}@cs.technion.ac.il

December 31, 2023

Abstract

Homomorphic encryption enables public computation over encrypted data. In the past few decades, homomorphic encryption has become a staple of both the theory and practice of cryptography. Nevertheless, while there is a general loose understanding of what it means for a scheme to be homomorphic, to date there is no single unifying minimal definition that captures all schemes. In this work, we propose a new definition, which we refer to as *combinatorially homomorphic encryption*, which attempts to give a broad base that captures the intuitive meaning of homomorphic encryption and draws a clear line between trivial and nontrivial homomorphism.

Our notion relates the ability to accomplish some task when given a ciphertext, to accomplishing the same task without the ciphertext, in the context of *communication complexity*. Thus, we say that a scheme is combinatorially homomorphic if there exists a communication complexity problem $f(x, y)$ (where x is Alice's input and y is Bob's input) which requires communication c, but can be solved with communication less than c when Alice is given in addition also an encryption $E_k(y)$ of Bob's input (using Bob's key k).

We show that this definition indeed captures pre-existing notions of homomorphic encryption and (suitable variants are) sufficiently strong to derive prior known implications of homomorphic encryption in a conceptually appealing way. These include constructions of (lossy) public-key encryption from homomorphic private-key encryption, as well as collision-resistant hash functions and private information retrieval schemes.

Contents

1 Introduction

Homomorphic encryption, originally proposed by Rivest, Adleman, and Dertouzos [RAD78], is one of the cornerstones of modern cryptography. Roughly speaking, an encryption scheme is homomorphic wrt to a function f if given an encryption of a message m, it is possible to generate an encryption of $f(m)$, without knowing the secret key. Homomorphic encryption is used extensively in cryptography, whether explicitly, or implicitly via homomorphisms offered by concrete schemes (e.g., based on factoring, discrete log, or lattices). Until 2009, the default interpretation of homomorphic encryption was for f to be a linear function; this is still a commonly used special case today both in theory and in practice. However, since then, we have seen the development of *fully* homomorphic encryption schemes [Gen09, BV14], which are homomorphic wrt to *all* functions f.

There are many different candidates for homomorphic encryption from the literature (Goldwasser-Micali [GM84], Benaloh [Ben94], ElGamal [ElG84], Naccache-Stern [NS98], Paillier [Pai99], Damgård-Jurik [DJ01], Regev [Reg05] and more) and many different interpretations and precise definitions for what exact form of homomorphism they achieve. However, all definitions that we are aware of (and are discussed in detail next) are either too strict, in the sense that they only capture a few of the candidates, or are too broad, in the sense that they do not draw a clear line between "trivial" and "nontrivial" homomorphism.

Thus, despite being a central notion in cryptography, there is no canonical definition of what it means for an encryption scheme to be homomorphic. The main goal of this work is to introduce such a broad notion (or rather several variants following one theme) that captures and abstracts the intuition underlying the concept of homomorphic encryption and may serve as a default "minimal" interpretation of what homomorphic encryption means.

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a (private-key or public-key) encryption scheme. We proceed to discuss several takes on the notion of homomorphic encryption, and what we find lacking in each.

Ideal Homomorphism: A very simple and strong definition of homomorphic encryption may require that a homomorphically evaluated ciphertext, generated by an evaluation of the function f on the ciphertext $E_{pk}(m)$, is distributed similarly[1] to $E_{pk}\big(f(m)\big)$.

This notion is extremely strong (and useful) and is satisfied by a few number theoretic based schemes such as Goldwasser-Micali [GM84] and Benaloh [Ben94] (ElGamal [ElG84] and Paillier/Damgård-Jurik [Pai99, DJ01] also offer some form of ideal homomorphism but suffer from caveats that are discussed below). Unfortunately, many other schemes, especially lattice-based ones, do not satisfy it. Moreover, this strong notion is an overkill for many applications.

Algebraic Homomorphism: (a.k.a. Linear Homomorphism or Additive Homomorphism) An algebraic perspective taken earlier on (and inspired by the number-theory based schemes available at the time), is to view the plaintext and ciphertexts spaces as groups, so that the encryption function is a homomorphism from the former to the latter.[2] Thus, running the group operation on the ciphertexts has the effect of implementing the corresponding group operation on the plaintexts.

[1] Several variants of the definition are possible depending on whether the similarity should be perfect, statistical or computational, and also whether it should hold even given additional information such as $E_{pk}(m)$, or even given the corresponding secret-key. We ignore these subtleties here.

[2] Indeed, this is the source of the term homomorphic encryption.

Unfortunately, this definition is quite restrictive. In particular, it does not capture homomorphisms that are non-linear such as [BGN05, IP07, GHV10b] let alone fully-homomorphic schemes (e.g., [Gen09, BV14, GSW13]). ElGamal with plaintexts implemented as group elements is only homomorphic wrt the underlying cryptographic group, whereas ElGamal with plaintexts in the exponent only supports decryption of small plaintext values. Lattice-based encryption schemes such as Regev [Reg05] only support a bounded number of operations that depends on the modulus-to-noise ratio.

Functional Homomorphism: A typical modern definition of (public-key) homomorphic encryption states that an encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is homomorphic wrt to a function f, or (more generally) a class $\mathcal{F}$ of functions, if there exists a poly-time Eval algorithm such that $\mathsf{Dec}_{sk}\Big(\mathsf{Eval}_{pk}\big(\mathsf{Enc}_{pk}(m), f\big)\Big) = f(m)$ for all messages m, key-pairs (pk, sk), and $f \in \mathcal{F}$. To avoid trivial solutions, the homomorphic evaluation algorithm is further assumed to be "compact." This is typically defined to mean that the size of the generated ciphertext or the decryption circuit is smaller than the circuit size of f.[3] The precise notion of compactness varies both quantitatively (Should the size of the evaluated ciphertext be independent of the circuit? Is a poly-logarithmic or even sub-linear dependence allowed?) and qualitatively (Why circuits? Which kind of circuits? How exactly is circuit complexity measured? What about redundancies in the representation?) In particular, it is unclear what a minimal notion of compactness that suffices for applications should be. Moreover, existing notions of compactness that refer to the encrypted output being "smaller than the circuit size" or "sub-linear in the circuit size" are not robust to the circuit model, in the sense that they change their meaning when switching from one standard circuit model to another. Beyond the difficulty with formalizing the common notion of compactness, we point out several additional difficulties with existing definitions of functional homomorphism:

1. Usually, lattice-based schemes only satisfy an approximate notion of this definition as there is a noise associated with each ciphertext, and this noise grows as the homomorphic evaluation is performed, until a point in which the ciphertext is undecryptable.

   This can sometimes be avoided by using a large modulus-to-noise ratio, but that is merely hiding the problem under the rug — we do think of the schemes as homomorphic even when the modulus-to-noise ratio is small, but the definition is not flexible enough to capture this.
2. Discrete-log based schemes such as ElGamal, over a cyclic plaintext group of order q, are often thought of as linearly homomorphic with addition in the group $\mathbb{Z}_q$. As briefly mentioned above though, one can only decrypt ciphertexts whose messages are polynomially small as decryption involves a discrete-log operation.

   Despite this well-known fact, ElGamal is considered to be additively homomorphic but capturing it within the existing framework is quite messy.
3. Lastly, if one wishes to define homomorphic encryption in general, that is, not specifically wrt some function f, this approach becomes problematic. For example, simply assuming

[3] If compactness is not required, then the homomorphic evaluation can be trivially delegated to the decryptor (e.g., by appending the description of the circuit the ciphertext). Nevertheless, some homomorphic schemes such as [SYY99] or constructions based on garbled circuits [CCKM00, HK07, GHV10a, IKO+11] are not compact but are circuit private, meaning that the ciphertext does not reveal the evaluated circuit. In this work, we focus on compact homomorphic encryption, which is meaningful even without circuit privacy.

the existence of *some* function f such that the scheme is functionally homomorphic wrt f is not very meaningful if f is the identity function or a constant function. More generally, it is not entirely clear what non-triviality constraints f needs to satisfy for this notion to be meaningful or useful.

1.1 Combinatorially Homomorphic Encryption

Our main contribution is proposing a new definition for homomorphic encryption. Our goal in this definition is threefold: (1) we wish to find a notion that is consistent and truly formalizes the intuitive meaning of homomorphic encryption, drawing precise lines between "trivial" and "nontrivial" homomorphism; (2) for the definition to be sufficiently broad to capture all schemes that are currently thought of as homomorphic (including both number-theory and lattice-based schemes) and (3) for the definition to be sufficiently strong to preserve the known implications of existing notions of homomorphic encryption such as public-key encryption (PKE), collision-resistant hashing (CRH) and private information retrieval (PIR). We believe that positioning homomorphic encryption as a true cryptographic primitive, similarly to "one-way function" or "public-key encryption", will facilitate a systematic study of its relation with other cryptographic primitives.

We call this new framework *combinatorially homomorphic encryption*, of which we describe several variants. The first variant refers to *communication complexity* [Yao79], which we briefly review. Recall that in *distributional communication complexity* there are two parties, Alice and Bob, who respectively get inputs x and y, drawn from some joint distribution. Their goal is to compute some function $f(x, y)$ while minimizing the number of bits exchanged between them to the extent possible. In our most basic definition (which is sufficient for most of the goals listed above), we focus specifically on *one-way* communication complexity — that is when communication is only allowed from Alice to Bob (and not in the other direction). In other words, the minimal number of bits that Alice needs to send to Bob so that he can compute $f(x, y)$. See [KN97, RY20] for a detailed introduction to communication complexity.

The first instantiation of our framework for homomorphic encryption takes the following operational perspective. We say that a scheme is *communication-complexity (CC) homomorphic* if there exists some one-way communication complexity problem f, which requires communication c, such that if Alice is given, in addition to x, a ciphertext $\mathsf{Enc}_k(y)$ of Bob's input using Bob's key k, then the communication problem can be solved using less than c bits (and where Alice and Bob both run in polynomial-time). Note that while it is possible to talk about CC-homomorphic encryption with respect to a specific communication complexity problem, our main definition refers to the *existence* of a communication complexity problem for which the notion is non-trivial.

Definition 1.1 (Informally Stated, see Section 3)**.** *We say that an encryption scheme* $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* CC homomorphic *if there exists a communication complexity problem f which requires communication c, but there exists a polynomial-time one-way protocol for solving the problem $f'\big((x, \mathsf{Enc}_k(y)), (y, k)\big)$, defined as $f'\big((x, \mathsf{Enc}_k(y)), (y, k)\big) = f(x, y)$, with communication less than c.*

The definition can be adapted to the public-key setting in the natural way (i.e., y is encrypted under the public key and Bob gets the corresponding private key).

CC homomorphic encryption captures the basic intuitive understanding that homomorphic encryption should enable *useful* computation on encrypted data. Here, Alice can perform such a computation in a way that helps Bob derive the output more efficiently than if Alice had not been given the ciphertext.

We also consider generalizations of this notion in two ways. First, we consider an interactive variant (presented in Section 4.4), in which the homomorphic communication game is allowed to be interactive and the communication complexity lower bound holds in the interactive setting (which is the standard model for communication complexity). Motivated by applications described below, we also consider comparing the "homomorphic communication complexity" to other combinatorial measures of the function f such as its VC dimension.[4] Lastly, while our basic definition considers distributional communication complexity over efficiently sampleable *product* distributions, it suffices for our results that the conditional marginal distributions are efficiently sampleable.

Existing Schemes in the Lens of Combinatorially Homomorphic Encryption. To see that CC homomorphic encryption indeed captures existing schemes, consider an encryption scheme that is linearly homomorphic mod 2, in the standard functional sense. To see that such a scheme is combinatorially homomorphic, consider the inner product communication complexity game in which Alice and Bob get as input random vectors $x, y \in \{0,1\}^n$ and Bob's goal is to compute their inner product $\langle x, y\rangle = \bigoplus_{i\in[n]} x_i y_i$. It is well-known that this task requires communication complexity $\Omega(n)$ (in fact, in the one-way version, this follows directly from the leftover hash lemma). However, if Alice is given in addition to x, also a bit-by-bit encryption $\mathsf{Enc}_k(y_1), \ldots, \mathsf{Enc}_k(y_n)$ of Bob's input, then using the linear homomorphism she can compute an encryption of $\langle x, y\rangle$ and send it to Bob, who can decrypt and retrieve the result. The compactness property of functional homomorphic encryption guarantees that the communication in this new protocol is smaller than the $\Omega(n)$ lower bound that holds when Alice is not given the encryption of Bob's input.

The above idea can be generalized to linear homomorphisms over any group, as stated in the following theorem. A simple unifying explanation is that traditional homomorphic schemes from the literature imply PIR, which can be thought of as being CC-homomorphic with respect to the "index" function. In particular, it shows that Goldwasser-Micali [GM84], Benaloh [Ben94] and Regev [Reg05] fall within our framework.

Theorem 1.2 (Informally Stated, see Lemma 3.7)**.** *Any linearly homomorphic private-key encryption scheme is combinatorially homomorphic.*

To illustrate a concrete instantiation, we show a simple private-key scheme based on Learning with Errors (LWE) that satisfies our definition. The secret key is a random vector $\mathbf{s} \leftarrow \mathbb{Z}_q^\lambda$. To encrypt a bit $b \in \{0,1\}$, sample a random $\mathbf{a} \leftarrow \mathbb{Z}_q^\lambda$ and output $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s}\rangle + e + \lfloor q/2\rfloor \cdot b)$ as the ciphertext, where $e \in \mathbb{Z}_q$ comes from a B-bounded noise distribution. The security of this private-key scheme follows almost tautologically from decisional LWE.

Now consider the communication complexity game in which Alice and Bob get as their respective inputs $x, y \in \{0,1\}^n$ and their goal is to compute the inner product. As mentioned above, it is well known that this problem requires communication complexity $\Omega(n)$. Suppose however that Alice is given a bit-by-bit encryption of Bob's input. Namely, ciphertexts $c_1, \ldots, c_n$ such that $c_i = \big(\mathbf{a}_i, \langle \mathbf{a}_i, \mathbf{s}\rangle + e_i + \lfloor q/2\rfloor \cdot y_i\big)$. Alice can now compute a new ciphertext $(\mathbf{a}', \sigma')$, where $\mathbf{a}' = \sum_i x_i \cdot \mathbf{a}_i$ and $\sigma' = \sum_i x_i \cdot \big(\langle \mathbf{a}_i, \mathbf{s}\rangle + e_i + \lfloor q/2\rfloor \cdot y_i\big) = \langle \mathbf{a}', \mathbf{s}\rangle + \sum_i x_i e_i + \lfloor q/2\rfloor \cdot \langle x, y\rangle$ (and all arithmetic is mod q). Alice sends this ciphertext to Bob who computes $\sigma' - \langle a', s\rangle = \sum_i x_i e_i + \lfloor q/2\rfloor \cdot \langle x, y\rangle$. As long as $\sum_i x_i e_i < q/4$ (which holds if $B \cdot n < q/4$), then Bob can now correctly round and obtain $\langle x, y\rangle$.

[4] More precisely, we consider the VC dimension of the function family $\big\{f_x : \{0,1\}^n \to \{0,1\}\big\}_x$, where $f_x(y) = f(x, y)$.

If the communication in this game (which is $(\lambda+1)\cdot\log(q)$) is smaller than the communication complexity lower bound of $\Omega(n)$, then this basic private-key scheme is CC homomorphic.[5]

Jumping ahead, one of our main applications is a construction of *public-key encryption* from any CC homomorphic *private-key* encryption (which extends the [Rot11] construction of public-key encryption from linearly homomorphic encryption). Thus, the above construction yields a public-key encryption scheme from LWE which, we believe, cleanly abstracts Regev's [Reg05] celebrated public-key scheme. Furthermore, our work is the first one to offer a qualitative notion of homomorphism, where each choice of parameters (including secret distribution and noise distribution) can be classified as either being combinatorially homomorphic or not.

Note that the definition of CC homomorphic encryption is sufficiently flexible to allow for variations of linear homomorphisms, and even for non-linear homomorphisms, that may be difficult to capture otherwise. All one needs to do is to adapt the communication complexity game to capture the specific functionality that is offered by the scheme and show the corresponding communication complexity lower bound (which is usually not difficult).

Consider, for example, the ElGamal cryptosystem [ElG84] with plaintexts in the exponent, which is widely considered to be linearly homomorphic, yet is not captured by the standard linearly homomorphic encryption definition (since decryption involves a discrete-log operation). Instead, we can view ElGamal encryption as being "OR-homomorphic" in the following natural way. The scheme uses a cyclic group $\mathbb{G}$ of order q with generator g. The secret key is a random $s \leftarrow \mathbb{Z}_q$. To encrypt a bit $b \in \{0,1\}$, sample a random $r \leftarrow \mathbb{Z}_q$ and output $(g^r, g^{s\cdot r+b})$. To decrypt a ciphertext (c_0, c_1), compute $z = c_1 \cdot c_0^{-s}$ and output 0 if $z = 1$ and 1 otherwise. The security of this private-key scheme follows from the decisional Diffie-Hellman assumption.

To show that the above encryption scheme is CC-homomorphic we will use the well-known Disjointness communication complexity problem, where Alice and Bob are given sets $x, y \subseteq [n]$ respectively, and need to determine whether their sets are disjoint.[6] Suppose that Alice is given, in addition to her input x, also bit-by-bit encryptions $c_1, ..., c_n$ of Bob's input, where $c_i = (g^{r_i}, g^{r_i \cdot s + y_i})$ and the input sets x and y are interpreted as indicator vectors. Alice then computes $\left(\prod_{i:x_i=1} g^{r_i}, \prod_{i:x_i=1} g^{r_i\cdot s+y_i}\right) = (g^{r'}, g^{r'\cdot s+\sum_{i\in\mathcal{I}(x)} y_i})$. Alice then sends the resulting ciphertext to Bob who can compute $z = g^{r'\cdot s+\sum_{i\in\mathcal{I}(x)} y_i} \cdot (g^{r'})^{-s} = g^{\sum_{i\in\mathcal{I}(x)} y_i}$. It holds that $z = 1$ if and only if the sets are disjoint. Therefore, if the communication complexity of this protocol (which is $2\log(q)$) is smaller than the communication complexity lower bound (which is $\Omega(n)$), then the private-key scheme is CC-homomorphic.

The above idea can be generalized to capture any encryption scheme that is homomorphic with respect to the OR operation, as stated in the following theorem.

Theorem 1.3 (Informally Stated, see Lemma 3.8)**.** *Any OR-homomorphic private-key encryption scheme is combinatorially homomorphic.*

We also show a specific instantiation of our scheme using low-noise LPN (i.e., when the absolute

[5]The homomorphic private-key to public-key transformation of Rothblum [Rot11] can also be viewed as morally giving an abstraction of Regev's scheme, but the actual formal definition of homomorphic encryption used in [Rot11] is not technically achieved by the above private-key scheme.

[6]As a matter of fact, it suffices for our purpose to consider the *one-way* communication complexity of disjointness. An $\Omega(n)$ lower bound for this problem follows from a simple lower bound on the one-way randomized communication complexity of the index problem, in which Alice is given an input $x \in \{0,1\}^n$ and Bob gets input $i \in [n]$ and needs to output x_i. A simple information-theoretic argument shows that when x and i are chosen uniformly at random, then x_i has high entropy given Alice's message (and i) and therefore, by Fano's inequality, is unpredictable.

noise is roughly $\log^2(\lambda)$). Using our framework in combination with the applications listed below, we can re-derive recent results on LPN (due to [BLVW19, BF22]) in a way that we find to be conceptually simpler. Note that unlike the previous examples, in which the communication game has perfect correctness, here there is only a small but non-negligible advantage over guessing a random output.

Applications. As our main technical results, we show that suitable variations of our basic notion of combinatorially homomorphic encryption suffice to derive some of the key applications that are known from (say) standard linearly homomorphic encryption.

Our first main result shows how to transform any combinatorially homomorphic *private-key* encryption into a public-key one. This generalizes the work of Rothblum [Rot11], who gave such a transformation for linearly homomorphic private-key encryption. As a matter of fact, we obtain the stronger notion of *lossy* public-key encryption [PVW08, BHY09] (which is equivalent to semi-honest two-message statistical oblivious transfer [HLOV11]).

Theorem 1.4 (Informally Stated, see Theorem 4.1)**.** *If there exists a combinatorially homomorphic private-key encryption scheme then there exists a lossy public-key encryption scheme.*

We remark that the security property required from the private-key scheme is very mild (and in particular is weaker than CPA security). Specifically, we merely need a weak notion of "distributional security" (see Definition 2.8) which, loosely speaking, requires that the distributions $\big(y, \mathsf{Enc}_k(y)\big)$ and $\big(y, \mathsf{Enc}_k(y')\big)$ are computationally indistinguishable, where y, y' are independent samples drawn from Bob's input distribution in the communication game.

As it is instructive to understanding the power of CC homomorphic encryption, we briefly sketch a simplified proof of Theorem 1.4 next. The public key of the scheme is $\big(y, \mathsf{Enc}_k(y)\big)$, where y is a random input for Bob in the communication game, and k is the private key of the private-key scheme. To encrypt a bit b, a random input x for Alice is sampled, and the ciphertext is Alice's message in the "homomorphic" protocol m_A, as well as $f(x, y) \oplus b$. To decrypt, we run Bob on input $\big((y, k), m_A\big)$ to compute $f(x, y)$, and then we can retrieve the message bit b. Correctness follows from the correctness of the homomorphic protocol. As for security, using the distributional security of the private-key scheme, we can switch the public key $\big(y, \mathsf{Enc}_k(y)\big)$ to the lossy public key $\big(y, \mathsf{Enc}_k(y')\big)$. Thus, the adversary's goal now is essentially to compute $f(x, y)$ given $\big(y, \mathsf{Enc}_k(y')\big)$ and m_A.

Assume that this is possible. Then we can derive a more efficient communication complexity protocol for computing f in the standard setting, in which Alice gets only x and Bob gets only y. Alice and Bob sample a key k and a ciphertext $\mathsf{Enc}_k(y')$ using shared randomness.[7] Then, Alice generates a message m_A from the homomorphic protocol and sends it to Bob, who can then run the adversary on input $((y, c), m_A)$ to compute $f(x, y)$. Since we required that Alice's message in the homomorphic protocol is shorter than the communication complexity of f, we derive a contradiction. Note that this argument immediately gives the stronger notion of *lossy* encryption.

This basic result can be generalized to interactive combinatorially homomorphic encryption in which case we derive a key agreement protocol (which can be thought of as an interactive analog of public-key encryption).

[7] As usual in distributional communication complexity, this shared randomness can be eliminated by non-uniformly fixing the best choice.

Theorem 1.5 (Informally Stated, see Theorem 4.11)**.** *If there exists an interactive combinatorially homomorphic encryption scheme then there exists a key agreement protocol.*

Ishai, Kushilevitz and Ostrovsky [IKO05] showed how to construct a *collision-resistant hash function* (CRH) from any linearly homomorphic encryption scheme. Recall that a CRH is a collection of shrinking hash functions so that no polynomial-time adversary can find a collision, given the description of a random function from the collection. We generalize the [IKO05] result and construct CRH from any CC homomorphic encryption.

Theorem 1.6 (Informally Stated, see Theorem 4.7)**.** *If there exists a combinatorially homomorphic encryption scheme (satisfying a mild non-triviality constraint) then there exists a collision-resistant hash function.*

(The mild non-triviality constraint that we require is that the communication complexity problem is defined wrt a function f such that the function family $\left\{f_y : \{0,1\}^n \to \{0,1\}\right\}_y$, where $f_y(x) = f(x,y)$, is a universal hash function family).

As in [IKO05], for this application, we do not need the decryption algorithm to be efficient, and a more general notion of "CC homomorphic commitment" (in which Bob can be inefficient in the communication game) suffices.

Next, we revisit the Kushilevitz-Ostrovsky [KO97] construction of *private information retrieval* (PIR) scheme from a linearly homomorphic encryption scheme.[8] Recall that a PIR scheme is a two-party protocol between a server, which is given a database $x \in \{0,1\}^n$, and a client who is given as input an index $i \in [n]$. The goal is for the client to reconstruct x_i whereas the index i is computationally hidden from the server (both parties are assumed to be polynomial-time). We say that the PIR scheme is non-trivial if the communication complexity is less than n.[9]

We generalize the [KO97] construction and derive PIR from combinatorially homomorphic encryption. For this result, we need the communication in the homomorphic variant of the communication game to be shorter than before. Specifically, rather than beating the communication complexity lower bound for the game, it should beat its *VC dimension*. We refer to schemes satisfying this (intuitively stronger) notion as *VC homomorphic*.

Theorem 1.7 (Informally Stated, see Theorem 4.9)**.** *Assume that there exists a VC homomorphic encryption scheme then there exists a non-trivial PIR scheme.*

Applications from Learning Parity with Noise. As noted above, we can capture a low noise variant of LPN (specifically with an absolute noise level of roughly $\log^2(n)$) in our framework, via a simple construction. Using Theorem 1.6, we can use LPN with this noise level to obtain CRH, thereby giving a conceptually simple derivation of recent results [BLVW19, YZW+19]. Similarly, using Theorem 1.4 we get a simple construction of semi-honest 2-message statistical OT from LPN. This can be viewed as an abstraction of a recent result of Bitansky and Freizeit [BF22]. We emphasize though that [BF22] use the semi-honest construction only as a stepping stone towards a construction that achieves security against malicious receivers (but additionally requires a Nisan-Wigderson style derandomization assumption).

[8] The [KO97] construction is based on the Quadratic Residuosity assumption, but is easy to generalize to compact linearly homomorphic encryption (for a suitable notion of compactness), see [Ste98, Lip05].

[9] While a PIR scheme with communication, say, $n-1$ does not seem directly useful, it is sufficient for deriving some important consequences of PIR such as CRH [IKO05], oblivious transfer [CMO00], lossy encryption [HLOV11] and SZK hardness [LV16].

1.2 Related Work

As previously mentioned, Rothblum [Rot11] showed that any linearly homomorphic encryption that satisfies a mild compactness property can be used to construct a public-key encryption scheme. His proof relies on the Leftover Hash Lemma and can be streamlined using our framework (see discussion in Section 1.1).

Alamati *et al.* [AMP19, AMPR19] study the possibility of constructing Cryptomania primitives (such as CRH and PKE) based on Minicrypt primitives that are equipped with certain algebraic structures. Their work is limited to primitives with group homomorphism over the input or output spaces. In particular, like [Rot11], their work does not consider non-linear homomorphisms.

Bogdanov and Lee [BL13] study the limits of security for homomorphic encryption. Along the way, they introduce a notion of sensitivity for homomorphically evaluated functions. While this notion suffices for their applications, it does not seem to be a minimal notion of non-triviality for functional homomorphisms.

Cohen and Naor [CN22] study a different connection between communication complexity and cryptography, and in particular, show that the existence of non-trivial communication complexity protocols in which the inputs are drawn from efficiently sampleable distributions imply cryptographic primitives (such as distribution collision-resistant hash functions).

2 Preliminaries

For a distribution D, we denote by $x \leftarrow D$ the process of sampling from D. For any joint distribution (X, Y) we will denote by $x \leftarrow \mathsf{Proj}_1(X, Y)$ or $y \leftarrow \mathsf{Proj}_2(X, Y)$ sampling from (X, Y) and keeping only the first or the second element of the pair, respectively. A function $\mu : \mathbb{N} \to [0, 1]$ is *negligible* if for every polynomial p and sufficiently large λ it holds that $\mu(\lambda) \leq 1/p(\lambda)$. All logarithms considered in this paper are in base 2.

Definition 2.1 (Statistical Distance)**.** *Let X and Y be two distributions over a finite domain U. The statistical distance between X and Y is defined as follows.*

$$\mathsf{SD}(X, Y) = \max_{f:U\to\{0,1\}} \left| \Pr\left[f(X) = 1\right] - \Pr\left[f(Y) = 1\right] \right|.$$

If $\mathsf{SD}(X, Y) \leq \epsilon$ we say that X is ϵ-close to Y.

Next, we define computational indistinguishability, which can be thought of as a computational analog of the statistical distance.

Definition 2.2 (Computational Indistinguishability)**.** *We say that two distribution ensembles $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ and $Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$ are* computationally indistinguishable*, and denote it by $X \approx_c Y$, if for every probabilistic polynomial-size distinguisher $\mathcal{D}$ there exists a negligible function μ such that for every $\lambda \in \mathbb{N}$,*

$$\left|\Pr\left[\mathcal{D}(X_\lambda) = 1\right] - \Pr\left[\mathcal{D}(Y_\lambda) = 1\right]\right| \leq \mu(\lambda).$$

2.1 Communication Complexity

Communication complexity (CC), introduced by Yao [Yao79], provides a mathematical model for the study of communication between two or more parties. It has proven to be a powerful tool

in a surprising variety of fields such as circuit complexity, streaming, and quantum computing. We refer to the books by Kushilevitz and Nisan [KN97] and by Rao and Yehudayoff [RY20] for a comprehensive introduction. We now turn to recall several CC-related definitions that will be used in this paper.

Let f be a 2-argument function. Consider the setting of two communicating parties, Alice and Bob, who are given inputs x and y respectively, and wish to *cooperatively* compute the value of $f(x,y)$ (without loss of generality we will require that only Bob outputs this value). The communication between them is conducted according to some fixed deterministic protocol π. The output of the protocol (i.e., Bob's output) on inputs x and y is denoted by $\pi(x,y)$.

Distributional Communication Complexity We allow the protocol to err with a small probability on some input distribution. Namely,

Definition 2.3 (Protocol Correctness)**.** *Given a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ and a joint input distribution (X,Y), we say that a deterministic protocol π computes f with error ϵ on (X,Y) if*

$$\Pr\Big[\pi(x,y) \neq f(x,y) : (x,y) \leftarrow (X,Y)\Big] \leq \epsilon.$$

Interchangeably, we can say that the protocol π computes f with correctness $1-\epsilon$ *on (X,Y).*

The communication complexity of a protocol π on inputs x and y is defined to be the number of bits exchanged by the parties while running the protocol on these inputs. The length of a protocol π on input distribution (X,Y), denoted by $\mathsf{CC}[\pi,(X,Y)]$, is defined to be the maximal communication complexity of the protocol on any input in the support of the distribution (notice that this measure is well-defined since these sets are finite).

The ϵ-error distributional communication complexity of f on (X,Y) is the minimal length of any deterministic protocol computing f with error ϵ with respect to (X,Y). That is,

Definition 2.4 (Distributional Communication Complexity)**.** *Given a function f and a joint input distribution (X,Y) we define the* ϵ-error (X,Y)-distributional communication complexity *of f as follows.*

$$\mathcal{D}^{A\leftrightarrow B}\big(f,(X,Y),\epsilon\big) := \min_{\substack{\pi \text{ computes } f \\ \text{with error } \epsilon \\ \text{on } (X,Y)}} \mathsf{CC}[\pi,(X,Y)].$$

The *one-way ϵ-error (X,Y)-distributional communication complexity* of f, which we denote by $\mathcal{D}^{A\to B}\big(f,(X,Y),\epsilon\big)$, is defined similarly but limited to one-round protocols that consist of only one message - from Alice to Bob.

Discrepancy The discrepancy method is a common technique for proving lower bounds on distributional communication complexity. We now define the discrepancy of a function with respect to an input distribution.

Definition 2.5 (Discrepancy)**.** *Given a function $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ and a joint input distribution (X,Y) we define the discrepancy of f on a rectangle $R = S \times T \subseteq (X,Y)$, denoted here by*

$\mathsf{Disc}\big(f,(X,Y);R\big)$, *as follows.*

$$\mathsf{Disc}\big(f,(X,Y);R\big) := \left|\Pr\Big[(x,y)\in R \wedge f(x,y)=1\Big] - \Pr\Big[(x,y)\in R \wedge f(x,y)=0\Big]\right|,$$

where $(x,y)\leftarrow(X,Y)$. *The discrepancy of* f *on* (X,Y) *is defined as*

$$\mathsf{Disc}\big(f,(X,Y)\big) := \max_{R} \mathsf{Disc}\big(f,(X,Y);R\big).$$

A well-known theorem (see, e.g., [RY20, Theorem 5.2]) shows that the discrepancy can be used to lower bound distributional communication complexity.

Theorem 2.6. *For any function* $f:\mathcal{X}\times\mathcal{Y}\to\{0,1\}$, *a joint input distribution* (X,Y) *and an error rate* $\epsilon\in(0,\frac{1}{2})$ *we have that*

$$\mathcal{D}^{A\to B}\big(f,(X,Y),\epsilon\big) \geq \log\left(\frac{1-2\epsilon}{\mathsf{Disc}\big(f,(X,Y)\big)}\right)$$

2.2 VC Dimension

Definition 2.7 (VC Dimension)**.** *Let* H *be a set of functions* $h:\mathcal{Y}\to\{0,1\}$. *We say that a set* $I\subseteq\mathcal{Y}$ *is* shattered *by* H, *if for every possible assignment* $A:I\to\{0,1\}$ *there exists a function* $h\in H$ *that is consistent with* A. *Namely,*

$$\forall y\in I, h(y)=A(y).$$

The largest value d *for which there exists a set* I *of size* d *that is shattered by* H *is the* Vapnik-Chervonenkis (VC) dimension *of* H, *denoted by* $\mathsf{VC}(H)$.

2.3 Encryption

In this subsection, we describe different notions of encryption that will be used throughout this work. We start by defining a notion of private-key encryption which is (one-time) secure with respect to a specific message distribution.

Definition 2.8 ($\mathcal{M}$-Distributional Secure Private-Key Encryption)**.** *Let* $\mathcal{M}=(\mathcal{M}_\lambda)_{\lambda\in\mathbb{N}}$ *be a message distribution. An* $\mathcal{M}$-distributional secure private-key encryption *scheme* $\mathcal{E}=(\mathsf{Gen},\mathsf{Enc},\mathsf{Dec})$, *with correctness error* $\epsilon=\epsilon(\lambda)$, *is a triplet of probabilistic polynomial-time algorithms with the following syntax.*

- ***Key generation.*** *Given a security parameter* 1^λ, *the algorithm* Gen *outputs a key* k.
- ***Encryption.*** *Given a message* $m\in\mathcal{M}_\lambda$ *and a key* k, *the algorithm* Enc *outputs a ciphertext* c.
- ***Decryption.*** *Given a ciphertext* c *and a key* k, *the algorithm* Dec *outputs a message* m.

We require $\mathcal{E}$ *to satisfy the following properties.*

- ***Correctness.*** *For any $\lambda \in \mathbb{N}$ and message $m \in \mathcal{M}_\lambda$ it holds that $\Pr\left[\mathsf{Dec}_k(c) = m\right] \geq 1-\epsilon(\lambda)$, where $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $c \leftarrow \mathsf{Enc}_k(m)$.*
- ***$\mathcal{M}$-distributional security.*** *$\left(m, \mathsf{Enc}_k(m)\right)_{\lambda\in\mathbb{N}} \approx_c \left(m, \mathsf{Enc}_k(m')\right)_{\lambda\in\mathbb{N}}$, where m and m' are two independent messages sampled from $\mathcal{M}$.*

We remark that the notion of distributional security defined above is weaker than standard security notions such as CPA security since (1) the adversary is not given access to an encryption oracle and (2) security needs to hold only wrt messages arising from the given distribution (rather than "worst-case" messages).

Definition 2.9 (CPA-Secure Private-Key Encryption)**.** *A* chosen-plaintext attack (CPA) secure private-key encryption *scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message length $\ell = \ell(\lambda)$ and correctness error $\epsilon = \epsilon(\lambda)$, is defined similarly to Definition 2.8 but the distributional security requirement is replaced with the following:*

- ***CPA Security.*** *Consider the following security game.*

  1. *The challenger samples a key $k \leftarrow \mathsf{Gen}(1^\lambda)$.*
  2. *The adversary chooses a message m of length $\ell(\lambda)$ and receives $\mathsf{Enc}_k(m)$ from the challenger. This step is repeated for a polynomial number of times.*
  3. *The adversary chooses two challenge message m_0, m_1 of length $\ell(\lambda)$ and receives from the challenger $\mathsf{Enc}_k(m_b)$.*
  4. *The adversary outputs a bit $b' \in \{0,1\}$.*

  For any probabilistic polynomial-size adversary $\mathcal{A}$, we denote by $\mathsf{CPA}^b_{\mathcal{A}}(1^\lambda)$ the output of $\mathcal{A}$ in the game above, and we require that there exists a negligible function μ such that for any $\lambda \in \mathbb{N}$,

$$\left|\Pr\left[\mathsf{CPA}^0_{\mathcal{A}}(1^\lambda) = 1\right] - \Pr\left[\mathsf{CPA}^1_{\mathcal{A}}(1^\lambda) = 1\right]\right| \leq \mu(\lambda).$$

We will next define a variant of lossy encryption [PVW08, BHY09], which is equivalent to a 2-message (semi-honest) statistical OT [PVW08].

Definition 2.10 (Lossy Encryption)**.** *Let $\nu = \nu(\lambda)$ and $\epsilon = \epsilon(\lambda)$. A* ν-lossy bit-encryption *scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LossyGen})$ with correctness error ϵ, is a quadruple of polynomial-time algorithms with the following syntax,*

- ***Key generation.*** *Given a security parameter 1^λ, the algorithm Gen outputs a secret key sk and a public key pk.*
- ***Encryption.*** *Given a bit b and a public key pk, the algorithm Enc outputs a ciphertext c.*
- ***Decryption.*** *Given a ciphertext c and a secret key sk, the algorithm Dec outputs a bit b.*
- ***Lossy key generation.*** *Given a security parameter 1^λ, the algorithm $\mathsf{LossyGen}$ outputs a lossy key lk.*

We require $\mathcal{E}$ to satisfy the following properties.

- ***Correctness.*** *For any* $\lambda \in \mathbb{N}$ *and bit* b *it holds that* $\Pr\left[\mathsf{Dec}_{sk}(c) = b\right] \geq 1 - \epsilon(\lambda)$*, where* $(sk, pk) \leftarrow \mathsf{Gen}(1^\lambda)$ *and* $c \leftarrow \mathsf{Enc}_{pk}(b)$*.*

- ***Key indistinguishability.*** $\left(\mathsf{Proj}_2\left(\mathsf{Gen}(1^\lambda)\right)\right)_{\lambda\in\mathbb{N}} \approx_c \left(\mathsf{LossyGen}(1^\lambda)\right)_{\lambda\in\mathbb{N}}$*.*

- ***Lossiness of lossy keys.*** *For any* $\lambda \in \mathbb{N}$*, we have that* $\left(lk, \mathsf{Enc}_{lk}(0)\right)$ *is* $\nu(\lambda)$*-close in statistical distance to* $\left(lk, \mathsf{Enc}_{lk}(1)\right)$*, where* $lk \leftarrow \mathsf{LossyGen}(1^\lambda)$*.*

If not otherwise specified, by default, we take the parameters ν and ϵ to be negligible in parameter λ. One can also consider relaxed notions of lossy encryption, where either the correctness error is high — namely, $\epsilon(\lambda) = \frac{1}{2} - \frac{1}{p(\lambda)}$, for some polynomial p — or the statistical distance between encryptions under a lossy key is large — namely, $\nu(\lambda) = 1 - \frac{1}{p(\lambda)}$, for some polynomial p. Next, we will show that both variants are equivalent to the standard definition. We note however that if both the correctness *and* lossiness are close to $1/2$, then amplification is not known (see [DNR04, HR05] for further discussion and relation to the circuit polarization problem).

Lemma 2.11 (Weak-Correctness Lossy Encryption implies Lossy Encryption)**.** *Assume there exists a lossy encryption scheme with correctness error* $\frac{1}{2} - \frac{1}{p(\lambda)}$*, for some polynomial* p*, then there exists a lossy encryption scheme (Definition 2.10).*

Lemma 2.12 (Weak-Lossiness Lossy Encryption implies Lossy Encryption)**.** *Assume there exists a* $(1 - \frac{1}{p(\lambda)})$*-lossy encryption scheme, for some polynomial* p*, then there exists a lossy encryption scheme (Definition 2.10).*

The proofs of Lemma 2.11 and Lemma 2.12 are given in Appendix B.

2.4 Collision Resistant Hash Function

Definition 2.13 (Collision Resistant Hash Function)**.** *A* collision resistant function *with input length* $\ell(n)$ *and output length* $\ell'(n) < \ell(n)$ *is defined by a pair of algorithms* $(\mathsf{Gen}, \mathsf{Eval})$ *with the following syntax,*

- ***Key generation.*** *Given* 1^λ *the probabilistic polynomial-time algorithm* Gen *outputs an index* s*.*

- ***Evaluation.*** *Given index* s *and input* x *of length* $\ell(\lambda)$*, the polynomial-time algorithm* Eval *outputs* $y \in \{0,1\}^{\ell'(\lambda)}$*.*

For any $\lambda \in \mathbb{N}$*,* $s \leftarrow \mathsf{Gen}(1^\lambda)$ *and* $x \in \{0,1\}^{\ell(\lambda)}$ *we define* $h_s(x) := \mathsf{Eval}(s, x)$*.*

We require the scheme to satisfy the following collision resistance property.

- ***Collision resistance.*** *for every probabilistic polynomial-size adversary* $\mathcal{A}$ *there exists a negligible function* μ *such that for any* $\lambda \in \mathbb{N}$*,*

$$\Pr\left[x \neq x' \wedge h_s(x) = h_s(x') : \begin{array}{c} s \leftarrow \mathsf{Gen}(1^\lambda), \\ (x, x') \leftarrow \mathcal{A}(s) \end{array}\right] \leq \mu(\lambda).$$

2.5 Private Information Retrieval

We define private information retrieval from a database. For our purposes, it will be convenient to view the database size as a function of the security parameter.

Definition 2.14 (Private Information Retrieval [CKGS98, KO97])**.** *A* private information retrieval *(PIR) scheme* $(\mathsf{Query}, \mathsf{Resp}, \mathsf{Recon})$ *with database length* $\ell = \ell(\lambda)$*, response length* $\ell'(\lambda) < \ell(\lambda)$ *and correctness error* $\epsilon = \epsilon(\lambda)$*, is a triplet of polynomial-time algorithms with the following syntax.*

- ***Query.*** *Given a security parameter* 1^λ *and an index* $i \in [\ell(\lambda)]$*, the probabilistic algorithm* Query *outputs a query* q *and a state* st*.*
- ***Response.*** *Given a database* $D \in \{0,1\}^{\ell(\lambda)}$ *and a query* q*, the deterministic algorithm* Resp *outputs a response* r *of length* $\ell'(\lambda)$*.*
- ***Reconstruct.*** *Given a state* st *and a response* r*, the deterministic algorithm* Recon *outputs a bit* b*.*

We require the scheme to satisfy the following properties.

- ***Correctness.*** *For any* $\lambda \in \mathbb{N}$*,* $D \in \{0,1\}^{\ell(\lambda)}$ *and* $i \in [\ell(\lambda)]$*,*

$$\Pr\Big[\mathsf{Recon}\big(st, \mathsf{Resp}(D, q)\big) = D_i\Big] \geq 1 - \epsilon(\lambda),$$

  where $(q, st) \leftarrow \mathsf{Query}(1^\lambda, i)$*.*

- ***Client privacy.*** *For any probabilistic polynomial-size adversary* $\mathcal{A}$*, there exists a negligible function* μ *such that for any* $\lambda \in \mathbb{N}$*,* $D \in \{0,1\}^{\ell(\lambda)}$ *and* $i, j \in [\ell(\lambda)]$*,*

$$\left|\Pr\Big[\mathcal{A}(1^\lambda, D, q_i) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, D, q_j) = 1\Big]\right| \leq \mu(\lambda),$$

  where $(q_i, st_i) \leftarrow \mathsf{Query}(1^\lambda, i)$ *and* $(q_j, st_j) \leftarrow \mathsf{Query}(1^\lambda, j)$*.*

- ***Response succinctness***[10]***.*** *For any* $\lambda \in \mathbb{N}$*,* $D \in \{0,1\}^{\ell(\lambda)}$ *and* $i \in [\ell(\lambda)]$*,*

$$\ell'(\lambda) := \big|\mathsf{Resp}(D, q)\big| < \ell(\lambda),$$

  where $(q, st) \leftarrow \mathsf{Query}(1^\lambda, i)$*.*

We say that a PIR scheme is nontrivial *if* $\ell'(\lambda) \leq \ell(\lambda) - 1$*.*

3 Combinatorially Homomorphic Encryption

First, we define an extension of a function ensemble and an input distribution ensemble with respect to a private key encryption scheme. These will be used throughout the following sections.

[10] PIR with this minimal notion of succinctness implies other primitives, such as oblivious transfer [CMO00], CRH [IKO05], and lossy encryption [HLOV11], but is not known to be implied by them [HHS08].

Let f be an ensemble of 2-argument functions. Let (X, Y) be an ensemble of input distributions, where $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ and $Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a private-key encryption scheme (see Definition 2.8). We extend f and (X, Y) by defining for every $\lambda \in \mathbb{N}$,

$$\mathsf{Ext}_{\mathcal{E}}(X_\lambda, Y_\lambda) := \left\{ \big((x,c),(y,k)\big) : \begin{array}{c} (x,y) \leftarrow (X_\lambda, Y_\lambda) \\ k \leftarrow \mathsf{Gen}(1^\lambda) \\ c \leftarrow \mathsf{Enc}_k(y) \end{array} \right\},$$
$$\mathsf{Ext}_{\mathcal{E}}(f_\lambda) : \big((x,c),(y,k)\big) \mapsto f_\lambda(x,y).$$

We denote $\mathsf{Ext}_{\mathcal{E}}(X,Y) := \big(\mathsf{Ext}_{\mathcal{E}}(X_\lambda, Y_\lambda)\big)_{\lambda\in\mathbb{N}}$ and $\mathsf{Ext}_{\mathcal{E}}(f) := \big(\mathsf{Ext}_{\mathcal{E}}(f_\lambda)\big)_{\lambda\in\mathbb{N}}$.

3.1 CC-Homomorphic Encryption

We now introduce our new homomorphic encryption definition. Informally, an encryption scheme $\mathcal{E}$ is combinatorially homomorphic if there exists a polynomial-time communication protocol for $\mathsf{Ext}_{\mathcal{E}}(f)$ that utilizes the homomorphic properties of $\mathcal{E}$ to achieve communication cost that is lower than the standard communication complexity of f, on a specific input distribution.

For this section, we require the function f to be *balanced* wrt to the joint distribution (X, Y). That is,

$$\Pr\left[f(x,y) = 0 : (x,y) \leftarrow (X,Y)\right] = \Pr\left[f(x,y) = 1 : (x,y) \leftarrow (X,Y)\right] = \frac{1}{2}.$$

In Appendix A, we present a generalization of our definition that allows us to remove this restriction. We postpone this generalization to the appendix since it requires a distinguishability-based adaptation of the communication complexity definition (in contrast to the standard predictability-based definition).

We put forward two variants of the definition. Namely, CC-homomorphism in the *perfect correctness regime*, where we require the "homomorphic protocol" for $\mathsf{Ext}_{\mathcal{E}}(f)$ to have (near) perfect correctness, and CC-homomorphism in the *statistical hardness regime*, where we allow imperfect correctness, but require any computationally unbounded protocol for f to have negligible advantage over a random guess.

Our definitions will require the input distribution to be *efficiently sampleable*, defined as follows.

Definition 3.1 (Efficiently Sampleable Distribution)**.** *We say that a distribution ensemble (X, Y) is* efficiently sampleable *if there exists a probabilistic polynomial-time sampling algorithm that given 1^λ outputs a random element from (X_λ, Y_λ).*

Definition 3.2 (Communication Complexity Homomorphic Encryption in the Perfect Correctness Regime)**.** *A private-key encryption scheme $\mathcal{E}$ (Definition 2.8) is* communication-complexity homomorphic *(or* CC-homomorphic*) in the* perfect correctness regime*, if there exists a function ensemble f, an efficiently sampleable* product *distribution ensemble (X, Y) and a function $c = c(\lambda)$ such that,*

- *There exists a polynomial-time one-way protocol that computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on input distribution $\mathsf{Ext}_{\mathcal{E}}(X, Y)$, using $c(\lambda)$ bits of communication,*

- *Any unbounded one-way protocol that computes f on (X,Y), using $c(\lambda)$ bits of communication has correctness at most $1-\frac{1}{p(\lambda)}$, for some polynomial p.*

Remark 3.3. *A natural relaxation of the definition allows a negligible failure probability in the homomorphic communication protocol. However, jumping ahead, having perfect correctness here will be useful as it will also lead to perfect correctness in some of our applications (e.g., lossy encryption, see Theorem 4.1).*

Remark 3.4. *Instead of requiring that (X,Y) is an ensemble of* product *distributions, it is sufficient to require it to be an ensemble of joint distributions such that the conditional distributions $X|Y$ are efficiently sampleable.*

Definition 3.5 (Communication Complexity Homomorphic Encryption in the Statistical Hardness Regime)**.** *A private-key encryption scheme $\mathcal{E}$ (Definition 2.8) is* communication-complexity homomorphic *(or* CC-homomorphic*) in the* statistical hardness regime*, if there exists a function ensemble f, an efficiently sampleable* product *distribution ensemble (X,Y) and a function $c=c(\lambda)$ such that,*

- *There exists a polynomial-time one-way protocol that computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with correctness at least $\frac{1}{2}+\frac{1}{p(\lambda)}$, for some polynomial p, on $\mathsf{Ext}_{\mathcal{E}}(X,Y)$ using c bits of communication,*
- *There exists a negligible function μ such that any unbounded one-way protocol that computes f on input distribution (X,Y) using c bits of communication has correctness at most $\frac{1}{2}+\mu(\lambda)$, for any sufficiently large λ.*

As a first step, we will show that these definitions generalize a typical homomorphic encryption definition that captures most traditional homomorphic encryption schemes. Namely, *linearly homomorphic encryption*, defined as follows.

Definition 3.6 (Linearly Homomorphic Encryption)**.** *Let $\mathcal{M}=(\mathcal{M}_\lambda)_{\lambda\in\mathbb{N}}$ be a plaintext space equipped with an additive group structure, and let $\mathcal{C}=(\mathcal{C}_\lambda)_{\lambda\in\mathbb{N}}$ be the ensemble of all single output polynomial-size circuits over the group. A private-key encryption scheme $(\mathsf{Gen},\mathsf{Enc},\mathsf{Dec})$ with plaintext space $\mathcal{M}$ and correctness error $\epsilon=\epsilon(\lambda)$ is* linearly homomorphic *if there exists an additional probabilistic polynomial-time algorithm Eval with following syntax.*

- ***Homomorphic evaluation.*** *Given a circuit $C\in\mathcal{C}$ with n inputs gates and given ciphertexts $c_1,...,c_n$, the algorithm Eval outputs a ciphertext c_{res}.*

We require the scheme to satisfy the following requirements.

- ***Correctness.*** *For any $\lambda\in\mathbb{N}$, circuit $C\in\mathcal{C}_\lambda$ with n input gates, and any messages $m_1,...,m_n\in\mathcal{M}_\lambda$,*

$$\Pr\Big[\mathsf{Dec}_k\big(\mathsf{Eval}(C,c_1,...,c_n)\big)=C(m_1,...,m_n)\Big]\geq 1-\epsilon(\lambda),$$

  where $\forall i\in[n], c_i\leftarrow\mathsf{Enc}_k(m_i)$ and $k\leftarrow\mathsf{Gen}(1^\lambda)$.

- ***Compactness.*** *There exists a polynomial p such that for any $\lambda\in\mathbb{N}$, circuit $C\in\mathcal{C}_\lambda$ and messages $m_1,...,m_n\in\mathcal{M}_\lambda$,*

$$\big|\mathsf{Eval}(C,c_1,...,c_n)\big|\leq p(\lambda).$$

Lemma 3.7 (Linearly HE implies CC-homomorphic encryption)**.** *Any private-key linearly homomorphic encryption scheme (Definition 3.6) with perfect correctness is CC-homomorphic in the perfect correctness regime (Definition 3.2).*

A simple explanation of Lemma 3.7, as formalized in the following proof, is that traditional homomorphic schemes from the literature imply PIR, which can be thought of as being CC-homomorphic with respect to the *index function*. We will prove Lemma 3.7 in the perfect correctness regime (Definition 3.2), but the proof can be adapted also to the statistical hardness regime (Definition 3.5).

Proof. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ be a private-key linearly homomorphic encryption scheme with plaintext space $\mathcal{M}$ and perfect correctness. Let p be a polynomial guaranteed to exist by the compactness property of $\mathcal{E}$, and denote $q = q(\lambda) = p^2(\lambda)$.

Let $e_1, ..., e_q$ be the unit vectors in $\mathbb{F}_2^q$ and let $f : \mathbb{F}_2^q \times \{e_1, ..., e_q\}$ be the *index* function where $f(x, e_i) = x_i$. I.e., given a vector x and a unit vector e_i the function outputs the i'th entry of x. We fix the input distribution (X, Y) to be the uniform distribution over the domain.

Consider the following polynomial-time one-way protocol π. Alice is given input $x \in X$ and bit-by-bit encryption $c_1, ..., c_q$ of Bob's input $y \in Y$. She computes $c_{res} = \mathsf{Eval}(C_x, c_1, ..., c_q)$, where $C_x(z_1, ..., z_q) = \sum_{i \in \mathcal{I}_x} z_i$ and $\mathcal{I}_x := \{i \ : \ x_i = 1\}$. Alice then sends c_{res} to Bob, who outputs $\mathsf{Dec}_k(c_{res})$. By the correctness of $\mathcal{E}$ we have that for any $\lambda \in \mathbb{N}$,

$$\Pr\Big[\pi\big((x,c),(y,k)\big) = f(x,y)\Big] = \Pr\Big[\mathsf{Dec}_k\big(\mathsf{Eval}(C_x, c_1, ..., c_q)\big) = C_x(y_1, ..., y_q)\Big] = 1,$$

where $(x, y) \leftarrow (X, Y)$, $k \leftarrow \mathsf{Gen}(1^\lambda)$, $\forall i \in [q], c_i \leftarrow \mathsf{Enc}_k(y_i)$ and $c = (c_1, ..., c_q)$. Meaning, the protocol π computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ using $|c_{res}| \leq p$ bits of communication.

On the other hand, fix a sufficiently large polynomial τ. By Kremer *et al.* [KNR99, Theorem 5] it holds that for any $\lambda \in \mathbb{N}$,

$$\mathcal{D}^{A \to B}\big(f, (X, Y), 1 - \frac{1}{\tau(\lambda)}\big) = \Omega(q) = \omega(p).$$

Meaning, that any unbounded one-way protocol that computes f using $O(p)$ bits of communication has correctness at most $1 - \tau(\lambda)$ on (X, Y).

Therefore, the encryption scheme $\mathcal{E}$ is CC-homomorphic by Definition 3.2. □

Lemma 3.7 shows that CC-homomorphism generalizes the standard definition of homomorphic encryption. Next, we will show that it is strictly more general by proving that any encryption scheme which is homomorphic with respect to the OR operation, is CC-homomorphic. A concrete example of such an encryption scheme is the ElGamal cryptosystem [ElG84], which is widely considered homomorphic, yet is not captured by the current standard HE definitions, e.g., Definition 3.6.

Lemma 3.8 (OR-homomorphic Encryption implies CC-homomorphic Encryption)**.** *Any private-key encryption scheme with perfect correctness that is homomorphic with respect to the OR function*[11] *is CC-homomorphic in the perfect correctness regime (Definition 3.2).*

[11] OR-homomorphic encryption is defined similarly to Definition 3.6, but with a binary plaintext space and evaluation algorithm that supports polynomial-size circuits with OR gates.

Lemma 3.8 can be proved using the index functionality, similarly to the proof of Lemma 3.7. However, we put forward an alternative proof using the well-known Disjointness communication complexity problem.

Proof. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ be a private-key OR-homomorphic encryption scheme with perfect correctness. Let p be a polynomial guaranteed to exist by the compactness property of $\mathcal{E}$, and denote $q = q(\lambda) = p^4(\lambda)$. Denote by $\mathsf{Or} : \{0,1\}^* \to \{0,1\}$ the OR function that outputs 1 iff there exists an input bit that equals 1.

Let $X = Y$ be the uniform distribution over all subsets of size $\sqrt{q}$ of $[q]$ (encoded as binary strings over $\{0,1\}^q$). Let f be the *disjointness* function over (X, Y), where $f(x, y) = 0$ iff $x \cap y = \emptyset$.

Consider the following polynomial-time one-way protocol π. Alice is given input $x \in X$ and bit-by-bit encryption $c_1, ..., c_q$ of Bob's input $y \in Y$. She computes $c_{res} = \mathsf{Eval}\big(\mathsf{Or}, (c_i)_{i \in \mathcal{I}_x}\big)$, where $\mathcal{I}_x := \{i \ : \ x_i = 1\}$. Alice then sends c_{res} to Bob, who outputs $\mathsf{Dec}_k(c_{res})$. By the correctness of $\mathcal{E}$ we have that for any $\lambda \in \mathbb{N}$,

$$\Pr\Big[\pi\big((x,c),(y,k)\big) = f(x,y)\Big] = \Pr\Big[\mathsf{Dec}_k\Big(\mathsf{Eval}\big(\mathsf{Or}, (c_i)_{i\in\mathcal{I}_x}\Big) = \mathsf{Or}\big((c_i)_{i\in\mathcal{I}_x}\big)\Big] = 1,$$

where $(x, y) \leftarrow (X, Y)$, $k \leftarrow \mathsf{Gen}(1^\lambda)$, $\forall i \in [q], c_i \leftarrow \mathsf{Enc}_k(y_i)$ and $c = (c_1, ..., c_q)$. Meaning, the protocol π computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ using $|c_{res}| \leq p$ bits of communication.

On the other hand, fix a sufficiently large polynomial τ. By Babai *et al.* [BFS86, Theorem 7.2] it holds that for any $\lambda \in \mathbb{N}$,

$$\mathcal{D}^{A\to B}\big(f, (X,Y), 1 - \frac{1}{\tau(\lambda)}\big) = \Omega(\sqrt{q}) = \omega(p).$$

Meaning, that any unbounded one-way protocol that computes f using $O(p)$ bits of communication has correctness at most $1 - \tau(\lambda)$ on (X, Y).

Therefore, the encryption scheme $\mathcal{E}$ is CC-homomorphic by Definition 3.2. □

3.2 VC-Homomorphic Encryption

We now define another variant of combinatorially homomorphic encryption based on an *efficiently computable* variant of the VC dimension measure (Definition 2.7) introduced by Boyle *et al.* [BIP18].

For any function ensemble f over input distribution ensemble (X, Y) we define the following function ensemble,

$$f_X = \Big(\big\{f(x,\cdot) \ : \ x \in X_\lambda\big\}\Big)_{\lambda\in\mathbb{N}}.$$

Definition 3.9 (Efficient Shattering Scheme)**.** *Fix $\psi : \mathbb{N} \to \mathbb{N}$ and let f be a function ensemble over input distribution ensemble (X, Y). An* efficient shattering *scheme for f with parameter ψ is defined by a pair of polynomial-time algorithms* $(\mathsf{Shatter}, \mathsf{Assign})$ *as follows.*

- ***Find a shattered set.*** *Given 1^λ the deterministic algorithm* Shatter *outputs a set $I \subset Y$ such that $|I| > \psi(\lambda)$ and is shattered by f_X.*
- ***Find a function for a given assignment.*** *Given $\lambda \in \mathbb{N}$, a shattered set $I \subset Y$ and an assignment $A : I \to \{0,1\}$, the algorithm* Assign *outputs a circuit computing a function $f \in f_X$ which is consistent with A.*

Remark 3.10. *Notice that by definition, the parameter ψ is strictly bounded from above by the VC dimension (Definition 2.7) of the function ensemble f_X.*

Remark 3.11. *Notice that if a function ensemble has an efficient shattering scheme with parameter ψ, it also has an efficient shattering scheme for any parameter $\psi' < \psi$.*

Definition 3.12 (VC Homomorphic Encryption)**.** *A private-key encryption scheme $\mathcal{E}$ (Definition 2.8) is* VC-homomorphic *if there exists a function ensemble f and a product distribution ensemble (X, Y) such that,*

- *There exists a polynomial-time one-way protocol that computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ using c bits of communication,*
- *There exists an efficient shattering scheme (Definition 3.9) for f with parameter c,*

for some function $c = c(\lambda)$.

There is an interesting connection between CC-homomorphism and VC-homomorphism arising from the following known claim.

Claim 3.13. *[KNR99, Theorem 3.2] For every 2-argument function f and error-rate $\epsilon \in (0, 1)$,*

$$\max_{\text{product } (X,Y)} \mathcal{D}^{A\to B}\big(f, (X, Y), \epsilon\big) = O\big(\mathsf{VC}(f_X)\big).$$

To use this theorem to link between CC-homomorphism and VC-homomorphism, we put forward a stronger variant of Definition 3.2.

Definition 3.14 (Strongly CC-Homomorphic Encryption)**.** *A CC-homomorphic encryption scheme (Definition 3.2) is* strongly CC-homomorphic *if it satisfies the following requirement.*

- *Any unbounded one-way protocol that computes f using $O(c)$ bits of communication has correctness at most $1 - \frac{1}{p(\lambda)}$ on (X, Y), for some polynomial p,*

where c is the parameter from Definition 3.2.

Remark 3.15. *The proof of Lemma 3.7 actually shows that any linearly homomorphic encryption is* strongly *CC-homomorphic.*

Theorem 3.16. *Any strongly CC-homomorphic encryption scheme (Definition 3.14) with respect to a function f that has an efficient shattering scheme (Definition 3.9) with parameter $\psi = \mathsf{VC}(f_X) - 1$, is VC-homomorphic (Definition 3.12).*

Proof. Let $\mathcal{E}$ be a strongly CC-homomorphic encryption scheme with respect to function ensemble f and input product distribution ensemble (X, Y), such that f has an efficient shattering scheme with parameter $\psi = \mathsf{VC}(f_X) - 1$.

By Definition 3.2, there exists a *polynomial-time* one-way protocol π that computes the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ using c bits of communication. Furthermore, any unbounded one-way protocol that computes f using $O(c)$ bits of communication has correctness at most $1 - \frac{1}{p(\lambda)}$ on (X, Y), for some polynomial p. Denote,

$$\rho := \mathcal{D}^{A\to B}\big(f, (X, Y), 1 - \frac{1}{p(\lambda)}\big)$$

Using the standard communication complexity notation (see Definition 2.4), for any $z \in \mathbb{R}$ and any sufficiently large $\lambda \in \mathbb{N}$ we have that $z \cdot c \leq \rho$. Furthermore, by Claim 3.13 we have that $\rho = O\big(\mathsf{VC}(f_X)\big)$. Meaning, there exists a constant $z \in \mathbb{R}$ such that for any sufficiently large $\lambda \in \mathbb{N}$, it holds that $\rho \leq z \cdot \mathsf{VC}(f_X)$. Therefore, for any sufficiently large $\lambda \in \mathbb{N}$ we have that $c < \frac{\rho}{z} \leq \mathsf{VC}(f_X) - 1 < \mathsf{VC}(f_X)$.

The scheme $\mathcal{E}$ satisfies the two conditions of Definition 3.12 and therefore it is VC-homomorphic. □

4 Applications

In this section, we demonstrate applications of our new notions of homomorphic encryption. In Section 4.1 we construct Lossy Encryption. In Section 4.2 we construct a Collision Resistant Hash function. In Section 4.3 we construct a Private Information Retrieval protocol. In Section 4.4 we construct a Key Agreement protocol.

4.1 Lossy Encryption

In this section, we show how to use CC-homomorphic encryption to construct lossy public-key encryption.

Theorem 4.1 (CC-homomorphic Encryption implies Lossy Encryption)**.** *Assume there exists a CC-homomorphic encryption scheme in either the perfect correctness regime (see Definition 3.2) or the statistical hardness regime (see Definition 3.5), then there exists a lossy encryption scheme.*

We will prove Theorem 4.1 in the statistical hardness regime (Definition 3.5). The proof in the perfect correctness regime (Definition 3.2) is similar, but produces a $(1 - \frac{1}{p(\lambda)})$-lossy encryption, for some polynomial p, with perfect correctness that can be amplified to full-fledged lossy encryption scheme using Lemma 2.12.

Proof of Theorem 4.1. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a Y-distributional secure CC-homomorphic encryption scheme with respect to function ensemble f and input product distribution ensemble (X, Y) such that $\Pr\left[f(x, y) = 0 : (x, y) \leftarrow (X, Y)\right] = \frac{1}{2}$. Let π be a polynomial-time one-way protocol computing the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$ with correctness $\frac{1}{2} + \frac{1}{p(\lambda)}$ on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$, for some polynomial p, with communication cost $c = c(\lambda)$, such that there exists a negligible function μ such that any unbounded protocol that computes f on (X, Y) using c bits of communication has correctness at most $\frac{1}{2} + \mu(\lambda)$.

For the following, given input $\big((x, c), (y, k)\big)$ from $\mathsf{Ext}_{\mathcal{E}}(X, Y)$, we denote by $\mathsf{Alice}(x, c)$ the message Alice generates in the protocol and we denote by $\mathsf{Bob}(y, k, m_A)$ the output of Bob after receiving a message m_A from Alice. Consider the following scheme $(\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*, \mathsf{LossyGen}^*)$.

- **Key generation.** Given a security parameter 1^λ the probabilistic polynomial-time algorithm Gen^* samples a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and an element $y \leftarrow Y$, and outputs the public key $pk = \big(y, \mathsf{Enc}_k(y)\big)$ and the secret key $sk = (y, k)$.
- **Encryption.** Given the public key $pk = (y, c)$ and a bit b, the probabilistic polynomial-time algorithm Enc^* samples $x \leftarrow X$ that satisfies $f(x, y) = b$ (by rejection sampling) and outputs $m_A = \mathsf{Alice}(x, c)$.

- **Decryption.** Given the secret key $sk = (y, k)$ and a ciphertext m_A, the deterministic polynomial-time algorithm Dec^* outputs $\mathsf{Bob}(y, k, m_A)$.
- **Lossy Key generation.** Given a security parameter 1^λ the probabilistic polynomial-time algorithm $\mathsf{LossyGen}^*$ samples a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and elements $y, y' \leftarrow Y$, and outputs the lossy key $lk = \big(y, \mathsf{Enc}_k(y')\big)$.

Claim 4.2. *The scheme satisfies correctness (see Definition 2.10).*

Proof. For any $\lambda \in \mathbb{N}$,

$$\begin{aligned}\Pr\left[\mathsf{Dec}^*_{sk}\big(\mathsf{Enc}^*_{pk}(b)\big) \neq b\right] &\underset{(1)}{=} \Pr\left[\mathsf{Bob}\big(y, k, \mathsf{Alice}(x, c)\big) \neq f(x, y) : \begin{array}{c}(x, y) \leftarrow (X, Y)\\ \text{s.t. } f(x, y) = b\end{array}\right]\\ &\underset{(2)}{=} \Pr\left[\mathsf{Bob}\big(y, k, \mathsf{Alice}(x, c)\big) \neq f(x, y) : (x, y) \leftarrow (X, Y)\right]\\ &\underset{(3)}{\leq} \frac{1}{2} - \frac{1}{p(\lambda)},\end{aligned}$$

where $b \leftarrow \{0, 1\}$, $(sk, pk) \leftarrow \mathsf{Gen}^*(1^\lambda)$, $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $c \leftarrow \mathsf{Enc}_k(y)$, and where (1) is by the definition of the scheme, (2) is since $\Pr\left[f(x, y) = 0 : (x, y) \leftarrow (X, Y)\right] = \frac{1}{2}$, and therefore sampling $b \leftarrow \{0, 1\}$ and then sampling from (X, Y) conditioned on $f(x, y) = b$ is the same as sampling directly from (X, Y), and (3) is since the protocol π computes $\mathsf{Ext}_{\mathcal{E}}(f)$ on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ with correctness $\frac{1}{2} + \frac{1}{p(\lambda)}$, and since $\big((x, c), (y, k)\big)$ is sampled similarly to a random sample from $\mathsf{Ext}_{\mathcal{E}}(X, Y)$. □

Claim 4.3. *The scheme satisfies key indistinguishability (see Definition 2.10).*

Proof. We have that for any fixed y and y' sampled from Y,

$$\Big(\mathsf{Proj}_2\big(\mathsf{Gen}^*(1^\lambda)\big)\Big)_{\lambda\in\mathbb{N}} = \big(y, c\big)_{\lambda\in\mathbb{N}} \approx_c \big(y, c'\big)_{\lambda\in\mathbb{N}} = \big(\mathsf{LossyGen}^*(1^\lambda)\big)_{\lambda\in\mathbb{N}},$$

where $k \leftarrow \mathsf{Gen}(1^\lambda)$, $c \leftarrow \mathsf{Enc}_k(y)$ and $c' \leftarrow \mathsf{Enc}_k(y')$, and where the equalities are by the definition of the scheme and the computational indistinguishability is by the Y-distributional security of $\mathcal{E}$. □

Claim 4.4. *The scheme satisfies lossiness of lossy keys (see Definition 2.10).*

Proof. We will show that given an unbounded distinguisher for encryptions under a lossy key, with non-negligible distinguishing advantage, one can construct a one-way protocol in the standard distributional communication complexity model (Section 2.1) that computes f with correctness $\frac{1}{2} + \frac{1}{\tau(\lambda)}$ on (X, Y), for some polynomial τ, with communication cost c. Such a protocol cannot exist by our assumption that $\mathcal{E}$ is CC-homomorphic in the statistical hardness regime with respect to f and (X, Y) (see Definition 3.5).

Assume towards a contradiction that there exists a (computationally unbounded) distinguisher $\mathcal{D}$ and a polynomial τ such that for infinitely many $\lambda \in \mathbb{N}$,

$$\Pr\left[\mathcal{D}\big(lk, \mathsf{Enc}^*_{lk}(b)\big) = b : b \leftarrow \{0, 1\},\ lk \leftarrow \mathsf{LossyGen}^*(1^\lambda)\right] \geq \frac{1}{2} + \frac{1}{\tau(\lambda)}.$$

By the definitions of $\mathsf{LossyGen}^*$ and Enc^* we have that for infinitely many $\lambda \in \mathbb{N}$,

$$\Pr\Big[\mathcal{D}\big(y, c, \mathsf{Alice}(x, c)\big) = f(x, y)\Big] \geq \frac{1}{2} + \frac{1}{\tau(\lambda)},$$

where $x \leftarrow X$, $y, y' \leftarrow Y$, $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $c \leftarrow \mathsf{Enc}_k(y')$.

We start by constructing a protocol in the standard distributional communication complexity model (Section 2.1) that uses shared randomness which we will eliminate later. Consider the following unbounded one-way protocol π^* between parties Alice^* and Bob^* who are given inputs x and y sampled from (X, Y) and have access to shared random coins.

1. Alice^* and Bob^* sample a key $k \leftarrow \mathsf{Gen}(1^\lambda)$, an element $y' \leftarrow Y$ and an encryption $c \leftarrow \mathsf{Enc}_k(y')$ using the shared random coins.
2. Alice^* sends $m_A = \mathsf{Alice}(x, c)$ to Bob^*.
3. Bob^* runs $\mathcal{D}$ on (y, c, m_A) and outputs its answer.

We denote by $\pi^*(x, y; r)$ the output of the protocol on inputs (x, y) and random coins r. infinitely many $\lambda \in \mathbb{N}$,

$$\Pr\left[\pi^*(x, y; r) = f(x, y) : \begin{array}{c}(x, y) \leftarrow (X, Y) \\ r \leftarrow \{0,1\}^*\end{array}\right] = \Pr\Big[\mathcal{D}\big(y, c, \mathsf{Alice}(x, c)\big) = f(x, y)\Big] \geq \frac{1}{2} + \frac{1}{\tau(\lambda)},$$

where $x \leftarrow X$, $y, y' \leftarrow Y$, $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $c \leftarrow \mathsf{Enc}_k(y')$.

The above statement holds over a random choice of r. However, by an averaging argument, for infinitely many $\lambda \in \mathbb{N}$ there exists a fixed randomness r^* such that

$$\Pr\Big[\pi^*(x, y; r^*) = f(x, y) : (x, y) \leftarrow (X, Y)\Big] \geq \frac{1}{2} + \frac{1}{\tau(\lambda)}.$$

To conclude, we have that π^* with fixed random coins r^* is an unbounded one-way protocol that computes f with correctness $\frac{1}{2} + \frac{1}{\tau(\lambda)}$ on (X, Y) with communication cost $|\mathsf{Alice}(x, c)| = c$, which is a contradiction to the assumption that such a protocol cannot exist. □

□

4.2 Collision Resistant Hash Function

Next, we use a variant of CC-homomorphic encryption to construct a collision resistant hash function. First, we define an *efficient encoding* algorithm for a set X.

Definition 4.5 (Efficient Encoding)**.** *Let $X = (X_\lambda)_{\lambda \in \mathbb{N}}$ be an ensemble of finite sets. We say that X supports an* efficient encoding *with input length $\ell = \ell(\lambda)$ if there exists an efficiently computable (polynomial-time) injective function* $\mathsf{Encode} : \{0,1\}^\ell \to X_\lambda$.

Our CRH construction will require a function f and input distribution (X, Y) such that the ensemble $f_Y = (f_\lambda)_{\lambda \in \mathbb{N}}$, where $f_\lambda := \big\{f(\cdot, y) \ : \ y \in Y_\lambda\big\}$, is a universal hash function family. We put forward the definition.

Definition 4.6 (Universal Hash Function Family)**.** *A set H of functions from X to $\{0,1\}$ is a universal hash function family if for every distinct $x_1, x_2 \in X$ the hash function family H satisfies the following constraint.*

$$\Pr\left[h(x_1) = h(x_2) : h \leftarrow H\right] \leq \frac{1}{2}.$$

Theorem 4.7 (CC-homomorphic encryption implies CRH)**.** *Assume there exists a CC-homomorphic encryption scheme (Definitions 3.2 and 3.5) with respect to function f, input distribution (X,Y) and parameter c that satisfies the following conditions.*

- *The function ensemble $\left(\left\{f(\cdot,y) \,:\, y \in Y_\lambda\right\}\right)_{\lambda\in\mathbb{N}}$ is a universal hash function family.*
- *The polynomial-time protocol for $\mathsf{Ext}_{\mathcal{E}}(f)$ is correct on any input from $\mathsf{Ext}_{\mathcal{E}}(X,Y)$ w.p. $\frac{1}{2} + \frac{1}{p(\lambda)}$, for some polynomial p,*
- *The ensemble X supports an efficient encoding with input length $\ell(\lambda) \geq c(\lambda)$ for any sufficiently large λ.*

Then, there exists a collision resistant hash function (Definition 2.13).

Remark 4.8. *As a matter of fact, similarly to [IKO05], a relaxed notion of encryption with an inefficient decryption algorithm (in other words, a commitment scheme) is sufficient.*

We will prove Theorem 4.7 in the statistical hardness regime (Definition 3.5), but it can also be adapted to the perfect correctness regime (Definition 3.2).

Proof of Theorem 4.7. Let f be a function ensemble and (X,Y) be an input distribution ensemble such that $\left(\left\{f(\cdot,y) \,:\, y \in Y_\lambda\right\}\right)_{\lambda\in\mathbb{N}}$ is a universal hash function family and such that X supports an efficient encoding with input length $\ell = \ell(\lambda)$. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a Y-distributional secure encryption scheme. Let π be a polynomial-time one-way protocol computing the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$ with correctness $\frac{1}{2} + \frac{1}{p(\lambda)}$ on any input from $\mathsf{Ext}_{\mathcal{E}}(X,Y)$, for some polynomial p, with communication cost $\ell'(\lambda) < \ell(\lambda)$.

Consider the following scheme $(\mathsf{Gen}^*, \mathsf{Eval}^*)$.

- **Key generation.** Given security parameter 1^λ, the probabilistic polynomial-time algorithm Gen^* samples $y \leftarrow Y$, $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $s \leftarrow \mathsf{Enc}_k(y)$ and outputs s.
- **Evaluation.** Given index s and input $m \in \{0,1\}^{\ell(\lambda)}$, the polynomial-time algorithm Eval^* outputs $\mathsf{Alice}\big(\mathsf{Encode}(m), s\big)$.

We first show that the scheme indeed compresses. Indeed, for any $\lambda \in \mathbb{N}$, $s \leftarrow \mathsf{Gen}^*(1^\lambda)$ and $m \in \{0,1\}^{\ell(\lambda)}$,

$$\left|h_s(m)\right| = \left|\mathsf{Alice}\big(\mathsf{Encode}(m), s\big)\right| \leq \ell'(\lambda) < \ell(\lambda).$$

Assume towards a contradiction that the scheme is not collision resistant. Therefore, there exists a probabilistic polynomial-size adversary $\mathcal{A}$ and a polynomial q such that for infinitely many $\lambda \in \mathbb{N}$,

$$\Pr\left[m \neq m' \wedge h_s(m) = h_s(m') : \begin{array}{c} s \leftarrow \mathsf{Gen}^*(1^\lambda), \\ (m,m') \leftarrow \mathcal{A}(s) \end{array}\right] = \frac{1}{q(\lambda)}.$$

Consider the distinguisher $\mathcal{D}$ for the Y-distributional security of $\mathcal{E}$. Given $\big(y_0, c\big)$, where $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y_0, y_1 \leftarrow Y$, $b \leftarrow \{0,1\}$ and $c \leftarrow \mathsf{Enc}_k(y_b)$, the distinguisher $\mathcal{D}$ computes $(m, m') \leftarrow \mathcal{A}(c_b)$. It then checks that $m \neq m'$, that $h_c(m) = h_c(m')$ and that $f\big(\mathsf{Encode}(m), y_0\big) = f\big(\mathsf{Encode}(m'), y_0\big)$. If all checks pass, it outputs 1. Otherwise, it outputs a random bit. For the following, we denote $x := \mathsf{Encode}(m)$, $x' := \mathsf{Encode}(m')$.

We first consider the case where $b = 0$. Given $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y_0 \leftarrow Y$, $c \leftarrow \mathsf{Enc}_k(y_0)$ and $(m, m') \leftarrow \mathcal{A}(c)$, we define the following events,

1. The event E_1 where $f(x, y_0) = f(x', y_0)$.
2. The event E_2 where $m \neq m'$ and $h_c(m) = h_c(m')$.
3. The event E_3 where $\pi\big((x, c), (y_0, k)\big) = \pi\big((x', c), (y_0, k)\big)$.
4. The event E_4 where the protocol π is correct on both $\big((x, c), (y_0, k)\big)$ and $\big((x', c), (y_0, k)\big)$, or is wrong on both of them.

First, since π is correct on any input w.p. at least $\frac{1}{2} + \frac{1}{p(\lambda)}$, there exists a function $\tau : \mathbb{N} \to \mathbb{N}$ such that π is correct on any input w.p. exactly $\frac{1}{2} + \frac{1}{\tau(\lambda)}$, and $\tau(\lambda) \leq p(\lambda)$ for any $\lambda \in \mathbb{N}$. Therefore,

$$\Pr[E_4] = \left(\frac{1}{2} + \frac{1}{\tau(\lambda)}\right)^2 + \left(\frac{1}{2} - \frac{1}{\tau(\lambda)}\right)^2 = \frac{1}{2} + \frac{2}{\tau^2(\lambda)} \geq \frac{1}{2} + \frac{2}{p^2(\lambda)}. \tag{1}$$

Furthermore, we have that,

$$\begin{aligned}
\Pr[E_1|E_2] &\underset{(1)}{=} \Pr[E_1|E_2 \wedge E_3] \\
&\geq \Pr[E_1 \wedge E_4|E_2 \wedge E_3] \\
&\underset{(2)}{=} \Pr[E_1|E_2 \wedge E_3 \wedge E_4] \cdot \Pr[E_4] \\
&\underset{(3)}{=} \Pr[E_4],
\end{aligned} \tag{2}$$

where (1) is since assuming E_2 happened, we have that $\mathsf{Alice}(x, c) = h_c(m) = h_c(m') = \mathsf{Alice}(x', c)$, and therefore, since π is a deterministic one-way protocol, we have that $\pi\big((x, c), (y_0, k)\big) = \pi\big((x', c), (y_0, k)\big)$, (2) is by conditional probability, and (3) is since if the protocol outputs the same output on both inputs and is correct on both of them or wrong on both of them, then $f(x, y_0) = f(x', y_0)$.

Finally, for infinitely many $\lambda \in \mathbb{N}$ we have that,

$$\begin{aligned}
\Pr\big[\mathcal{D}(y_0, c) = 1\big] &\underset{(1)}{=} \Pr[E_1 \wedge E_2] + \frac{1}{2} \cdot \big(1 - \Pr[E_1 \wedge E_2]\big) \\
&= \frac{1}{2} + \frac{1}{2} \cdot \Pr[E_1 \wedge E_2] \\
&= \frac{1}{2} + \frac{1}{2} \Pr[E_1|E_2] \cdot \Pr[E_2] \\
&\underset{(2)}{=} \frac{1}{2} + \frac{1}{2q(\lambda)} \Pr[E_1|E_2] \\
&\underset{(3)}{\geq} \frac{1}{2} + \frac{1}{2q(\lambda)} \cdot \left(\frac{1}{2} + \frac{2}{p^2(\lambda)}\right),
\end{aligned}$$

where $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y_0 \leftarrow Y$, $c \leftarrow \mathsf{Enc}_k(y_0)$ and $(m, m') \leftarrow \mathcal{A}(c)$, and where (1) is by the definition of $\mathcal{D}$, (2) is since $\mathcal{D}$ simulates for the adversary $\mathcal{A}$ a proper collision resistant game, and event E_2 is the event where $\mathcal{A}$ wins in this game, which happens w.p. $1/q(\lambda)$, and (3) is by Eqs. (1) and (2).

On the other hand, for the case where $b = 1$, we have that for any $\lambda \in \mathbb{N}$,

$$\begin{aligned}\Pr\left[\mathcal{D}(y_0, c) = 1\right] &\underset{(1)}{=} \frac{1}{2} + \frac{1}{2q(\lambda)} \Pr\left[f(x, y_0) = f(x', y_0) | m \neq m' \wedge h_s(m) = h_s(m')\right] \\ &\leq \frac{1}{2} + \frac{1}{2q(\lambda)} \Pr\left[f(x, y_0) = f(x', y_0)\right] \\ &\underset{(2)}{=} \frac{1}{2} + \frac{1}{2q(\lambda)} \cdot \frac{1}{2},\end{aligned}$$

where $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y_0, y_1 \leftarrow Y$, $c \leftarrow \mathsf{Enc}_k(y_1)$ and $(m, m') \leftarrow \mathcal{A}(c)$, and where (1) follows by similar reasoning as in the case where $b = 0$ and (2) is since x and x' are independent of y_0 and since f_Y is a universal hash family, and therefore the probability that $f(x, y_0) = f(x', y_0)$ is $1/2$.

Therefore, for infinitely many $\lambda \in \mathbb{N}$,

$$\begin{aligned}\left|\Pr\left[\mathcal{D}(y_0, c_0) = 1\right] - \Pr\left[\mathcal{D}(y_0, c_1) = 1\right]\right| &\geq \left(\frac{1}{2} + \frac{1}{2q(\lambda)} \cdot \left(\frac{1}{2} + \frac{2}{p^2(\lambda)}\right)\right) - \left(\frac{1}{2} + \frac{1}{2q(\lambda)} \cdot \frac{1}{2}\right) \\ &= \frac{2}{2q(\lambda) \cdot p^2(\lambda)},\end{aligned}$$

where $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y_0, y_1 \leftarrow Y$ and $c_b \leftarrow \mathsf{Enc}_k(y_b)$ for $b \in \{0, 1\}$, in contradiction to the assumption that $\mathcal{E}$ is Y-distributional secure. □

4.3 Private Information Retrieval

Theorem 4.9 (VC-homomorphic encryption implies nontrivial PIR)**.** *Assume there exists a CPA-secure (Definition 2.9) VC-homomorphic encryption scheme (Definition 3.12), then there exists a nontrivial private information retrieval scheme (Definition 2.14).*

Proof. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a CPA-secure VC-homomorphic encryption scheme. Let $\pi = (\mathsf{Alice}, \mathsf{Bob})$ be a polynomial-time protocol computing $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ using c bits of communication. Let $(\mathsf{Shatter}, \mathsf{Assign})$ be an efficient shattering scheme (Definition 3.9) for f with parameter c guaranteed to exist.

Set $n = n(\lambda) = c(\lambda) + 1$. Consider the following scheme $(\mathsf{Query}, \mathsf{Resp}, \mathsf{Recon})$.

- **Query.** Given a security parameter 1^λ and an index $i \in [n]$ the probabilistic algorithm Query uses $\mathsf{Shatter}$ to find a set $\{y_1, ..., y_n\} \subset Y$ shattered by f_X, generates a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and outputs a query $q = \big(\{y_1, ..., y_n\}, \mathsf{Enc}_k(y_i)\big)$ and a state $st = (y_i, k)$.
- **Response.** Given a database $D \in \{0, 1\}^n$ and a query $q = \big(\{y_1, ..., y_n\}, c\big)$, the deterministic algorithm Resp uses Assign to find $x \in X$ such that $\forall j \in [n]$, $f(x, y_j) = D_j$, and outputs a response $r = \mathsf{Alice}(x, c)$.
- **Reconstruct.** Given a state st and a response r, the deterministic algorithm Recon outputs a bit $b = \mathsf{Bob}(st, r)$.

Correctness follows by the correctness of π. Namely for any $\lambda \in \mathbb{N}$, $D \in \{0,1\}^n$ and $i \in [n]$,

$$\Pr\left[\mathsf{Recon}\big(st, \mathsf{Resp}(D,q)\big) = D_i\right] = \Pr\left[\pi\big((x,c),(y_i,k)\big) = f(x,y_i) : \begin{array}{c} k \leftarrow \mathsf{Gen}(1^\lambda) \\ c \leftarrow \mathsf{Enc}_k(y_i) \end{array}\right] = 1.$$

where $(q, st) \leftarrow \mathsf{Query}(1^\lambda, i)$.

Client privacy follows by the CPA security of the encryption scheme. Let $\mathcal{A}$ be a probabilistic polynomial-size adversary and let $\lambda \in \mathbb{N}$, $D \in \{0,1\}^n$ and $i, j \in [n]$. Consider an adversary $\mathcal{A}'$ for the CPA security game of $\mathcal{E}$ that given a 1^λ uses $\mathsf{Shatter}$ to get $y_1, ..., y_n$, chooses messages $m_0 = y_i$ and $m_1 = y_j$, gets from the challenger encryption c of m_b and outputs $\mathcal{A}(1^\lambda, 0^n, c)$. We have that

$$\left|\Pr\left[\mathcal{A}(1^\lambda, D, q_i) = 1\right] - \Pr\left[\mathcal{A}(1^\lambda, D, q_j) = 1\right]\right| = \left|\Pr\left[\mathcal{A}'(1^\lambda, c_i) = 1\right] - \Pr\left[\mathcal{A}'(1^\lambda, c_j) = 1\right]\right|,$$

where $(q_k, st_k) \leftarrow \mathsf{Query}(1^\lambda, k)$ and $c_k \leftarrow \mathsf{Enc}_k(y_k)$ for $k \in \{i, j\}$.

Response succinctness follows by the VC-homomorphic definition. Namely, for any $\lambda \in \mathbb{N}$, $D \in \{0,1\}^n$ and $i \in [n]$ we have that $\big|\mathsf{Resp}(D,q)\big| = c < n$, for $(q, st) \leftarrow \mathsf{Query}(1^\lambda, i)$. □

4.4 Key Agreement

To instantiate a key agreement protocol we define an interactive variant of Definition 3.2. We note that using a direct interactive adaptation of the definition would allow Bob to send the secret key from which Alice can deduce the result. This will give the model excessive power that will render it incomparable to the standard communication complexity model. To overcome this obstacle we introduce another participant to the model, a referee, who holds the secret key but is not allowed to participate in the communication phase. Rather the referee sees the entire transcript, and the key, and based on these needs to decide.

As before, let $\mathcal{E}$ be a Y-distributional secure private-key encryption scheme (Definition 2.8). Let f be a function ensemble. Let (X, Y) be an input distribution ensemble. Let Alice and Bob be two polynomial-time communicating parties working in accordance with a protocol π and let the referee be a polynomial-time algorithm. The model works as follows. For any $\lambda \in \mathbb{N}$,

1. Inputs $(x, y) \leftarrow (X, Y)$, a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and a ciphertext $c \leftarrow \mathsf{Enc}_k(y)$ are sampled.
2. Alice and Bob get (x, c) and y respectively.
3. Alice and Bob exchange messages in accordance with π.
4. Given the transcript of the protocol and the key k the referee outputs σ, which is defined to be the output of the protocol and is denoted by $\pi\big((x,c), y, k\big)$.

Similarly to as in Definition 2.3, we say that the protocol computes f with error $\epsilon = \epsilon(\lambda)$ if for every sufficiently large $\lambda \in \mathbb{N}$,

$$\Pr\left[\pi\big((x,c), y, k\big) \neq f_\lambda(x,y) : \big((x,c),(y,k)\big) \leftarrow \mathsf{Ext}_{\mathcal{E}}(X,Y)\right] \leq \epsilon.$$

We define CC homomorphic encryption with respect to interactive protocols. The following definition is given in the balanced regime (**??**), but it can also be adapted to the perfect correctness regime (Definition 3.2).

Definition 4.10 (Interactive CC-Homomorphic Encryption)**.** *A private-key encryption scheme $\mathcal{E}$ (Definition 2.8) is* interactive communication-complexity homomorphic *in the balanced regime if there exists a function ensemble f and an efficiently sampleable* product *distribution ensemble (X, Y) such that,*

- $\Pr[f(x, y) = 0 : (x, y) \leftarrow (X, Y)] = \frac{1}{2}$,
- *There exists a polynomial-time protocol that computes $\mathsf{Ext}_{\mathcal{E}}(f)$ with correctness $\frac{1}{2} + \frac{1}{p(\lambda)}$ on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$, for some polynomial p, using c bits of communication,*
- *There exists a negligible function μ such that any unbounded protocol that computes f on input distribution (X, Y) using c bits of communication has correctness at most $\frac{1}{2} + \mu(\lambda)$ for any sufficiently large λ,*

for some function $c = c(\lambda)$.

Theorem 4.11 (Interactive CC-homomorphic Encryption implies Weak Key Agreement Protocol)**.** *Assume there exists an interactive CC-homomorphic encryption scheme (Definition 4.10), then there exists a weak key agreement protocol.*

Informally, a key agreement protocol (a.k.a. key exchange protocol) enables two parties to agree on a secret key while communicating over a public channel. The weak correctness property requires that by the end of the protocol, both parties have the same secret key value w.p. $\frac{1}{2} + \frac{1}{p(\lambda)}$, for some polynomial p, and the security property requires that the public communication do not leak any information about the agreed secret key.

Informal proof. Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a Y-distributional secure interactive CC-homomorphic encryption scheme. Let $\pi_0 = (\mathsf{Alice}, \mathsf{Bob})$ be the polynomial-time interactive protocol computing $\mathsf{Ext}_{\mathcal{E}}(f)$ with error $\frac{1}{2} + \frac{1}{\tau(\lambda)}$ on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$, for some polynomial τ.

Consider the following protocol π_1 between two parties P_1 and P_2 given a security parameter $\lambda \in \mathbb{N}$.

1. Party P_1 samples $k \leftarrow \mathsf{Gen}(1^\lambda)$, $y \leftarrow Y_\lambda$ and $c \leftarrow \mathsf{Enc}_k(y)$, and sends (y, c) to P_2.
2. Party P_2 samples a random bit $b \leftarrow \{0, 1\}$ and $x \leftarrow X_\lambda$ and sends $\sigma = f(x, y) \oplus b$ to P_1.
3. Both parties run the protocol π_0 on input $\big((x, c), (y, k)\big)$ where P_1 plays the role of Bob and P_2 plays the role of Alice.
4. At the end, P_2 outputs b and P_1 outputs $\sigma \oplus \phi$ where ϕ is the output of π_0.

The weak correctness of π_1 follows by the weak correctness of π_0.

For security, assume towards a contradiction that there exists a polynomial-size adversary $\mathcal{A}$ that given the transcript of the protocol outputs the key with probability $\frac{1}{2} + \frac{1}{p(\lambda)}$, for some polynomial p.

Similarly to the proof of Theorem 4.1 we start by constructing an unbounded protocol with shared randomness and we will eliminate the shared randomness later. Consider the following *probabilistic public-coin* protocol π_2 between Alice and Bob who are given inputs x and y respectively sampled from (X, Y). Alice and Bob both generate a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and an encryption $c \leftarrow \mathsf{Enc}_k(y')$ for some $y' \leftarrow Y$ using the shared randomness. Then, they run the protocol π_0.

Finally, Bob feeds $(y, \mathsf{Enc}(y'))$, and random bit σ and the transcript of the protocol π into $\mathcal{A}$ and outputs $\sigma \oplus \phi$ where ϕ is the output of $\mathcal{A}$.

By the Y-distributional security of $\mathcal{E}$ we have that $\mathcal{A}$ still outputs the key with non-negligible advantage, even when $\mathsf{Enc}(y)$ is replaced by $\mathsf{Enc}(y')$. Therefore, we have that π_2 computes f with correctness $\frac{1}{2} + \frac{1}{p(\lambda)}$ and has communication complexity of c, in contradiction to the fact that any unbounded protocol that computes f on (X, Y) using c bits of communication, has a correctness error of at most $\frac{1}{2} + \mu(\lambda)$, where μ is negligible. □

5 Instantiations

5.1 LWE

In this section, we will construct a CC-homomorphic and VC-homomorphic encryption scheme from LWE. This construction derives Regev's scheme [Reg05] from our framework and gives an alternative to the proof by Rothblum [Rot11] using communication complexity (rather than Fourier analysis or Leftover Hash Lemma). We first present the learning with errors assumption.

Definition 5.1 (Learning With Errors Assumption)**.** *For an integer $q = q(\lambda)$ and an error distribution $\chi = \chi(\lambda)$ over $\mathbb{Z}_q$, the learning with errors assumption $\mathsf{LWE}_{q,\chi}$ is that for any $m(\lambda) = \lambda^{O(1)}$,*

$$(A, As + e)_{\lambda \in \mathbb{N}} \approx_c (A, u)_{\lambda \in \mathbb{N}},$$

where $A \leftarrow \mathbb{F}_q^{m \times \lambda}$, $s \leftarrow \mathbb{F}_q^{\lambda}$, $e \leftarrow \chi^m$ and $u \leftarrow \mathbb{F}_q^m$.

We consider the operation $(a \mod q)$ as mapping the integer a into the interval $(-\frac{q}{2}, \frac{q}{2}]$. We say that χ is *B-bounded* if $|a| < B$ for any $a \leftarrow \chi$. For the following, set $B = \frac{q}{4\lambda^2 \cdot \log^2 q}$ for any $\lambda \in \mathbb{N}$ and integer $q = q(\lambda)$.

Theorem 5.2 (Combinatorially Homomorphic Encryption from LWE)**.** *Assuming $\mathsf{LWE}_{q,\chi}$ (Definition 5.1), where χ is B-bounded, there exists a CC-homomorphic encryption scheme in the perfect correctness regime (Definition 3.2) and a VC-homomorphic encryption scheme (Definition 3.12).*

In fact, we will construct a CC-homomorphic encryption scheme that satisfies the conditions of Theorem 4.7, thus deriving the following three theorems.

Theorem 5.3 (Lossy Encryption from LWE)**.** *Assuming $\mathsf{LWE}_{q,\chi}$ (Definition 5.1), where χ is B-bounded, there exists a lossy encryption scheme (Definition 2.10).*

Theorem 5.4 (CRH from LWE)**.** *Assuming $\mathsf{LWE}_{q,\chi}$ (Definition 5.1), where χ is B-bounded, there exists a collision resistant hash function (Definition 2.13).*

Theorem 5.5 (PIR from LWE)**.** *Assuming $\mathsf{LWE}_{q,\chi}$ (Definition 5.1), where χ is B-bounded, there exists a private information retrieval protocol (Definition 2.14).*

Theorems 5.3 to 5.5 follows directly from Theorems 4.1, 4.7, 4.9 and 5.2. We now describe a private-key encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ based on LWE with parameters $q = q(\lambda)$ and $\chi = \chi(\lambda)$. Set $m = m(\lambda) = \lambda^2 \cdot \log^2 q$.

- **Key generation.** Given a security parameter 1^λ, the probabilistic algorithm Gen outputs a private key $s \leftarrow \mathbb{F}_q^{\lambda}$.

- **Encryption.** Given a bit b and a private key s, the probabilistic algorithm Enc samples $a \leftarrow \mathbb{Z}_q^\lambda$ and a random noise $e \leftarrow \chi$, and outputs the ciphertext $(a, \langle a, s\rangle + e + \lfloor q/2 \rfloor \cdot b)$.
- **Decryption.** Given a ciphertext (a, b), the deterministic algorithm Dec computes $z = b - \langle a, s\rangle$ and outputs 0 iff $|z| \leq q/4$.

We will show that $\mathcal{E}$ is CC-homomorphic with respect to the inner product functionality $f = \left(f_\lambda(x, y) = x^\top y\right)_{\lambda\in\mathbb{N}}$ over the uniform input distribution (X, Y) where $X = Y = \mathbb{F}_2^m$. Looking ahead, we will construct a polynomial-time protocol for $\mathsf{Ext}_{\mathcal{E}}(f)$ with perfect correctness on $\mathsf{Ext}_{\mathcal{E}}(X, Y)$ that uses $c = c(\lambda) = (\lambda + 1) \cdot \log q$ bits of communication.

Notice that $\left(\left\{f(\cdot, y) \;:\; y \in Y_\lambda\right\}\right)_{\lambda\in\mathbb{N}}$ is a universal hash function family. Furthermore, the ensemble X supports an efficient encoding with input length $m \geq c$ using the identity function, and there exists an efficient shattering scheme (Definition 3.9) for f with parameter c (take the set of m unit vectors over $\mathbb{F}_2^m$ as the shattered set, and find a function for any assignment efficiently using Gaussian elimination). In addition, it is known that any unbounded one-way protocol that computes f on (X, Y) using c bits of communication has correctness at most $1 - \frac{1}{p(\lambda)}$, for some polynomial p [RY20, Theorem 5.6].

First, we will show that the private-key encryption scheme $\mathcal{E}$ is CPA secure (Definition 2.9).

Claim 5.6 (CPA Security of $\mathcal{E}$)**.** *Assuming* $\mathsf{LWE}_{q,\chi}$ *(Definition 5.1), for every* $\lambda \in \mathbb{N}$ *we have that,*

$$\left(\mathsf{Enc}_s(0)\right)_{\lambda\in\mathbb{N}} \approx_c \left(\mathsf{Enc}_s(1)\right)_{\lambda\in\mathbb{N}},$$

where $s \leftarrow \mathsf{Gen}(1^\lambda)$.

Proof. For any $b \in \{0, 1\}$,

$$\left(\mathsf{Enc}_s(b)\right)_{\lambda\in\mathbb{N}} = \left((a, \langle a, s\rangle + e + \lfloor q/2 \rfloor \cdot b)\right)_{\lambda\in\mathbb{N}} \underset{(*)}{\approx_c} \left((a, u + \lfloor q/2 \rfloor \cdot b)\right)_{\lambda\in\mathbb{N}} = \left((a, u)\right)_{\lambda\in\mathbb{N}}$$

where $u \leftarrow \mathbb{F}_2^m$, $a, s \leftarrow \mathbb{F}_2^\lambda$ and $e \leftarrow \chi$, and where (*) holds by the $\mathsf{LWE}_{q,\chi}$ assumption. □

Now, consider the following polynomial-time one-way protocol for the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$. Alice is given a bit-by-bit encryption of Bob's input. Namely, ciphertexts $c_1, \ldots, c_m$ such that $c_i = \left(a_i, \langle a_i, s\rangle + e_i + \lfloor q/2 \rfloor \cdot y_i\right)$. Alice computes $m_A = (a', \sigma')$, where $a' = \sum_i x_i \cdot a_i$ and $\sigma' = \sum_i x_i \cdot \left(\langle a_i, s\rangle + e_i + \lfloor q/2 \rfloor \cdot y_i\right)$ and sends m_A to Bob, who outputs $\mathsf{Dec}_s(m_A)$.

The communication cost of this protocol is $c(\lambda) = |m_A| = (\lambda + 1) \cdot \log q$.

Claim 5.7 (Protocol Correctness)**.** *For every* $\lambda \in \mathbb{N}$*,* $x \in X$ *and* $y \in Y$ *we have that*

$$\Pr\left[\mathsf{Dec}_s\Big(\mathsf{Alice}\big(x, \mathsf{Enc}_s(y_i)_{i\in[m]}\big)\Big) = x^\top \cdot y : s \leftarrow \mathsf{Gen}(1^\lambda)\right] = 1,$$

Proof. For every $\lambda \in \mathbb{N}$, $x \in X$ and $y \in Y$ we have that

$$\Pr\left[\mathsf{Dec}_s\Big(\mathsf{Alice}\big(x, \mathsf{Enc}_s(y_i)_{i\in[m]}\big)\Big) = x^\top \cdot y\right] \underset{(1)}{=} \Pr\left[\sum_{i=1}^m x_i \cdot e_i < q/4\right] \underset{(2)}{=} 1,$$

where $s \leftarrow \mathsf{Gen}(1^\lambda)$ and $e \leftarrow \chi^m$, and where (1) is by the scheme's definition and (2) is since χ is $B = \frac{q}{4m}$-bounded. □

5.2 Low Noise LPN

In this section we will construct a CC-homomorphic encryption scheme from low noise LPN, thereby giving a conceptually simple derivation of recent results [BLVW19,YZW+19,BF22]. We first present the learning parity with noise assumption. For $\mu \in [0,1]$ we denote by Ber_μ the Bernoulli distribution with mean μ.

Definition 5.8 (Learning Parity with Noise Assumption)**.** *For noise rate $\mu = \mu(\lambda) \in (0, \frac{1}{2})$, the LPN_μ assumption is that for any $m(\lambda) = \lambda^{O(1)}$,*

$$(A, As + e)_{\lambda\in\mathbb{N}} \approx_c (A, u)_{\lambda\in\mathbb{N}},$$

where $A \leftarrow \mathbb{F}_2^{m\times\lambda}$, $s \leftarrow \mathbb{F}_2^\lambda$, $e \leftarrow \mathsf{Ber}_\mu^m$ and $u \leftarrow \mathbb{F}_2^m$.

Theorem 5.9 (CC-homomorphic Encryption from Low Noise LPN)**.** *Assuming $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ (Definition 5.8) there exists a CC-homomorphic encryption scheme in the statistical hardness regime (Definition 3.5).*

In fact, we will construct a CC-homomorphic encryption scheme that satisfies the conditions of Theorem 4.7, thus deriving the following two theorems.

Theorem 5.10 (Lossy Encryption from Low Noise LPN)**.** *Assuming $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ (Definition 5.8) there exists a lossy encryption scheme (Definition 2.10).*

Theorem 5.11 (CRH from Low Noise LPN)**.** *Assuming $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ (Definition 5.8) there exists a collision resistant hash function (Definition 2.13).*

Theorems 5.10 and 5.11 follows directly from Theorems 4.1, 4.7 and 5.9. We note however that we do not know how to use LPN to derive a similar result to Alekhnovich's scheme [Ale03] via our framework. Indeed, the stronger conclusions implied by our framework (lossy encryption, CRH) are not known from the flavor of LPN used by Alekhnovich. In addition, we note that we do not know how to construct a VC-homomorphic encryption scheme (Definition 3.12) even from the very mild $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ assumption, thus not achieving PIR via Theorem 4.9. See further discussion in Section 5.2.

We now describe a private-key encryption scheme $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ based on low noise LPN.

- **Key generation.** Given a security parameter 1^λ, the probabilistic algorithm Gen outputs a private key $s \leftarrow \mathbb{F}_2^\lambda$.
- **Encryption.** Given a message $y \in \mathbb{F}_2^{\lambda^2}$ and a private key s, the probabilistic algorithm Enc samples a random matrix $A \leftarrow \mathbb{F}_2^{\lambda^2\times\lambda}$ and a random noise $e \leftarrow \mathsf{Ber}_{\frac{\log^2\lambda}{\lambda}}^{\lambda^2}$, and outputs a ciphertext $(A, A\cdot s + e + y)$.
- **Decryption.** Given a ciphertext (A, b), the deterministic algorithm Dec outputs $b - A\cdot s$.

We define the following homomorphic operation that supports ciphertext-plaintext multiplication.

- **Ciphertext-plaintext multiplication.** Given a plaintext $x \in \mathbb{F}_2^{\lambda^2}$ and a ciphertext (A, b), where $A \in \mathbb{F}_2^{\lambda^2\times\lambda}$ and $b \in \mathbb{F}_2^{\lambda^2}$, the deterministic algorithm $\mathsf{PlainMult}$ outputs $(x^\top \cdot A, x^\top \cdot b)$.

We will show that $\mathcal{E}$ is CC-homomorphic with respect to the inner product functionality $f = \left(f_\lambda(x,y) = x^\top y\right)_{\lambda\in\mathbb{N}}$ over the uniform input distribution (X,Y) where X and Y contain vectors in $\mathbb{F}_2^{\lambda^2}$, while X_λ is restricted to vectors with Hamming weight $\frac{2\lambda}{\log\lambda}$. Looking ahead, we will construct a polynomial-time protocol for $\mathsf{Ext}_{\mathcal{E}}(f)$ with correctness $\frac{1}{2}+\frac{1}{p(\lambda)}$ on $\mathsf{Ext}_{\mathcal{E}}(X,Y)$, for some polynomial p, that uses $c = c(\lambda) = \lambda + 1$ bits of communication. Furthermore, we will show that there exists a negligible function μ such that any unbounded one-way protocol that computes f on (X,Y) using c bits of communication has correctness at most $\frac{1}{2} + \mu(\lambda)$, for any sufficiently large λ.

Notice that

$$\Pr\left[f(x,y) = 0 : (x,y) \leftarrow (X,Y)\right] = \frac{1}{2},$$

and that $\left(\left\{f(\cdot,y) \;:\; y \in Y_\lambda\right\}\right)_{\lambda\in\mathbb{N}}$ is a universal hash function family. Furthermore, the ensemble X supports an efficient encoding with input length $2\lambda \geq c$, for any sufficiently large λ. Namely, given a vector $m \in \mathbb{F}_2^{2\lambda}$ we map every $\log\lambda$ bits of m to a unit vector in $\mathbb{F}_2^{\lambda}$. Then, we concatenate these unit vectors to a vector in $\mathbb{F}_2^{\lambda^2}$ with Hamming weight $\frac{2\lambda}{\log\lambda}$.

First, we will show that the private-key encryption scheme $\mathcal{E}$ is Y-distributional secure (Definition 2.8).

Claim 5.12 (Y-Distributional Security of $\mathcal{E}$)**.** *Assuming* $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ *(Definition 5.8), for every* $\lambda\in\mathbb{N}$ *and* $y, y' \leftarrow \mathbb{F}_2^{\lambda^2}$ *we have that,*

$$\left(y, \mathsf{Enc}_s(y)\right)_{\lambda\in\mathbb{N}} \approx_c \left(y, \mathsf{Enc}_s(y')\right)_{\lambda\in\mathbb{N}},$$

where $s \leftarrow \mathsf{Gen}(1^\lambda)$.

Proof. For any fixed $y, y' \in \mathbb{F}_2^{\lambda^2}$,

$$\left(y, \mathsf{Enc}_s(y')\right)_{\lambda\in\mathbb{N}} = \left(y, (A, A\cdot s + e + y')\right)_{\lambda\in\mathbb{N}} \underset{(*)}{\approx_c} \left(y, (A, u + y')\right)_{\lambda\in\mathbb{N}} = \left(y, (A,u)\right)_{\lambda\in\mathbb{N}}$$

where $u \leftarrow \mathbb{F}_2^{\lambda^2}$, $s \leftarrow \mathbb{F}_2^{\lambda}$, $A \leftarrow \mathbb{F}_2^{\lambda^2\times\lambda}$ and $e \leftarrow \mathsf{Ber}^{\lambda^2}_{\frac{\log^2\lambda}{\lambda}}$, and where (*) holds by the $\mathsf{LPN}_{\frac{\log^2\lambda}{\lambda}}$ assumption. □

Now, consider the following polynomial-time one-way protocol for the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$. Given inputs x and $c = \mathsf{Enc}_k(y)$, Alice computes $m_A = \mathsf{PlainMult}(x,c)$ and sends it to Bob, who outputs $\mathsf{Dec}_k(m_A)$.

The communication cost of this protocol is $c(\lambda) = |m_A| = \lambda + 1$. We show the correctness probability of the protocol using the Piling-Up Lemma.

Lemma 5.13 (The Piling-Up Lemma [Mat94])**.** *Let* $e_1, ..., e_k \in \mathbb{F}_2$ *be i.i.d. random variables such that* $\Pr[e_i = 1] = \epsilon$*, then*

$$\Pr\left[\bigoplus_{i=1}^{k} e_i = 0\right] = \frac{1}{2} + \frac{1}{2}(1-2\epsilon)^k.$$

Claim 5.14 (Protocol Correctness)**.** *For every* $\lambda\in\mathbb{N}$, $x \in X$ *and* $y \in Y$ *we have that*

$$\Pr\left[\mathsf{Dec}_s\Big(\mathsf{PlainMult}\big(x, \mathsf{Enc}_s(y)\big)\Big) = x^\top\cdot y : s \leftarrow \mathsf{Gen}(1^\lambda)\right] > \frac{1}{2} + \frac{1}{2\lambda^8}.$$

Proof. By the definition of $\mathcal{E}$ it's enough to show that $\Pr\left[x^\top \cdot e = 0\right] > \frac{1}{2} + \frac{1}{2\lambda^8}$. By Lemma 5.13 we have that

$$\Pr\left[x^\top \cdot e = 0\right] = \Pr\left[\bigoplus_{i=1}^{\frac{2\lambda}{\log\lambda}} e_i = 0\right] \geq \frac{1}{2} + \frac{1}{2}(1 - 2\frac{\log^2\lambda}{\lambda})^{\frac{2\lambda}{\log\lambda}} \geq \frac{1}{2} + \frac{1}{2} \cdot 2^{-4\frac{\log^2\lambda}{\lambda}\frac{2\lambda}{\log\lambda}} = \frac{1}{2} + \frac{1}{2\lambda^8},$$

where the second inequality holds since $1 - x \geq 2^{-2x}$ for $x \leq \frac{1}{2}$. □

Finally, we will show that for the negligible function $\mu = 2^{-\lambda}$ we have that any unbounded one-way protocol that computes f on input distribution (X, Y) using $c(\lambda) = \lambda + 1$ bits of communication has correctness at most $\frac{1}{2} + \mu(\lambda)$, for any sufficiently large λ.

Claim 5.15 (Distributional Communication Complexity Lower Bound for f)**.** *For any $\lambda \in \mathbb{N}$,*

$$\mathcal{D}^{A\to B}\big(f, (X, Y), \frac{1}{2} - 2^{-\lambda}\big) = 2\lambda$$

Proof. Take $\lambda \in \mathbb{N}$. Let H be a matrix such that $H(x, y) = (-1)^{<x,y>}$. It is easy to check that the matrix H satisfies $HH^\top = H^\top H = 2^{\lambda^2} I$. Therefore, $\|H\| = \sqrt{2^{\lambda^2}}$. Let $R = S \times T$ be a rectangle on (X_λ, Y_λ). We have that

$$\begin{aligned}
\mathsf{Disc}\big(f_\lambda; S \times T\big) &\underset{(1)}{=} \left|\sum_{(x,y)\in S\times T} \Pr\left[x, y \in (X, Y)\right](-1)^{<x,y>}\right| \\
&\underset{(2)}{\leq} \left|\sum_{(x,y)\in S\times T} \frac{1}{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}} \frac{1}{2^{\lambda^2}} H(x, y)\right| \\
&= \frac{1}{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}} \frac{1}{2^{\lambda^2}} \left|\mathbb{1}_S \cdot H \cdot \mathbb{1}_T\right| \\
&\underset{(3)}{\leq} \frac{1}{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}} \frac{1}{2^{\lambda^2}} \|\mathbb{1}_S\| \cdot \|H\| \cdot \|\mathbb{1}_T\| \\
&\underset{(4)}{\leq} \frac{1}{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}} \frac{1}{2^{\lambda^2}} \sqrt{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}} \cdot 2^{\frac{\lambda^2}{2}} \cdot 2^{\frac{\lambda^2}{2}} \\
&= \frac{1}{\sqrt{\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}}},
\end{aligned}$$

where (1) is by definition, (2) is since X_λ and Y_λ are independent and distributed uniformly over vectors with Hamming weight $\frac{2\lambda}{\log\lambda}$ in $\mathbb{F}_2^{\lambda^2}$ and over $\mathbb{F}_2^{\lambda^2}$ respectively, (3) is by Cauchy–Schwarz and (4) is since $\|H\| = \sqrt{2^{\lambda^2}}$ and since S and T can contain at most $\binom{\lambda^2}{\frac{2\lambda}{\log\lambda}}$ and 2^{λ^2} elements respectively.

Therefore, by Theorem 2.6 we have for error-rate $\epsilon(\lambda) = \frac{1}{2} - 2^{-\lambda}$ the following,

$$\begin{aligned}
\mathcal{D}^{A\to B}(f) &\geq \log\left(\frac{1-2\epsilon(\lambda)}{\mathsf{Disc}\big(f,(X,Y)\big)}\right) \\
&\geq \frac{1}{2}\log\left(\frac{\lambda^2}{\frac{2\lambda}{\log\lambda}}\right) - \lambda \\
&\underset{(*)}{=} \frac{\lambda}{\log\lambda}\cdot\log\left(\frac{1}{2}\lambda\log\lambda\right) - \lambda \\
&\geq 2\lambda
\end{aligned}$$

where (*) is since $\binom{n}{k} \geq (\frac{n}{k})^k$ for any n and k. □

Challenges with instantiating VC-homomorphic encryption from low noise LPN. Recall that to instantiate VC-homomorphic encryption (Definition 3.12), we need the homomorphic protocol to have negligible correctness error and a communication cost that is smaller than the VC dimension of f_X. Informally, the problem of using a similar idea as in the proof of Theorem 5.9 to construct such protocol, stems from the following observations. For the homomorphic protocol to have negligible correctness error, we need the correctness error of the private-key scheme to be negligible. As shown in the proof of Claim 5.14, to get negligible correctness error we need the Hamming weight of Alice's input x to be $<< \frac{\lambda}{\log\lambda}$. However, the VC dimension of the inner product functionality over inputs with such low Hamming weight is $<< \lambda$, which is the cost of the homomorphic protocol.

Acknowledgements. We thank Aayush Jain and the TCC reviewers for their helpful comments. Y. Ishai was supported in part by ERC Project NTSC (742754), BSF grant 2018393, and ISF grant 2774/20. R. Rothblum is funded by the European Union (ERC, FASTPROOF, 101041208). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

[Ale03] Michael Alekhnovich. More on average case vs approximation complexity. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 298–307. IEEE Computer Society, 2003.

[AMP19] Navid Alamati, Hart Montgomery, and Sikhar Patranabis. Symmetric primitives with structured secrets. In *Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part I 39*, pages 650–679. Springer, 2019.

[AMPR19] Navid Alamati, Hart Montgomery, Sikhar Patranabis, and Arnab Roy. Minicrypt primitives with algebraic structure and applications. In *Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part II 38*, pages 55–82. Springer, 2019.

[Ben94] Josh Benaloh. Dense probabilistic encryption. In *Selected Areas of Cryptography*, May 1994.

[BF22] Nir Bitansky and Sapir Freizeit. Statistically sender-private OT from LPN and derandomization. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part III*, volume 13509 of *Lecture Notes in Computer Science*, pages 625–653. Springer, 2022.

[BFS86] László Babai, Peter Frankl, and Janos Simon. Complexity classes in communication complexity theory. In *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pages 337–347. IEEE, 1986.

[BGN05] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-DNF formulas on ciphertexts. In Joe Kilian, editor, *Theory of Cryptography, Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005, Proceedings*, volume 3378 of *Lecture Notes in Computer Science*, pages 325–341. Springer, 2005.

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *Advances in Cryptology-EUROCRYPT 2009: 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings 28*, pages 1–35. Springer, 2009.

[BIP18] Elette Boyle, Yuval Ishai, and Antigoni Polychroniadou. Limits of practical sublinear secure computation. In *Advances in Cryptology–CRYPTO 2018: 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19–23, 2018, Proceedings, Part III 38*, pages 302–332. Springer, 2018.

[BL13] Andrej Bogdanov and Chin Ho Lee. Limits of provable security for homomorphic encryption. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 111–128. Springer, 2013.

[BLVW19] Zvika Brakerski, Vadim Lyubashevsky, Vinod Vaikuntanathan, and Daniel Wichs. Worst-case hardness for LPN and cryptographic hashing via code smoothing. In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III*, volume 11478 of *Lecture Notes in Computer Science*, pages 619–635. Springer, 2019.

[BV14] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. *SIAM J. Comput.*, 43(2):831–871, 2014.

[CCKM00] Christian Cachin, Jan Camenisch, Joe Kilian, and Joy Müller. One-round secure computation and secure autonomous mobile agents. In Ugo Montanari, José D. P. Rolim, and Emo Welzl, editors, *Automata, Languages and Programming, 27th International*

Colloquium, ICALP 2000, Geneva, Switzerland, July 9-15, 2000, Proceedings, volume 1853 of *Lecture Notes in Computer Science*, pages 512–523. Springer, 2000.

[CKGS98] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. Private information retrieval. *Journal of the ACM (JACM)*, 45(6):965–981, 1998.

[CMO00] Giovanni Di Crescenzo, Tal Malkin, and Rafail Ostrovsky. Single database private information retrieval implies oblivious transfer. In Bart Preneel, editor, *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, volume 1807 of *Lecture Notes in Computer Science*, pages 122–138. Springer, 2000.

[CN22] Shahar P. Cohen and Moni Naor. Low communication complexity protocols, collision resistant hash functions and secret key-agreement protocols. In Yevgeniy Dodis and Thomas Shrimpton, editors, *Advances in Cryptology - CRYPTO 2022 - 42nd Annual International Cryptology Conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15-18, 2022, Proceedings, Part III*, volume 13509 of *Lecture Notes in Computer Science*, pages 252–281. Springer, 2022.

[DJ01] Ivan Damgård and Mads Jurik. A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In Kwangjo Kim, editor, *Public Key Cryptography, 4th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2001, Cheju Island, Korea, February 13-15, 2001, Proceedings*, volume 1992 of *Lecture Notes in Computer Science*, pages 119–136. Springer, 2001.

[DNR04] Cynthia Dwork, Moni Naor, and Omer Reingold. Immunizing encryption schemes from decryption errors. In *Advances in Cryptology-EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004. Proceedings 23*, pages 342–360. Springer, 2004.

[ElG84] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer, 1984.

[Gen09] Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, USA, 2009.

[GHV10a] Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. i-hop homomorphic encryption and rerandomizable Yao circuits. In Tal Rabin, editor, *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, volume 6223 of *Lecture Notes in Computer Science*, pages 155–172. Springer, 2010.

[GHV10b] Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. A simple BGN-type cryptosystem from LWE. In Henri Gilbert, editor, *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic*

Techniques, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings, volume 6110 of *Lecture Notes in Computer Science*, pages 506–522. Springer, 2010.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[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 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I*, volume 8042 of *Lecture Notes in Computer Science*, pages 75–92. Springer, 2013.

[HHS08] Iftach Haitner, Jonathan J Hoch, and Gil Segev. A linear lower bound on the communication complexity of single-server private information retrieval. In *Theory of Cryptography: Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008. Proceedings 5*, pages 445–464. Springer, 2008.

[HK07] Omer Horvitz and Jonathan Katz. Universally-composable two-party computation in two rounds. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings*, volume 4622 of *Lecture Notes in Computer Science*, pages 111–129. Springer, 2007.

[HLOV11] Brett Hemenway, Benoit Libert, Rafail Ostrovsky, and Damien Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In *Advances in Cryptology–ASIACRYPT 2011: 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings 17*, pages 70–88. Springer, 2011.

[HR05] Thomas Holenstein and Renato Renner. One-way secret-key agreement and applications to circuit polarization and immunization of public-key encryption. In *Annual International Cryptology Conference*, pages 478–493. Springer, 2005.

[IKO05] Yuval Ishai, Eyal Kushilevitz, and Rafail Ostrovsky. Sufficient conditions for collision-resistant hashing. In *Theory of Cryptography: Second Theory of Cryptography Conference, TCC 2005, Cambridge, MA, USA, February 10-12, 2005. Proceedings 2*, pages 445–456. Springer, 2005.

[IKO+11] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, Manoj Prabhakaran, and Amit Sahai. Efficient non-interactive secure computation. In Kenneth G. Paterson, editor, *Advances in Cryptology - EUROCRYPT 2011 - 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Tallinn, Estonia, May 15-19, 2011. Proceedings*, volume 6632 of *Lecture Notes in Computer Science*, pages 406–425. Springer, 2011.

[IP07] Yuval Ishai and Anat Paskin. Evaluating branching programs on encrypted data. In Salil P. Vadhan, editor, *Theory of Cryptography, 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007, Proceedings*, volume 4392 of *Lecture Notes in Computer Science*, pages 575–594. Springer, 2007.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication complexity*. Cambridge University Press, 1997.

[KNR99] Ilan Kremer, Noam Nisan, and Dana Ron. On randomized one-round communication complexity. *Computational Complexity*, 8:21–49, 1999.

[KO97] Eyal Kushilevitz and Rafail Ostrovsky. Replication is NOT needed: SINGLE database, computationally-private information retrieval. In *38th Annual Symposium on Foundations of Computer Science, FOCS '97, Miami Beach, Florida, USA, October 19-22, 1997*, pages 364–373. IEEE Computer Society, 1997.

[Lip05] Helger Lipmaa. An oblivious transfer protocol with log-squared communication. In Jianying Zhou, Javier López, Robert H. Deng, and Feng Bao, editors, *Information Security, 8th International Conference, ISC 2005, Singapore, September 20-23, 2005, Proceedings*, volume 3650 of *Lecture Notes in Computer Science*, pages 314–328. Springer, 2005.

[LV16] Tianren Liu and Vinod Vaikuntanathan. On basing private information retrieval on np-hardness. In Eyal Kushilevitz and Tal Malkin, editors, *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, volume 9562 of *Lecture Notes in Computer Science*, pages 372–386. Springer, 2016.

[Mat94] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology—EUROCRYPT'93: Workshop on the Theory and Application of Cryptographic Techniques Lofthus, Norway, May 23–27, 1993 Proceedings 12*, pages 386–397. Springer, 1994.

[NS98] David Naccache and Jacques Stern. A new public key cryptosystem based on higher residues. In Li Gong and Michael K. Reiter, editors, *CCS '98, Proceedings of the 5th ACM Conference on Computer and Communications Security, San Francisco, CA, USA, November 3-5, 1998*, pages 59–66. ACM, 1998.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *Advances in Cryptology - EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 1999.

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In *Advances in Cryptology–CRYPTO 2008: 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings 28*, pages 554–571. Springer, 2008.

[RAD78] R L Rivest, L Adleman, and M L Dertouzos. On data banks and privacy homomorphisms. *Foundations of Secure Computation, Academia Press*, pages 169–179, 1978.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM*

Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005, pages 84–93. ACM, 2005.

[Rot11] Ron Rothblum. Homomorphic encryption: From private-key to public-key. In Yuval Ishai, editor, *Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011, Providence, RI, USA, March 28-30, 2011. Proceedings*, volume 6597 of *Lecture Notes in Computer Science*, pages 219–234. Springer, 2011.

[RY20] Anup Rao and Amir Yehudayoff. *Communication Complexity: and Applications*. Cambridge University Press, 2020.

[Ste98] Julien P. Stern. A new efficient all-or-nothing disclosure of secrets protocol. In Kazuo Ohta and Dingyi Pei, editors, *Advances in Cryptology - ASIACRYPT '98, International Conference on the Theory and Applications of Cryptology and Information Security, Beijing, China, October 18-22, 1998, Proceedings*, volume 1514 of *Lecture Notes in Computer Science*, pages 357–371. Springer, 1998.

[SV97] Amit Sahai and Salil P Vadhan. Manipulating statistical difference. In *Randomization Methods in Algorithm Design*, pages 251–270, 1997.

[SYY99] Tomas Sander, Adam L. Young, and Moti Yung. Non-interactive cryptocomputing for nc^1. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, pages 554–567. IEEE Computer Society, 1999.

[Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 209–213, 1979.

[YZW+19] Yu Yu, Jiang Zhang, Jian Weng, Chun Guo, and Xiangxue Li. Collision resistant hashing from sub-exponential learning parity with noise. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 3–24. Springer, 2019.

A CC-homomorphic Encryption Generalization

In this section, we present generalizations of the CC-homomorphism definitions that we introduced in Section 3.1. Recall that we required there that the function f is *balanced* (wrt to the given joint distribution (X, Y)), in the sense that,

$$\Pr\left[f(x,y) = 0 : (x,y) \leftarrow (X,Y)\right] = \frac{1}{2}.$$

A quick intuition for the restriction above is that in our applications, and specifically in our lossy encryption construction (Theorem 4.1), we use the output of f on some randomly sampled input to mask the encrypted bit. In the security proof, we claim that no adversary can break the encryption scheme — i.e., distinguish between encryptions of 0 and encryptions of 1 — since the existence of such an adversary yields a protocol that breaks the communication lower bound of f (which is, of course, impossible).

However, the standard communication complexity definition deals with the communication lower bounds of protocols that *predict* the output of the function, while the adversary of the encryption scheme can only *distinguish* between the different encryptions. Therefore, to construct the protocol that would "break" the communication lower bound and would prove that the aforementioned adversary cannot actually exist, we need to make a distinguishability to predictability transformation. Such a transformation is common in cryptography (for example, Yao's next bit predictor), however, it requires the random variable to be distributed uniformly, and this limits us balanced functions.

The purpose of the generalized definition of this section is to remove this restriction. Meaning, to handle the case of *imbalanced* functions. For this, we present a new *distinguishability*-based definition for communication complexity (in contrast to the standard *predictability*-based definition we discussed above). Intuitively, instead of requiring the protocol to help the parties predict the output of the function, we will require the protocol to provide advantage in distinguishing between the real output of the function and a "simulated" output.

For a given one-way protocol π, we denote by $\mathsf{View}_{\mathsf{Bob}}(x,y)$ the view of Bob in the protocol (i.e., Bob's input and Alice's message) when running on input (x,y).

Next, we define the *advantage* of a protocol. Intuitively, the advantage of a protocol is the ability of a Bob, given its *view*, to distinguish between the real output of the function, on the given inputs, and a "simulated" output of the function on a semi-random inputs (semi, because we only re-sample Alice's input).

Definition A.1 (Protocol Advantage)**.** *Given a function ensemble f and a product input distribution ensemble (X,Y), we say that a protocol π computes f with advantage $\delta = \delta(\lambda)$ on (X,Y), if there exists a distinguisher $\mathcal{D}$ such that for any $\lambda \in \mathbb{N}$,*

$$\left| \Pr\left[\mathcal{D}\big(\mathsf{View}_{\mathsf{Bob}}(x,y), f(x,y)\big) = 1\right] - \Pr\left[\mathcal{D}\big(\mathsf{View}_{\mathsf{Bob}}(x,y), f(x',y)\big) = 1\right] \right| \geq \delta(\lambda),$$

where $x, x' \leftarrow X$ and $y \leftarrow Y$. If the protocol is polynomial-time we require $\mathcal{D}$ to be polynomial-time as well.

We now introduce a distinguishing based definition for one-way distributional communication complexity.

Definition A.2 (Distinguishing-based Distributional Communication Complexity)**.** *Given a function f and a joint input distribution (X,Y) we define the* δ-advantage distinguishing-based (X,Y)-distributional communication complexity *of f as follows.*

$$\min_{\substack{\pi \text{ computes } f \\ \text{with advantage } \delta \\ \text{on } (X,Y)}} \mathsf{CC}[\pi,(X,Y)].$$

We now define CC-homomorphism with respect to the distinguishing-based definition for distributional communication complexity (Definition A.2).

Definition A.3 (Generalized Communication Complexity Homomorphic Encryption)**.** *A private-key encryption scheme $\mathcal{E}$ (Definition 2.8) is* communication-complexity homomorphic *(or* CC-homomorphic*), if there exists a function ensemble f, an efficiently sampleable* product *distribution ensemble (X,Y) and functions $c = c(\lambda)$, $\delta = \delta(\lambda)$ and $\delta' = \delta'(\lambda)$ such that,*

- *There exists a polynomial-time one-way protocol that computes* $\mathsf{Ext}_{\mathcal{E}}(f)$ *with advantage* δ *on input distribution* $\mathsf{Ext}_{\mathcal{E}}(X,Y)$*, using* c *bits of communication,*
- *Any unbounded one-way protocol that computes* f *on* (X,Y)*, using* c *bits of communication has advantage at most* δ'*.*

We require that there exist a negligible function μ *and a polynomial* p *such that one of the following conditions is satisfied.*

- $\delta(\lambda) = 1 - \mu(\lambda)$ *and* $\delta'(\lambda) = 1 - 1/p(\lambda)$*. In this case, we say that* $\mathcal{E}$ *is CC-homomorphic in the* prefect correctness regime*.*
- $\delta(\lambda) = 1/p(\lambda)$ *and* $\delta'(\lambda) = \mu(\lambda)$*. In this case, we say that* $\mathcal{E}$ *is CC-homomorphic in the* statistical hardness regime*.*

We will now show that any CC-homomorphic encryption implies lossy encryption.

Theorem A.4 (CC-homomorphic Encryption implies Lossy Encryption)**.** *Assume there exists a CC-homomorphic encryption scheme (Definition A.3), in either the perfect correctness regime or the statistical hardness regime, then there exists a lossy encryption scheme (Definition 2.10).*

Proof. We will use a construction similar to the one in the proof of Theorem 4.1, while utilizing the guaranteed distinguisher of Definition A.1. We will prove Theorem A.4 in the perfect correctness regime, while the proof in the statistical hardness regime is similar. Note that, using Lemma 2.12, it suffices to construct a scheme with weak lossiness.

Let $\mathcal{E} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a Y-distributional secure CC-homomorphic encryption scheme, in the perfect correctness regime, with respect to function ensemble f and input product distribution ensemble (X,Y). Let μ be a negligible function and let π be a polynomial-time one-way protocol computing the extended function ensemble $\mathsf{Ext}_{\mathcal{E}}(f)$ with advantage $1-\mu(\lambda)$ on $\mathsf{Ext}_{\mathcal{E}}(X,Y)$ and communication cost $c = c(\lambda)$, such that any unbounded protocol that computes f on (X,Y) using c bits of communication has advantage at most $1 - \frac{1}{p(\lambda)}$, for some fixed polynomial p. Let $\mathcal{D}$ be the guaranteed polynomial-time distinguisher for the homomorphic protocol.

For the following, given input $\big((x,c),(y,k)\big)$ from $\mathsf{Ext}_{\mathcal{E}}(X,Y)$, we denote by $\mathsf{Alice}(x,c)$ the message Alice generates in the protocol and we denote by $\mathsf{Bob}(y,k,m_A)$ the output of Bob after receiving a message m_A from Alice. Consider the following scheme $(\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*, \mathsf{LossyGen}^*)$.

- **Key generation.** Given a security parameter 1^λ the probabilistic polynomial-time algorithm Gen^* samples a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and an element $y \leftarrow Y$, and outputs the public key $pk = \big(y, \mathsf{Enc}_k(y)\big)$ and the secret key $sk = (y,k)$.
- **Encryption.** Given the public key $pk = (y,c)$ and a bit b, the probabilistic polynomial-time algorithm Enc^* samples $x, x' \leftarrow X$ and $\sigma \leftarrow \{0,1\}$, and outputs $\big(\mathsf{Alice}(x,c), f(z,y), b \oplus \sigma\big)$, where $z = x$ in case $\sigma = 0$ and $z = x'$ otherwise.
- **Decryption.** Given the secret key $sk = (y,k)$ and a ciphertext (m_A, α, β), the deterministic polynomial-time algorithm Dec^* outputs $\mathcal{D}(y,k,m_A,\alpha) \oplus \beta$.
- **Lossy Key generation.** Given a security parameter 1^λ the probabilistic polynomial-time algorithm $\mathsf{LossyGen}^*$ samples a key $k \leftarrow \mathsf{Gen}(1^\lambda)$ and elements $y, y' \leftarrow Y$, and outputs the lossy key $lk = \big(y, \mathsf{Enc}_k(y')\big)$.

Claim A.5. *The scheme satisfies correctness (see Definition 2.10).*

Proof. For any $\lambda \in \mathbb{N}$ and $b \in \{0,1\}$,

$$\begin{aligned}\Pr\left[\mathsf{Dec}^*_{sk}\big(\mathsf{Enc}^*_{pk}(b)\big) = b\right] &\underset{(1)}{=} \Pr\left[\mathcal{D}\big(y,k,\mathsf{Alice}(x,c),\alpha\big) = \sigma\right] \\ &\underset{(2)}{=} \frac{1}{2}\cdot\Pr\left[\mathcal{D}\big(y,k,\mathsf{Alice}(x,c),f(x,y)\big) = 1\right] \\ &\quad + \frac{1}{2}\cdot\Pr\left[\mathcal{D}\big(y,k,\mathsf{Alice}(x,c),f(x',y)\big) = 0\right] \\ &\underset{(3)}{\geq} 1 - \frac{\mu(\lambda)}{2},\end{aligned}$$

where $\sigma \leftarrow \{0,1\}$, $(sk,pk) \leftarrow \mathsf{Gen}^*(1^\lambda)$, $k \leftarrow \mathsf{Gen}(1^\lambda)$, $c \leftarrow \mathsf{Enc}_k(y)$, $x,x' \leftarrow X$, $y \leftarrow Y$ and $\alpha = f(z,y)$, where $z = x$ in case $\sigma = 0$ and $z = x'$ otherwise, and where (1) is by the definition of the scheme, (2) is by the law of total probability, and (3) is since the protocol π computes $\mathsf{Ext}_{\mathcal{E}}(f)$ on $\mathsf{Ext}_{\mathcal{E}}(X,Y)$ with advantage $1-\mu(\lambda)$. □

Claim A.6. *The scheme satisfies key indistinguishability (see Definition 2.10).*

The proof, which follows from the Y-distributional security, is similar to that in Theorem 4.1 and is omitted.

Claim A.7. *The scheme satisfies weak-lossiness of lossy keys (see Definition 2.10).*

Proof. Consider the (computationally unbounded) protocol $\pi^* = (\mathsf{Alice}^*, \mathsf{Bob}^*)$ introduced in the proof of Theorem 4.1, that computes f using c bits of communication. Assume towards a contradiction that there exists a negligible function ν such that for any sufficiently large $\lambda \in \mathbb{N}$,

$$\mathsf{SD}\Big(\big(lk,\mathsf{Enc}^*_{lk}(1)\big),\big(lk,\mathsf{Enc}^*_{lk}(0)\big)\Big) \geq 1-\nu(\lambda),$$

where $lk \leftarrow \mathsf{LossyGen}^*(1^\lambda)$. Therefore, by the definitions of $\mathsf{LossyGen}^*$ and Enc^* we have that for any sufficiently large $\lambda \in \mathbb{N}$,

$$\mathsf{SD}\Big(\big(y,c,\mathsf{Alice}^*(x,c),f(x,y)\big),\big(y,c,\mathsf{Alice}^*(x,c),f(x',y)\big)\Big) \geq 1-\nu(\lambda),$$

for $x,x' \leftarrow X$, $y,y' \leftarrow Y$, $k \leftarrow \mathsf{Gen}(1^\lambda)$ and $c \leftarrow \mathsf{Enc}_k(y')$. Meaning, that there exists a distinguisher $\mathcal{D}$ such that for any sufficiently large $\lambda \in \mathbb{N}$,

$$\left|\Pr\left[\mathcal{D}\big(\mathsf{View}_{\mathsf{Bob}^*}(x,y),f(x,y)\big)=1\right] - \Pr\left[\mathcal{D}\big(\mathsf{View}_{\mathsf{Bob}^*}(x,y),f(x',y)\big)=1\right]\right| \geq 1-\nu(\lambda),$$

where $x,x' \leftarrow X$ and $y \leftarrow Y$. However, this is a contradiction to the fact that π^* has advantage at most $1-\frac{1}{p(\lambda)}$ since $\mathcal{E}$ is CC-homomorphic in the statistical hardness regime. □

This concludes the proof of Theorem A.4. □

B Amplification for Weak Lossy Encryption

In this section we prove Lemma 2.11 and Lemma 2.12 which show how to upgrade a weak lossy encryption scheme to a full-fledged lossy scheme. For convenience, we first restate the claims.

Lemma 2.11 (Weak-Correctness Lossy Encryption implies Lossy Encryption)**.** *Assume there exists a lossy encryption scheme with correctness error $\frac{1}{2} - \frac{1}{p(\lambda)}$, for some polynomial p, then there exists a lossy encryption scheme (Definition 2.10).*

Lemma 2.12 (Weak-Lossiness Lossy Encryption implies Lossy Encryption)**.** *Assume there exists a $(1 - \frac{1}{p(\lambda)})$-lossy encryption scheme, for some polynomial p, then there exists a lossy encryption scheme (Definition 2.10).*

Lemma 2.11 considers lossy encryption schemes that are *weak* in the sense that the correctness of the decryption algorithm is merely noticeably better than a random guess. Lemma 2.12 considers schemes that are weak in the sense that the statistical distance between encryptions of 0 and encryptions of 1 under a lossy key is only noticeably bounded away from 1 (in contrast to a negligible statistical distance, as required by a full-fledged lossy encryption scheme). The lemmas show that in both of these extreme cases we can amplify the scheme to a full-fledged lossy encryption scheme (given that only *one* of the weakness conditions hold).

B.1 Proof of Lemma 2.11

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LossyGen})$ be a weak lossy encryption scheme with correctness at least $\frac{1}{2} + \frac{1}{\lambda^c}$ for some $c \in \mathbb{N}$. We set $\ell = \ell(\lambda) = \lambda^{2c+1}$. We denote by $\mathsf{Maj} : \{0,1\}^\ell \to \{0,1\}$ the majority function that outputs 1 if more than half of the input bits are 1, and 0 otherwise. Consider the scheme $(\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*, \mathsf{LossyGen}^*)$, defined as follows.

- **Key generation.** Given a security parameter 1^λ, the algorithm Gen^* samples $(sk_i, pk_i) \leftarrow \mathsf{Gen}(1^\lambda)$, for $i \in [\ell]$, and outputs the secret key $(sk_i)_{i\in[\ell]}$ and the public key $(pk_i)_{i\in[\ell]}$.
- **Encryption.** Given a bit b and a public key $pk = (pk_i)_{i\in[\ell]}$, the algorithm Enc^* samples ciphertexts $c_i \leftarrow \mathsf{Enc}_{pk_i}(b)$, for $i \in [\ell]$, and outputs the ciphertext $(c_i)_{i\in[\ell]}$.
- **Decryption.** Given a ciphertext $c = (c_i)_{i\in[\ell]}$ and a secret key $sk = (sk_i)_{i\in[\ell]}$, the algorithm Dec^* computes $b_i = \mathsf{Dec}_{sk_i}(c_i)$, for $i \in [\ell]$, and outputs $\mathsf{Maj}(b_1, ..., b_\ell)$.
- **Lossy key generation.** Given a security parameter 1^λ, the algorithm $\mathsf{LossyGen}^*$ samples $lk_i \leftarrow \mathsf{LossyGen}(1^\lambda)$, for $i \in [\ell]$, and outputs the lossy key $(lk_i)_{i\in[\ell]}$.

Correctness follows by the Chernoff bound.

Theorem B.1 (Chernoff-Hoeffding Bound)**.** *If $X_1, ..., X_n$ are i.i.d. random variables such that $X_i \in [0,1]$ and $\mathbb{E}[X_i] = p$ for every $i \in [n]$, then for any $\epsilon > 0$,*

$$\Pr\left[\left|\sum_{i=0}^{\ell} X_i - p \cdot n\right| > \epsilon \cdot n\right] < 2 \cdot e^{-2\epsilon^2 \cdot n}.$$

Claim B.2. *The scheme satisfies correctness (see Definition 2.10).*

Proof. We have that for any $b \in \{0,1\}$ and $\lambda \in \mathbb{N}$,

$$\begin{aligned}
\Pr\left[\mathsf{Dec}^*_{sk}\big(\mathsf{Enc}^*_{pk}(b)\big) = b\right] &\underset{(1)}{=} \Pr\left[\mathsf{Maj}\big(\mathsf{Dec}_{sk_1}(c_1), ..., \mathsf{Dec}_{sk_\ell}(c_\ell)\big) = b\right] \\
&\underset{(2)}{=} \Pr\left[\sum_{i=0}^{\ell} X_i \geq \frac{\ell}{2}\right] \\
&\geq \Pr\left[\left|\sum_{i=0}^{\ell} X_i - (\frac{1}{2} + \frac{1}{\lambda^c}) \cdot \ell\right| \leq \frac{\ell}{\lambda^c}\right] \\
&\underset{(3)}{\geq} 1 - 2e^{-2\frac{\ell}{\lambda^{2c}}} \\
&= 1 - 2e^{-2\lambda},
\end{aligned}$$

where $(sk, pk) \leftarrow \mathsf{Gen}^*(1^\lambda)$ and $(sk_i, pk_i) \leftarrow \mathsf{Gen}(1^\lambda)$, $c_i \leftarrow \mathsf{Enc}_{pk_i}(b)$, $X_i \sim \mathsf{Ber}_{\frac{1}{2}+\frac{1}{\lambda^c}}$ for $i \in [\ell]$, and where (1) is by the scheme's definition, (2) is by the definition of Maj and since the original scheme has correctness $\frac{1}{2} + \frac{1}{\lambda^c}$ and (3) is by Theorem B.1. □

Claim B.3. *The scheme satisfies key indistinguishability (see Definition 2.10).*

The claim follows by the transitivity of computational indistinguishability using a standard hybrid argument.

Claim B.4. *The scheme satisfies lossiness of lossy keys (see Definition 2.10).*

The claim follows by the triangle inequality for statistical distance.

B.2 Proof of Lemma 2.12

Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LossyGen})$ be a weak $(1 - \frac{1}{\lambda^c})$-lossy encryption scheme, for some $c \in \mathbb{N}$. We set $\ell = \ell(\lambda) = \lambda^{c+1}$. Consider the scheme $(\mathsf{Gen}^*, \mathsf{Enc}^*, \mathsf{Dec}^*, \mathsf{LossyGen}^*)$, defined as follows.

- **Key generation.** Given a security parameter 1^λ, the algorithm Gen^* samples $(sk_i, pk_i) \leftarrow \mathsf{Gen}(1^\lambda)$, for $i \in [\ell]$, and outputs the secret key $(sk_i)_{i\in[\ell]}$ and the public key $(pk_i)_{i\in[\ell]}$.
- **Encryption.** Given a bit b and a public key $pk = (pk_i)_{i\in[\ell]}$, the algorithm Enc^* samples uniformly random bits $b_1, ...b_\ell$ such that $\bigoplus_{i=1}^{\ell} b_i = b$ and ciphertexts $c_i \leftarrow \mathsf{Enc}_{pk_i}(b_i)$, for $i \in [\ell]$, and outputs the ciphertext $(c_i)_{i\in[\ell]}$.
- **Decryption.** Given a ciphertext $c = (c_i)_{i\in[\ell]}$ and a secret key $sk = (sk_i)_{i\in[\ell]}$, the algorithm Dec^* computes $b'_i = \mathsf{Dec}_{sk_i}(c_i)$, for $i \in [\ell]$, and outputs $\bigoplus_{i=1}^{\ell} b'_i$.
- **Lossy key generation.** Given a security parameter 1^λ, the algorithm $\mathsf{LossyGen}^*$ samples $lk_i \leftarrow \mathsf{LossyGen}(1^\lambda)$, for $i \in [\ell]$, and outputs the lossy key $(lk_i)_{i\in[\ell]}$.

Claim B.5. *The scheme satisfies correctness (see Definition 2.10).*

The claim follows by the (nearly) perfect correctness of the original scheme.

Claim B.6. *The scheme satisfies key indistinguishability (see Definition 2.10).*

The claim follows by the transitivity of computational indistinguishability using standard hybrid argument.

Lossiness under lossy keys follows from the following XOR lemma .

Lemma B.7 (XOR Lemma [SV97, Lemma 3.2])**.** *Fix any two distributions D_0 and D_1. For any bit b and $n \in \mathbb{N}$, denote by $D_b^{\oplus n}$ the distribution sampled as follows. Sample uniformly random bits $b_1, ..., b_n$ such that $\bigoplus_{i=1}^{n} = b$, sample $x_i \leftarrow D_{b_i}$ for each $i \in [n]$, and output $(x_1, ..., x_n)$. Then,*

$$\mathsf{SD}(D_0^{\oplus n}, D_1^{\oplus n}) = \mathsf{SD}(D_0, D_1)^n$$

Claim B.8. *The scheme satisfies lossiness of lossy keys (see Definition 2.10).*

Proof. For any $\lambda \in \mathbb{N}$,

$$\begin{aligned}
\mathsf{SD}\Big(\big(lk, \mathsf{Enc}^*_{lk}(0)\big), \big(lk, \mathsf{Enc}^*_{lk}(1)\big)\Big) &\underset{(1)}{=} \mathsf{SD}\Big(\big(lk_i, c_i^{(0)}\big)_{i\in[\ell]}, \big(lk_i, c_i^{(1)}\big)_{i\in[\ell]}\Big) \\
&\underset{(2)}{=} \mathsf{SD}\Big(\big(lk_0, c_0^{(0)}\big)\big(lk_0, c_0^{(1)}\big)\Big)^{\ell} \\
&\underset{(3)}{=} (1 - \frac{1}{\lambda^c})^{\lambda^{c+1}} \\
&\underset{(4)}{\leq} 2^{-\lambda}
\end{aligned}$$

where $lk \leftarrow \mathsf{LossyGen}^*(1^\lambda)$ and for $i \in [\ell]$ we have that $lk_i \leftarrow \mathsf{LossyGen}(1^\lambda)$ and $c_i^{(b)} \leftarrow \mathsf{Enc}_{lk_i}(b_i^{(b)})$ with uniformly random $b_i^{(b)}$ such that $\bigoplus_{i=1}^{\ell} b_i^{(b)} = b$ for $b \in \{0, 1\}$, and where (1) is by the scheme's definition, (2) is by Lemma B.7, (3) is since the original scheme is $(1 - \frac{1}{\lambda^c})$-lossy, and (4) is since $1 - x \leq 2^{-x}$ for $x \in [0, 1]$. □