

The Mother of All Leakages: How to Simulate Noisy Leakages via Bounded Leakage (Almost) for Free

Gianluca Brian* Antonio Faonio[†] Maciej Obremski[‡] João Ribeiro[§]
Mark Simkin[¶] Maciej Skórski^{||} Daniele Venturi^{**}

Abstract

We show that noisy leakage can be simulated in the information-theoretic setting using a single query of bounded leakage, up to a small statistical simulation error and a slight loss in the leakage parameter. The latter holds true in particular for one of the most used noisy-leakage models, where the noisiness is measured using the conditional average min-entropy (Naor and Segev, CRYPTO'09 and SICOMP'12).

Our reductions between noisy and bounded leakage are achieved in two steps. First, we put forward a new leakage model (dubbed the *dense leakage* model) and prove that dense leakage can be simulated in the information-theoretic setting using a single query of bounded leakage, up to small statistical distance. Second, we show that the most common noisy-leakage models fall within the class of dense leakage, with good parameters. We also provide a complete picture of the relationships between different noisy-leakage models, and prove a lower bound on the amount of bounded leakage required for simulation with sub-constant error, showing that our reductions are nearly optimal.

Our result finds applications to leakage-resilient cryptography, where we are often able to lift security in the presence of bounded leakage to security in the presence of noisy leakage, both in the information-theoretic and in the computational setting. Additionally, we show how to use lower bounds in communication complexity to prove that bounded-collusion protocols (Kumar, Meka, and Sahai, FOCS'19) for certain functions do not only require long transcripts, but also necessarily need to reveal enough information about the inputs.

Keywords: foundations, information theory, leakage-resilient cryptography.

*Sapienza University of Rome, Italy. Email: brian@di.uniroma1.it.

[†]EURECOM, Sophia Antipolis, France. Email: antonio.faonio@eurecom.fr.

[‡]National University of Singapore. Email: obremski.math@gmail.com.

[§]Imperial College London, UK. Email: j.lourenco-ribeiro17@imperial.ac.uk.

[¶]Aarhus University. Email: simkin@cs.au.dk.

^{||}University of Luxembourg. Email: maciej.skorski@uni.lu.

^{**}Sapienza University of Rome, Italy. Email: venturi@di.uniroma1.it.

1 Introduction

1.1 Background

The security analysis of cryptographic primitives typically relies on the assumption that the underlying secrets (including, e.g., secret keys and internal randomness) are uniformly random to the eyes of the attacker. In reality, however, this assumption may simply be false due to the presence of so-called side-channel attacks [Koc96, KJJ99, AARR03], where an adversary can obtain partial information (also known as leakage) on the secret state of an implementation of a cryptographic scheme, by exploiting physical phenomena.

Leakage-resilient cryptography [ISW03, MR04, DP08] aims at bridging this gap by allowing the adversary to launch leakage attacks in theoretical models too. The last decade has seen an impressive amount of work in this area, thanks to which we now dispose of a large number of leakage-resilient cryptographic primitives in different leakage models. We refer the reader to the recent survey by Kalai and Reyzin [KR19] for an overview of these results.

Bounded leakage. From an abstract viewpoint, we can think of the leakage on a random variable X (corresponding, say, to the secret key of an encryption scheme) as a correlated random variable $Z = f(X)$ for some leakage function f that can be chosen by the adversary. Depending on the restriction¹ we put on f , we obtain different leakage models. The first such restriction, introduced for the first time by Dziembowski and Pietrzak [DP08], is to simply assume that the length $\ell \in \mathbb{N}$ of the leakage Z is small enough. This yields the so-called *Bounded Leakage Model*. Thanks to its simplicity and versatility, this model has been used to construct many cryptographic primitives that remain secure in the presence of bounded leakage.

Noisy leakage. A considerable limitation of the Bounded Leakage Model is the fact that, in real-world side-channel attacks, the leakage obtained by the attacker is rarely bounded in length. For instance, the power trace on a physical implementation of AES typically consists of several Megabytes of information, which is much larger than the length of the secret key.

This motivates a more general notion of *noisy leakage*, where there is no upper bound on the length of Z but instead we assume the leakage is somewhat noisy, in the sense that it does not reveal too much information about X . It turns out that the level of noisiness of the leakage can be measured in several ways, each yielding a different leakage model. The first such model, proposed for the first time by Naor and Segev [NS09, NS12] in the setting of leakage-resilient public-key encryption, assumes that the uncertainty of X given Z drops at most by some parameter $\ell \in \mathbb{R}_{>0}$. The latter can be formalized by means of conditional² average min-entropy [DORS08], i.e. by requiring that $\tilde{\mathbb{H}}_\infty(X|Z) \geq \mathbb{H}_\infty(X) - \ell$. In this work, we will refer to this model as the *Min-Entropy-Noisy (ME-Noisy) Leakage Model*. Dodis, Haralambiev, López-Alt, and Wichs [DHLW10] considered a similar model, which we refer to as the *Uniform-Noisy (U-Noisy) Leakage Model*, where the condition about the min-entropy drop is defined w.r.t. the uniform distribution U (rather than on X which may not³ be uniform).

Another variant of noisy leakage was pioneered by Prouff and Rivain [PR13] (building on previous work by Chari, Jutla, Rao, and Rohatgi [CJRR99]), who suggested to measure the noisiness of the leakage by bounding the Euclidean norm between the joint distribution P_{XZ} and the product distribution $P_X \otimes P_Z$ with some parameter $\eta \in (0, 1)$. Follow-up

¹Clearly, there must be some restriction as otherwise $f(X) = X$ and there is no hope for security.

²Intuitively, the conditional average min-entropy of a random variable X given Z measures how hard it is to predict X given Z on average (by an unbounded predictor).

³For instance, in the setting of public-key encryption [NS09], the random variable X corresponds to the distribution of the secret key SK given the public key PK , which may not be uniform.

works by Duc, Dziembowski, and Faust [DDF14, DDF19] and by Prest, Goudarzi, Martinelli, and Passelègue [PGMP19] replaced the Euclidean norm, respectively, with the statistical distance and the mutual information, yielding what we refer to as the *SD-Noisy Leakage* and the *MI-Noisy Leakage Models*. More precisely,⁴ Duc, Dziembowski, and Faust considered a strict subset of SD-noisy leakage—hereafter dubbed *DDF-noisy leakage*—for the special case where $X = (X_1, \dots, X_n)$, for some fixed parameter $n \in \mathbb{N}$, and the function f has a type $f = (f_1, \dots, f_n)$ such that $\Delta(P_{X_i} \otimes P_{Z_i}, P_{X_i Z_i}) \leq \eta$ for each X_i and $Z_i = f_i(X_i)$. All of these works studied noisy leakage in the setting of leakage-resilient circuit compilers (see §1.4).

The different flavors of noisy leakage discussed above capture either a more general class of leakage functions than bounded leakage (as in the case of ME-noisy and U-noisy leakage), or an orthogonal class of leakage functions (as in the case of SD-noisy and MI-noisy leakage). On the other hand, it is usually easiest (and most common) to prove security of a cryptographic primitive against bounded leakage, whereas extending the analysis to other types of noisy leakage requires non-trivial specialized proofs for each primitive. Motivated by this situation, we consider the following question:

*Can we reduce noisy-leakage resilience
to bounded-leakage resilience in a general way?*

1.2 Our Results

In this work, we answer the above question to the positive in the information-theoretic setting. In a nutshell, we achieve this by proving that a novel and very general leakage model, which we refer to as the *Dense Leakage Model* and that encompasses all the aforementioned noisy-leakage models, can be simulated almost for free (albeit possibly inefficiently) using a single query of bounded leakage. Our result allows us to show in a streamlined way that many cryptographic primitives which have only been proved to be resilient against bounded leakage are also secure against noisy leakage, with only a small loss in parameters. Importantly, the latter does not only hold for cryptographic schemes with information-theoretic security, but also for ones with computational security only. We elaborate on our contributions in more details in the paragraphs below, and refer the reader to §1.3 for a more technical overview.

Simulating dense leakage with bounded leakage. As the starting point for our work, in §4, we introduce a meaningful simulation paradigm between leakage models. Informally, given some random variable X and two families of leakage functions \mathcal{F} and \mathcal{G} on X , we say \mathcal{F} is ε -simulatable from \mathcal{G} if for every $f \in \mathcal{F}$ we can simulate $(X, f(X))$ to within statistical distance ε using a *single* query of the form $g(X)$ for some $g \in \mathcal{G}$.

Taking into account the above simulation paradigm, the question we tackle is whether we can have simulation theorems stating that different noisy-leakage families \mathcal{F} are ε -simulatable from the the family \mathcal{G} of ℓ -bounded leakage (for some small ε). We prove such a simulation theorem for a new leakage model that we call *dense leakage*.

In order to define the Dense Leakage Model, we begin with the concept of δ -density: Given two distributions P and P' over a discrete set \mathcal{X} , we say P is δ -dense in P' if $P(x) \leq \frac{P'(x)}{\delta}$ for all $x \in \mathcal{X}$. In particular, δ -density implies that $P(x) = 0$ whenever $P'(x) = 0$, and thus this concept is connected to the notion of absolute continuity of one measure with respect to another. Given this notion, it is simple to describe the Dense Leakage Model. If $Z = f(X)$ denotes some leakage from X , then Z is (p, γ, δ) -dense leakage from X if, with probability $1 - p$

⁴The work by Prest, Goudarzi, Martinelli, and Passelègue considered a similar restriction for MI-noisy leakage.

over the choice of $X = x$, we have

$$P_{Z|X=x}(z) \leq \frac{P_Z(z)}{\delta} \quad (1)$$

with probability $1 - \gamma$ over the choice of $Z = z$. Intuitively, Z being a dense leakage of X essentially corresponds to the distributions $P_{Z|X=x}$ being “approximately” dense in the marginal distribution P_Z for most choices of $x \in \mathcal{X}$.

Our first result is a simulation theorem for dense leakage with respect to bounded leakage, which we state in simplified form below.

Theorem 1 (Informal). *For any random variable X , and every parameter $\varepsilon \in (0, 1)$, the family of (p, γ, δ) -dense leakage functions on X is $(\varepsilon + \varepsilon^{1/4\delta} + \gamma + p)$ -simulatable from the family of ℓ -bounded leakage functions on X , so long as*

$$\ell \geq \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2.$$

On the power of dense leakage. Second, we show that dense leakage captures all of the noisy-leakage models considered above. In particular, we obtain the following informal result.

Theorem 2 (Informal). *The families of ME-noisy, U-noisy, and DDF-noisy leakages fall within the family of dense leakage with good⁵ parameters.*

By combining Theorem 1 and Theorem 2, we obtain non-trivial simulation theorems for the families of ME-noisy, U-noisy, and DDF-noisy leakage from bounded leakage, with small simulation error and small bounded leakage parameter. It is worth mentioning that, for the specific case of ME-noisy leakage, Theorem 2 only holds for distributions X that are almost flat. As we shall prove, this restriction is nearly optimal in the sense that there exist “non-flat” distributions X for which we cannot simulate ME-noisy leakage on X from bounded leakage on X with good parameters, *even when the drop in min-entropy is minimal*.

Turning to the families of SD-noisy and MI-noisy leakage, one can show that they fall within the family of dense leakage too. However, the parameters we obtain in this case are not good enough to be combined with Theorem 1 in order to yield interesting applications. In fact, we prove that the families of η -SD-noisy and η -MI-noisy leakage are trivially simulatable with statistical error roughly η even from the degenerate family of 0-bounded leakage. Unfortunately, this is inherent for the general form of SD-noisy and MI-noisy leakage we consider, in the sense that no simulator can achieve simulation error significantly smaller than η even when leaking almost all of the input. In contrast, Duc, Dziembowski, and Faust [DDF14, DDF19] gave a non-trivial⁶ simulation theorem for the family of DDF-noisy leakage (which is a strict subset of SD-noisy leakage) from a special type of bounded leakage called *threshold probing leakage*. Consistently, Theorem 2 establishes that DDF-noisy leakage is dense leakage with good parameters which in combination with Theorem 1 gives an alternative (non-trivial) simulation theorem for DDF-noisy leakage from bounded leakage. While this result is not new, we believe it showcases the generality of our techniques.

A complete picture, and near-optimality of our simulation theorems. We also provide a complete picture of inclusions and separations between the different leakage models, as depicted in Figure 1. Some of these relationships were already known (e.g., the fact that the

⁵In particular, small enough in order to be combined with Theorem 1 yielding interesting applications.

⁶In particular, with negligible simulation error and small bounded leakage parameter even for constant η .

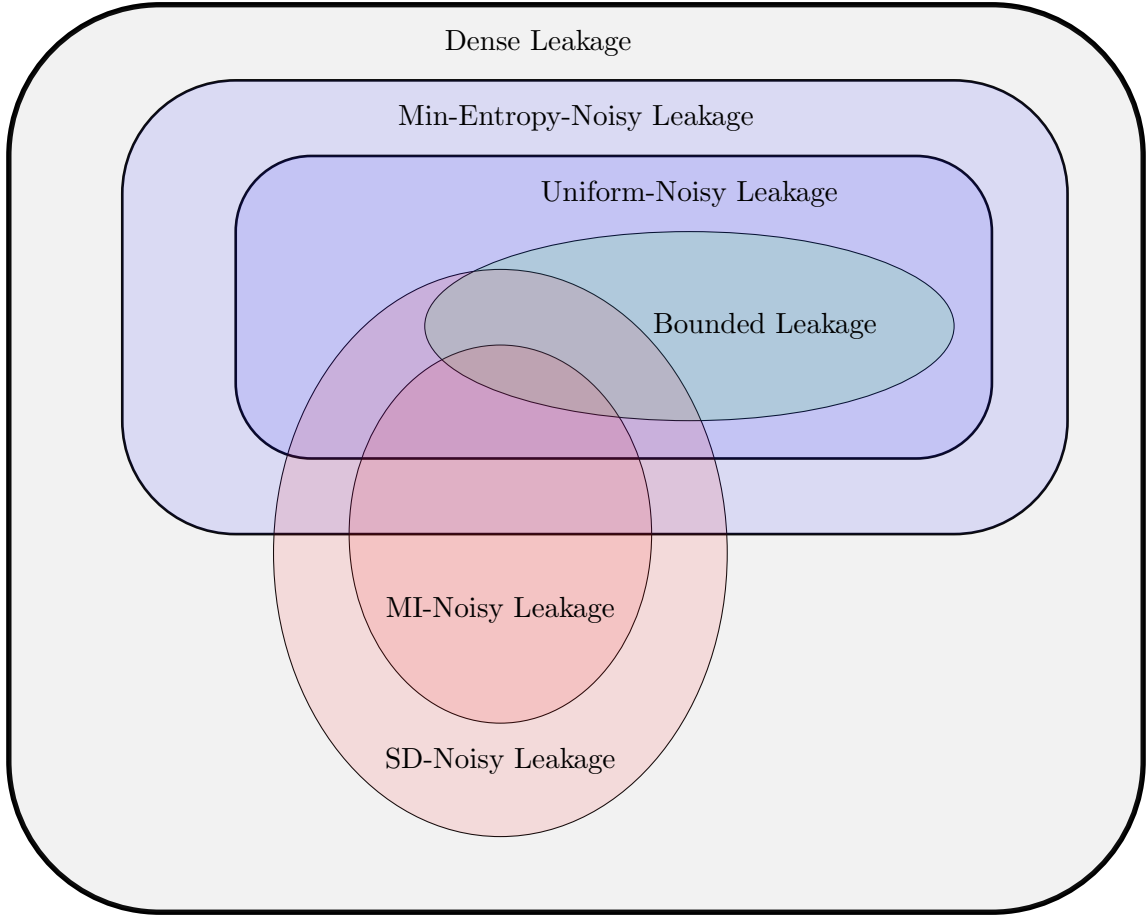


Figure 1: Containment of the different leakage models considered in this paper. Our main result is that a single query of bounded leakage is enough to simulate dense leakage to within small statistical distance.

family of U-noisy leakage is a strict subset of the family of ME-noisy leakage), and some are new (e.g., the separations between the family of SD-noisy leakage and the families of ME-noisy and MI-noisy leakage).

Moreover, we prove a series of results showing that the amount of bounded leakage we use in our simulation theorems is nearly optimal with respect to the desired simulation error.

Applications in brief. Next, we explore applications of our results to leakage-resilient cryptography. Intuitively, the reason why the simulation paradigm is useful is that it may allow us to reduce leakage resilience of a cryptographic scheme against \mathcal{F} to leakage resilience against \mathcal{G} . In particular, when \mathcal{G} is taken to be the family of bounded-leakage functions, we obtain that many primitives which were already known to be secure against bounded leakage are also secure against dense (and thus noisy) leakage. Examples include forward-secure storage [Dzi06a], leakage-resilient one-way functions and public-key encryption [ADVW13], cylinder-intersection extractors [KMS19], symmetric non-interactive key exchange [LMQW20], leakage-resilient secret sharing [BDIR18, SV19, ADN⁺19, KMS19, LCG⁺20] and two-party computation [GIM⁺16].

1.3 Technical Overview

Simulation via rejection sampling. We begin by giving an overview of the approach we use to simulate dense leakage from bounded leakage. As discussed before, our goal is to show that, for a random variable X and some associated dense leakage function f (where f may be randomized), there is a (possibly inefficient) simulator that makes *at most one* black-box query $g(X)$ for some ℓ -bounded leakage function $g : \mathcal{X} \rightarrow \{0, 1\}^\ell$ and outputs \tilde{Z} such that

$$(X, f(X)) \approx_\varepsilon (X, \tilde{Z}), \quad (2)$$

where \approx_ε denotes statistical distance at most ε . For simplicity, we focus here on the setting where f is “exactly” δ -dense leakage from X , meaning that, if $Z = f(X)$, we have

$$P_{Z|X=x}(z) \leq \frac{P_Z(z)}{\delta} \quad (3)$$

for all x and z . This setting is already appropriate to showcase our main ideas.

The key observation that enables the design of our simulator, as we formalize in §3, is that if a distribution P is δ -dense in P' , then it is possible to sample \tilde{P} satisfying $\tilde{P} \approx_\varepsilon P$ with access only to $s = \frac{\log(1/\varepsilon)}{\delta}$ i.i.d. samples from P' , say z_1, z_2, \dots, z_s , and knowledge of the distribution P , via *rejection sampling*: For $i = 1, 2, \dots, s$, either output z_i with probability $\delta P(z_i)/P'(z_i) \leq 1$, or move to $i + 1$ otherwise (if $i = s + 1$, abort).

This suggests the following simulator for f exploiting Eq. (3): The simulator generates s i.i.d. samples $\mathbf{z} = (z_1, z_2, \dots, z_s)$ from P_Z . Then, it queries the bounded-leakage oracle with the randomized function $g_{\mathbf{z}}$ which, with full knowledge of x , performs rejection sampling of $P_{Z|X=x}$ from P_Z using \mathbf{z} . If rejection sampling outputs z_i , then $g_{\mathbf{z}}(x) = i$, and if rejection sampling aborts we may set $g_{\mathbf{z}}(x) = \perp$. In particular, $g_{\mathbf{z}}$ has $1 + s$ possible outputs, and so it is ℓ -bounded-leakage from X with $\ell = \log(1 + s) \leq \log(1/\delta) + \log \log(1/\varepsilon) + 1$. The behavior of the simulator is now clear: Since it knows \mathbf{z} , it can simply output $\tilde{Z} = z_i$ (or $\tilde{Z} = \perp$ if rejection sampling aborted). The discussion above guarantees that the output of the simulator is ε -close in statistical distance to $f(x)$, which yields Eq. (2).

As previously discussed, in the actual proof (which appears in §5.1) we must deal with an approximate variant of Eq. (3). However, we show that the above approach still works in the setting of approximate density at the price of some additional small terms in the simulation error and in the bounded leakage length.

Noisy leakage is dense leakage. As an example of how we manage to frame many types of noisy leakage as dense leakage with good parameters, we discuss how this can be accomplished for ME-noisy leakage assuming X satisfies a property we call α -semi-flatness. The full proof appears in §5.2. The property states that X satisfies $P_X(x) \leq 2^\alpha \cdot P_X(x')$ for all $x, x' \in \text{supp}(X)$, and, as we shall see, it is usually satisfied in applications with small α (or even $\alpha = 0$, which corresponds to a flat distribution). We stress that for the case of U-noisy, DDF-noisy, SD-noisy, and MI-noisy leakages, *no* assumption is required on X to place these types of leakage inside the set of dense leakages. More details can be found in §5.3 and §5.4.

Consider some α -semi-flat X and leakage function f such that $Z = f(X)$ satisfies

$$\mathbb{H}_\infty(X|Z = z) \geq \mathbb{H}_\infty(X) - \ell \quad (4)$$

for some $\ell > 0$ and all z . Note that this is a special case of ME-noisy leakage, but it suffices to present the main ideas of our approach. Our goal is to show that f is $(0, 0, \delta)$ -dense leakage of

X for an appropriate parameter δ , meaning that we wish to prove that $P_{Z|X=x}(z) \leq \frac{P_Z(z)}{\delta}$ for all x and z (recall Eq. (1)). Observe that, by Eq. (4), we have

$$P_{X|Z=z}(x) \leq 2^\ell \max_{x'} P_X(x') \leq 2^{\ell+\alpha} P_X(x)$$

for all x and z , where the rightmost inequality makes use of the fact that X is α -semi-flat. Rewriting the inequality above with the help of Bayes' theorem yields

$$P_{Z|X=x}(z) \leq 2^{\ell+\alpha} P_Z(z),$$

meaning that f is $(p = 0, \gamma = 0, \delta = 2^{-\ell-\alpha})$ -dense leakage of X . By Theorem 1, we then have that $f(X)$ can be simulated with statistical error 2ε using $\ell' = \ell + \alpha + \log \log(1/\varepsilon) + 2$ bits of bounded leakage from X . This statement allows for significant flexibility in the choice of parameters. For example, setting $\varepsilon = 2^{-\lambda}$ for some security parameter λ yields negligible simulation error from $\ell + \alpha + \log(\lambda) + 2$ bits of bounded leakage. Since α is usually very small in applications (often we have $\alpha = 0$), in practice we can achieve negligible simulation error using $\ell + \log(\lambda) + O(1)$ bits of bounded leakage, i.e., by paying only an extra $\log(\lambda) + O(1)$ bits of leakage. Extending the argument above to general ME-noisy leakage from X requires the addition of small error terms p and γ , but setting parameters similarly to the above still allows us to simulate general ℓ -ME-noisy leakage from X using only, say, $\ell + O(\log^2(\lambda))$ bits of bounded leakage from X .

Trivial simulation of SD-noisy and MI-noisy leakages. Consider the trivial simulator that given the function f simply samples \tilde{X} according to the distribution of X and then outputs $\tilde{Z} = f(\tilde{X})$. Assuming f belongs to the family of η -SD-noisy leakage, the above gives a simulation theorem for SD-noisy leakage with simulation error η (and without requiring any leakage from X). By Pinsker inequality, the above also implies a simulation theorem for η -MI-noisy leakage with simulation error $\sqrt{2\eta}$ (again without leaking anything from X).

Unfortunately, it turns out that one cannot do much better than the trivial simulator (even when using large bounded leakage) for our general definition of SD-noisy leakage. More specifically, there exists some X such that any simulator for a function f that is η -SD-noisy leakage from X must incur a simulation error of at least $\eta/2$ even when leaking all but one bit from X . In the case of MI-noisy leakage, we prove a similar result: There exists an X such that any simulator must have simulation error at least $\frac{\eta}{2n}$ when simulating η -MI-noisy leakage from X , even when leaking all but one bit of X . Notably, this means that negligible simulation error is impossible to achieve when η is non-negligible, and thus one cannot do significantly better than the trivial simulator for MI-noisy leakage either.

It is instructive to compare the above trivial simulation theorem for SD-noisy leakage with the result by Duc, Dziembowski, and Faust [DDF14, DDF19], who gave a non-trivial simulation theorem for DDF-noisy leakage from a special case of bounded leakage known as threshold probing leakage. Notice that by the triangle inequality, the trivial simulation theorem for η -SD-noisy leakage implies a trivial simulation theorem for η -DDF-noisy leakage with large simulation error $n \cdot \eta$, which in particular becomes uninteresting as soon as η is non-negligible.

Nevertheless, in §5.5, we show that the family of η -DDF-noisy leakage falls within the family of U-noisy (and thus dense) leakage with good parameters, which in turn gives a non-trivial simulation theorem for η -DDF-noisy leakage from ℓ -bounded leakage with negligible simulation error and for small bounded leakage parameter ℓ , even when $\eta \in (0, 1)$ is constant.

Separations between leakage families, and tradeoffs between simulation error and bounded leakage parameter. We complement our positive results in several ways. First,

we present missing separations between the different types of leakages we consider, leading to a complete picture of their relationships (as depicted in Figure 1). Second, we study the minimum amount of bounded leakage required to simulate different types of noisy leakage with a given simulation error, and show that our simulation theorems are close to optimal. For example, in the case of ME-noisy leakage, for a large range of ℓ and α we show that $\ell + \alpha - O(1)$ bits of bounded leakage are required to simulate ℓ -ME-noisy leakage from some α semi-flat X . In contrast, as discussed above, our simulation theorem states that approximately $\ell + \alpha$ bits of bounded leakage are sufficient to achieve negligible simulation error.

To showcase our approach towards obtaining tradeoffs between simulation error and the bounded leakage parameter, we discuss here one particularly insightful implication of a more general theorem we obtain, which states that enforcing α -semi-flatness of X is necessary to obtain a non-trivial simulation theorem for ME-noisy leakage with sub-constant simulation error. More precisely, there exists X with support in $\{0, 1\}^n$ with an associated 0-noisy leakage function f (meaning that $\tilde{\mathbb{H}}_\infty(X|f(X)) = \mathbb{H}_\infty(X)$) with the property that simulating $Z = f(X)$ with simulation error less than $1/4$ requires one ℓ' -bounded-leakage query for $\ell' \geq n - 2$. In other words, to achieve small simulation error without semi-flatness, we must leak almost all of the input X . The statement above is proved as follows. Consider $X \in \{0, 1\}^n$ satisfying

$$P_X(x) = \begin{cases} 1/2, & \text{if } x = 0^n, \\ \frac{1}{2(2^n - 1)}, & \text{otherwise.} \end{cases}$$

Moreover, set $Z = f(X)$ for a leakage function f such that $f(0^n)$ is uniformly distributed over $\{0, 1\}^n \setminus \{0^n\}$ and $f(x) = x$ with probability 1 for $x \neq 0^n$. Routine calculations show that $\mathbb{H}_\infty(X) = 1$ and $\mathbb{H}_\infty(X|Z = z) = 1$ for all z , meaning that $\tilde{\mathbb{H}}_\infty(X|Z = z) = 1 = \mathbb{H}_\infty(X)$, as desired. Finally, every simulator for (X, Z) above with access to one query of ℓ' -bounded-leakage for $\ell' \leq n - 2$ must have simulation error $1/4$ because, conditioned on $X \neq 0^n$ (which holds with probability $1/2$), we have $f(X) = X$ and X uniform over $\{0, 1\}^n \setminus \{0^n\}$. Therefore, under this conditioning, we can only correctly guess $f(X)$ with probability at most $1/2$ from any one $(n - 2)$ -bounded-leakage query of X .

Sample Application: leakage-resilient secret sharing. We now explain how to use our result in order to lift bounded-leakage resilience to noisy-leakage resilience (almost) for free in cryptographic applications. In fact, in the information-theoretic setting, the latter is an almost immediate consequence of our result.

For the purpose of this overview, let us focus on the concrete setting of secret sharing schemes with local leakage resilience [BDIR18]. Briefly, a t -out-of- n secret sharing scheme allows to share a message y into n shares (x_1, \dots, x_n) in such a way that y can be efficiently recovered using any subset of t shares. Local leakage resilience intuitively says that no unbounded attacker obtaining in full all of the shares $x_{\mathcal{U}}$ within an unauthorized subset $\mathcal{U} \subset [n]$ of size $u < t$, and further leaking at most ℓ bits of information z_i from each of the shares x_i independently, should be able to tell apart a secret sharing of message y_0 from a secret sharing of message y_1 . Benhamouda, Degwekar, Ishai and Rabin [BDIR18] recently proved that both Shamir secret sharing and additive secret sharing satisfy local leakage resilience for certain ranges of parameters.

Thanks to Theorem 1, in §6.1, we show that any secret sharing scheme meeting the above property continues to be secure even if the attacker obtains dense (rather than bounded) leakage on each of the shares x_i independently. The proof of this fact is simple. We move to a mental experiment in which leakages (z_1, \dots, z_n) corresponding to dense-leakage functions (f_1, \dots, f_n) are replaced by $(\tilde{z}_1, \dots, \tilde{z}_n)$ obtained as follows: For each $i \in [n]$, first run the simulator guaranteed by Theorem 1 in order to obtain an ℓ' -bounded leakage function f'_i and compute $z'_i = f'_i(x_i)$; then, run the simulator upon input z'_i in order to obtain a simulated leakage \tilde{z}_i .

By a hybrid argument, the above experiment is statistically close to the original experiment. Furthermore, we can reduce a successful attacker in the mental experiment to an attacker breaking local bounded-leakage resilience. The proofs follows. Finally, thanks to Theorem 2, we can use the abstraction of dense leakage in order to obtain security also in the presence of ME-noisy and U-noisy leakage as well. Note that in the case of ME-noisy leakage, for the second step to work, we need that the distribution X_i of each share outside \mathcal{U} given the shares $x_{\mathcal{U}}$ is almost flat, which is the case for Shamir and additive secret sharing.

Applications in the computational setting. The above proof technique can be essentially applied to any cryptographic primitive with bounded leakage resilience in the information-theoretic setting. Further examples include, e.g., forward-secure storage [Dzi06a], leakage-resilient storage [DDV10], leakage-resilient non-malleable codes [ADKO15], non-malleable secret sharing [KMS19, BFO⁺20] and algebraic manipulation detection codes [AS14, LSW16, AKO18]. (We work out the details for some of these primitives in §A of the appendix.) However, we cannot apply the same trick in the computational setting or when in the proof of security we need to define an efficient simulator (e.g., for leakage-resilient non-interactive zero knowledge [AJS17] and leakage-resilient multi party computation [BDIR18, GIM⁺16]), as the simulation of dense leakage with bounded leakage guaranteed by Theorem 1 may not be efficient.

Nevertheless, we show that our results are still useful for lifting bounded-leakage to noisy-leakage resilience in the computational setting too. In particular, in §6.2, we exemplify how to do that for the concrete construction of leakage-resilient one-way functions in the floppy model proposed by Agrawal, Dodis, Vaikuntanathan and Wichs [ADVW13], and in §A.4 we show how the latter can be achieved in the setting of multi-party computation (MPC).

Let us start with an overview of the former application. Let \mathbb{G} be a cyclic group with generator g and prime order q , and define $g_i = g^{\tau_i}$ for each $i \in [n]$. Upon input a vector $\mathbf{x} = (x_1, \dots, x_n)$, the one-way function outputs $y = \prod_{i=1}^n g_i^{x_i}$; moreover, there is a refreshing procedure that given y and $\boldsymbol{\tau} = (\tau_1, \dots, \tau_n)$ can generate a fresh pre-image \mathbf{x}' of y by simply letting $\mathbf{x}' = \mathbf{x} + \boldsymbol{\sigma}$ for randomly chosen $\boldsymbol{\sigma}$ orthogonal to $\boldsymbol{\tau}$. Here, one should think of $\boldsymbol{\tau}$ as a sort of master secret key to be stored in some secure hardware (i.e., the floppy). Agrawal, Dodis, Vaikuntanathan, and Wichs proved that, under the discrete logarithm assumption in \mathbb{G} , no efficient attacker can successfully invert y even when given ℓ -bounded leakage on \mathbf{x} , so long as $\ell \approx (n - 3) \log(q)$ and assuming that after each leakage query the value \mathbf{x} is refreshed using the floppy. The proof of this fact follows in two steps. First, we move to a mental experiment where each of the leakage queries is answered using a random $(n - 2)$ -dimensional subspace $\mathcal{S} \subseteq \ker(\boldsymbol{\tau})$. By the subspace hiding lemma [BKKV10], this experiment is *statistically close* to the original experiment. Thus, we can use Theorem 1 and Theorem 2 to show that the above still holds in the case of ME-noisy and U-noisy leakage.⁷ Second, one finally reduces a successful attacker in the mental experiment to an efficient breaker for the discrete logarithm problem; in this last step, however, the reduction can trivially answer leakage queries by using \mathcal{S} , and thus it does not matter whether the leakage is bounded or noisy. We believe the above blueprint can be applied to analyze other cryptographic primitives whose leakage resilience is derived through the subspace hiding lemma; we mention a few natural candidates at the end of §6.2.

Turning to simulation-based security, additional work is required before we can apply our main theorem. To keep the exposition simple, let us focus here on the concrete setting of zero-knowledge (ZK) proofs (although in §A.4 we deal with MPC for arbitrary functionalities). The idea is to downgrade the notion of security from statistical ZK in the presence of bounded leakage to the weaker notion of statistical witness indistinguishability (WI) in the presence of

⁷The former requires the distribution of \mathbf{x} given y and $(\mathbb{G}, g, g_1, \dots, g_n, q)$ to be almost flat which is easily seen to be the case.

bounded leakage. In the latter setting, we can use our main theorem to show that any protocol that is statistically WI in the presence of bounded leakage is also statistically WI in the presence of dense leakage. Hence, we can achieve computational ZK in the presence of dense leakage by applying the standard transformation from WI to ZK in the common reference string model (see, e.g., [GOS06]). Finally, we give a concrete instantiation based on the DDH assumption by invoking a result from Goyal, Ishai, Maji, Sahai and Sherstov [GIM⁺16] for passively secure two-party computation in the presence of bounded leakage.

Bounded-collusion protocols. Finally, motivated by additional applications to leakage-resilient cryptography and by exploring new lower bounds in communication complexity [Yao79], in §6.3, we investigate the setting of bounded-collusion protocols (BCPs) as proposed by Kumar, Meka, and Sahai [KMS19]. Here, a set of n parties each holding an input x_i wishes to evaluate a Boolean function ϕ of their inputs by means of an interactive protocol π . At the j -th round, a subset of k parties (where $k < n$ is called the collusion bound) is selected, and appends to the protocol transcript τ an arbitrary (possibly unbounded) function f_j of their joint inputs. The goal is to minimize the size ℓ of the transcript, which leads to what we call an ℓ -bounded communication k -bounded collusion protocol (BC-BCP). BC-BCPs interpolate nicely between the well-studied number-in-hand (NIH) [PVZ12] (which corresponds to $k = 1$) and number-on-forehead (NOF) [CFL83] (which corresponds to $k = n - 1$) models.

We put forward two natural generalizations of BC-BCPs, dubbed *dense* (resp. *noisy*) communication k -bounded collusion protocols (DC-BCPs, resp. NC-BCP), in which there is no restriction on the length of the final transcript τ but the round functions are either dense or U-noisy leakage functions. It is easy to see that any BC-BCP is also a NC-BCP as well as a DC-BCP. By Theorem 1 and Theorem 2, we are able to show that the converse is also true: namely, we can simulate⁸ the transcript τ of any DC-BCP or NC-BCP π using the transcript τ' of a related BC-BCP π' up to a small statistical distance. Protocol π' roughly runs π and uses the simulation paradigm in order to translate the functions used within π into functions to be used within π' . The proof requires a hybrid argument, and thus the final simulation error grows linearly with the number of rounds of the underlying BC-BCP.

The above fact has two consequences. The first consequence is that we can translate communication complexity lower bounds for BC-BCPs into lower bounds on the noisiness of NC-BCPs. A communication complexity lower bound for a Boolean function ϕ says that any BC-BCP computing ϕ with good probability must have long transcripts (i.e., large ℓ). Concrete examples of such functions ϕ include those based on the generalized inner product and on quadratic residues in the NOF model with logarithmic (in the input length) number of parties [Chu90, BNS92], and more recently a new function (based on the Bourgain extractor [Bou05]) for more general values of k and even for super-logarithmic number of parties [KMZ20]. Note that the above lower bounds do not necessarily say how much information a transcript must reveal about the inputs. Thanks to our results, we can show that any NC-BCP (i.e., where there is no upper bound on the transcript length) computing the above functions with good probability must also in some sense reveal enough information about the inputs. However, for technical reasons, the latter holds true only so long as the number of rounds is not too large. We refer the reader to §6.3.1 for further details.

The second consequence is that we can lift the security of cryptographic primitives whose leakage resilience is modeled as a BC-BCP (which intuitively corresponds to security against adaptive bounded joint leakage) to the more general setting where leakage resilience is modeled

⁸The reason for not considering NC-BCPs where the round functions are ME-noisy (instead of U-noisy) leakage functions is that simulating ME-noisy leakage with bounded leakage inherently requires semi-flatness, but we cannot ensure this condition is maintained throughout the entire execution of a leakage protocol.

as a NC-BCP or DC-BCP (which intuitively corresponds to security against adaptive noisy joint leakage). Examples includes secret sharing with security against adaptive joint leakage [KMS19, CGGL20, KMZ20] (see §6.3.2), extractors for cylinder-intersection sources [KMS19, LMQW20, CGGL20, KMZ20] (see §A.2) and leakage-resilient non-interactive key exchange [LMQW20] (see §A.3). Interestingly, the security of these applications in the bounded-leakage setting has been derived exploiting communication complexity lower bounds for BC-BCPs. We can instead directly lift security to the dense and U-noisy leakage setting in a fully black-box way, and thus without re-doing the analysis.

1.4 Related Work

Naor and Segev [NS09, NS12] conjectured that ME-noisy leakage may be compressed to small leakage in the information-theoretic setting. In this light, our results prove this conjecture to be false for arbitrary distributions and establish the exact conditions under which the above statement holds true not only in the case of ME-noisy leakage, but also for U-noisy leakage.

Most relevant to our work is the line of research on leakage-resilient circuit compilers (see, e.g., [ISW03, FRR⁺10, FRR⁺14]), where the equivalence of different leakage models has also been explored. For instance, the beautiful work by Duc, Dziembowski, and Faust [DDF14, DDF19] shows that DDF-noisy leakage on masking schemes used to protect the internal values within a cryptographic circuit can be simulated by probing a limited number of wires (which can be thought of as bounded leakage in the circuit setting). The notion of DDF-noisy leakage was studied further, both experimentally and theoretically, by Duc, Faust, and Standaert [DFS15a, DFS19]. Follow-up work by Dziembowski, Faust, and Skórski [DFS15b] and by Prest, Goudarzi, Martinelli, and Passelègue [PGMP19] further improved the parameters of such a reduction and extended it to other noisy-leakage models as well. The difference between the above results and our work is that we prove simulation theorems between very abstract and general leakage models, which ultimately allows us to obtain a broad range of applications which goes beyond the setting of leakage-resilient circuits. In a complementary direction, Fuller and Hamlin [FH15] studied the relationship between different types of computational leakage.

Harsha, Ishai, Kilian, Nissim, and Venkatesh [HIK⁺04] investigate tradeoffs between communication complexity and time complexity in non-cryptographic settings, including deterministic two-party protocols, query complexity and property testing. Our simulation theorems can be thought of as similar tradeoffs in the cryptographic setting.

2 Preliminaries

2.1 Notation

We denote by $[n]$ the set $\{1, \dots, n\}$. For a string $x \in \{0, 1\}^*$, we denote its length by $|x|$; if \mathcal{X} is a set, $|\mathcal{X}|$ represents the number of elements in \mathcal{X} . When x is chosen randomly in \mathcal{X} , we write $x \leftarrow_{\$} \mathcal{X}$. When A is a randomized algorithm, we write $y \leftarrow_{\$} A(x)$ to denote a run of A on input x (and implicit random coins r) and output y ; the value y is a random variable and $A(x; r)$ denotes a run of A on input x and randomness r . An algorithm A is *probabilistic polynomial-time* (PPT for short) if A is randomized and for any input $x, r \in \{0, 1\}^*$, the computation of $A(x; r)$ terminates in a polynomial number of steps (in the size of the input).

Negligible functions. We denote with $\lambda \in \mathbb{N}$ the security parameter. A function p is *polynomial* (in the security parameter), denoted $p \in \text{poly}(\lambda)$, if $p(\lambda) \in O(\lambda^c)$ for some constant $c > 0$. A function $\nu : \mathbb{N} \rightarrow [0, 1]$ is *negligible* (in the security parameter) if it vanishes faster than

the inverse of any polynomial in λ , i.e. $\nu(\lambda) \in O(1/p(\lambda))$ for all positive polynomials $p(\lambda)$. We sometimes write $\text{negl}(\lambda)$ to denote an unspecified negligible function. Unless stated otherwise, throughout the paper, we implicitly assume that the security parameter is given as input (in unary) to all algorithms.

2.2 Random Variables

For a random variable X , we write $\mathbb{P}[X = x]$ for the probability that X takes on a particular value $x \in \mathcal{X}$, with \mathcal{X} being the set where X is defined. The probability mass function of X is denoted P_X , i.e., $P_X(x) = \mathbb{P}[X = x]$ for all $x \in \mathcal{X}$; we sometimes omit X and just write P when X is clear from the context. For a set (or event) $\mathcal{S} \subseteq \mathcal{X}$, we write $P(\mathcal{S})$ for the probability of event \mathcal{S} , i.e. $P(\mathcal{S}) = \sum_{x \in \mathcal{S}} P(x)$.

The statistical distance between two random variables X and Y over \mathcal{X} is defined as

$$\Delta(X; Y) := \frac{1}{2} \sum_{x \in \mathcal{X}} |\mathbb{P}[X = x] - \mathbb{P}[Y = x]|.$$

Given two ensembles $X = \{X_\lambda\}_{\lambda \in \mathbb{N}}$ and $Y = \{Y_\lambda\}_{\lambda \in \mathbb{N}}$, we write $X \equiv Y$ to denote that they are identically distributed, $X \approx Y$ to denote that they are *statistically close*, i.e. $\Delta(X_\lambda; Y_\lambda) \leq \text{negl}(\lambda)$, and $X \overset{c}{\approx} Y$ to denote that they are *computationally indistinguishable*, i.e. for all PPT distinguishers D :

$$|\mathbb{P}[D(X_\lambda) = 1] - \mathbb{P}[D(Y_\lambda) = 1]| \leq \text{negl}(\lambda).$$

Sometimes we explicitly denote by $X \approx_\varepsilon Y$ the fact that $\Delta(X_\lambda; Y_\lambda) \leq \varepsilon$ for a parameter $\varepsilon = \varepsilon(\lambda)$.

Average min-entropy. The min-entropy of a random variable X with domain \mathcal{X} is defined as $\mathbb{H}_\infty(X) := -\log \max_{x \in \mathcal{X}} \mathbb{P}[X = x]$, which intuitively measures the best chance to predict X (by a computationally unbounded algorithm). For conditional distributions, unpredictability is measured by the conditional average min-entropy [DRS04, DORS08], defined as

$$\tilde{\mathbb{H}}_\infty(X|Y) := -\log \mathbb{E}_{y \sim Y} \left[2^{-\mathbb{H}_\infty(X|Y=y)} \right].$$

The lemmas below are sometimes known as the “chain rule” for conditional average min-entropy.

Lemma 1 ([DORS08]). *For arbitrary random variables X , Y , and Z such that $|\text{supp}(Z)| \leq 2^\ell$ it holds that*

$$\tilde{\mathbb{H}}_\infty(X|Y, Z) \geq \tilde{\mathbb{H}}_\infty(X, Z|Y) - \ell \geq \tilde{\mathbb{H}}_\infty(X|Y) - \ell.$$

Lemma 2 ([MW97]). *For arbitrary random variables X , Y , and Z , and for every $\delta \geq 0$, it holds that*

$$\mathbb{P}_{z \sim Z} \left[\tilde{\mathbb{H}}_\infty(X|Y, Z = z) \geq \tilde{\mathbb{H}}_\infty(X|Y, Z) - \log(1/\delta) \right] \geq 1 - \delta.$$

2.3 Hardness Assumptions

Let $\text{GroupGen}(1^\lambda)$ be a randomized algorithm outputting the description of a cyclic group \mathbb{G} with generator g and prime order $q = q(\lambda)$.

Definition 1 (Discrete logarithm assumption). *We say that the discrete logarithm (DL for short) assumption holds for GroupGen , if for all PPT attackers A there is a negligible function $\nu : \mathbb{N} \rightarrow [0, 1]$ such that*

$$\mathbb{P} \left[x' = x : (\mathbb{G}, g, q) \leftarrow \text{GroupGen}(1^\lambda); x \leftarrow \mathbb{Z}_q; x' \leftarrow A(\mathbb{G}, g, q, g^x) \right] \leq \nu(\lambda).$$

3 Rejection Sampling for Approximate Density

The problem that we consider in this section is the following: How can we sample from a distribution P with statistical error at most ε , given only *black-box* access to i.i.d. samples from another distribution P' ?

It turns out that the problem above can be solved via rejection sampling, assuming that P is approximately dense in P' as defined below.

Definition 2 (δ -density). *Given two distributions P and P' over some set \mathcal{X} and some $\delta \in (0, 1]$, we say P is δ -dense in P' if for every $x \in \mathcal{X}$ it holds that*

$$P(x) \leq \frac{P'(x)}{\delta}.$$

Remark 1 (On the parameter δ). Note that the definition of δ -density only makes sense when $\delta \leq 1$. In fact, if $\delta > 1$ above, then P cannot be a probability distribution.

Definition 3 ((γ, δ) -density). *Given two distributions P and P' over some set \mathcal{X} and some $\gamma, \delta \in [0, 1]$, we say P is γ -approximate δ -dense in P' , or simply (γ, δ) -dense in P' , if there exists a set $\mathcal{S} \subseteq \mathcal{X}$ such that $P(\mathcal{S}), P'(\mathcal{S}) \geq 1 - \gamma$, and for all $x \in \mathcal{S}$ it holds that*

$$P(x) \leq \frac{P'(x)}{\delta}.$$

3.1 The Case of Exact Density

First, we consider the special case where P is δ -dense in P' .

Lemma 3. *Suppose P is δ -dense in P' . Then, for any $\varepsilon > 0$, it is possible to sample \tilde{P} such that $\tilde{P} \approx_\varepsilon P$ given access to*

$$s = \frac{\log(1/\varepsilon)}{\delta}$$

i.i.d. samples from P' .

Proof. Consider the following rejection sampling algorithm:

1. Sample x_1, \dots, x_s i.i.d. according to the distribution P' , and set $y := \perp$;
2. For $i = 1, \dots, s$ do the following: Set $B_i := 1$ with probability $p_i = \frac{\delta P(x_i)}{P'(x_i)}$ and $B_i := 0$ otherwise. If $B_i := 1$, set $y := x_i$ and stop the cycle;
3. Output y .

Observe that $\frac{\delta P(x_i)}{P'(x_i)} \leq 1$ for all x_i (hence the algorithm above is valid), and that the probability that the algorithm outputs some x in the i -th round is

$$\mathbb{P}[B_i = 1] = \sum_x P'(x) \cdot \frac{\delta P(x)}{P'(x)} = \delta. \quad (5)$$

Let \tilde{P} denote the distribution of the output of the algorithm above and let Y be the corresponding output. Observe that $(Y|Y \neq \perp)$ is distributed exactly like P . This holds because, in view of Eq. (5), the probability that the algorithm outputs x in the i -th round given that it stops in the i -th round is

$$\mathbb{P}[Y = x | B_i = 1, \forall j < i : B_j = 0] = \frac{(1 - \delta)^{i-1} \cdot P'(x) \cdot \frac{\delta P(x)}{P'(x)}}{(1 - \delta)^{i-1} \cdot \delta} = P(x).$$

Moreover, we have

$$\mathbb{P}[Y = \perp] = (1 - \delta)^s \leq \exp(-\delta \cdot s) = \varepsilon.$$

From these observations, we conclude that $\tilde{P} \approx_\varepsilon P$. \square

3.2 The Case of Approximate Density

We are now ready to prove an analogous result for approximate density.

Lemma 4. *Suppose P is (γ, δ) -dense in P' . Then, for any $\varepsilon > 0$, it is possible to sample \tilde{P} such that $\tilde{P} \approx_{\varepsilon + \frac{1}{4\delta} + \gamma} P$ given access to*

$$\frac{2 \log(1/\varepsilon)}{\delta(1 - \gamma)^2}$$

i.i.d. samples from P' .

Proof. Let \mathcal{S} denote the set such that $P(\mathcal{S}), P'(\mathcal{S}) \geq 1 - \gamma$, and $P(x) \leq \frac{P'(x)}{\delta}$ for all $x \in \mathcal{S}$. Then, if $P_{\mathcal{S}}$ and $P'_{\mathcal{S}}$ denote distributions over \mathcal{S} satisfying

$$P_{\mathcal{S}}(x) = \frac{P(x)}{P(\mathcal{S})} \quad \text{and} \quad P'_{\mathcal{S}}(x) = \frac{P'(x)}{P'(\mathcal{S})}$$

for $x \in \mathcal{S}$, we conclude that

$$P_{\mathcal{S}}(x) = \frac{P(x)}{P(\mathcal{S})} \leq \frac{P'(x)}{\delta P(\mathcal{S})} \leq \frac{P'_{\mathcal{S}}(x)}{\delta(1 - \gamma)}$$

for all $x \in \mathcal{S}$, where we have used the facts that $P(\mathcal{S}) \geq 1 - \gamma$. Therefore, it follows that $P_{\mathcal{S}}$ is $(\delta(1 - \gamma))$ -dense in $P'_{\mathcal{S}}$. By Lemma 3, we can sample $\tilde{P}_{\mathcal{S}}$ such that $\tilde{P}_{\mathcal{S}} \approx_\varepsilon P_{\mathcal{S}}$ from $s = \frac{\log(1/\varepsilon)}{\delta(1 - \gamma)}$ i.i.d. samples from $P'_{\mathcal{S}}$. Since $P_{\mathcal{S}} \approx_\gamma P$, by the triangle inequality we have that $\tilde{P}_{\mathcal{S}} \approx_{\varepsilon + \gamma} P$.

To complete the proof, it suffices to observe that, if we have access to $\frac{2s}{1 - \gamma}$ i.i.d. samples from P' and W denotes the number of such samples that are in \mathcal{S} , a straightforward application of the Chernoff bound⁹ shows that

$$\begin{aligned} \mathbb{P}[W < s] &\leq \mathbb{P}\left[W < \frac{\mathbb{E}[W]}{2}\right] \\ &\leq \exp(-s/4) \\ &\leq \varepsilon^{\frac{1}{4\delta}}. \end{aligned}$$

Therefore, we can add $\varepsilon^{\frac{1}{4\delta}}$ to the statistical error and assume that we have access to at least s i.i.d. samples from $P'_{\mathcal{S}}$. \square

4 Leakage Models

In this section, we review several leakage models from the literature, and introduce the simulation paradigm which will later allow us to draw connections between different leakage models.

Our take is very general, in that we think as the leakage as a randomized function f on a random variable X , over a set \mathcal{X} , which yields a correlated random variable $Z = f(X)$. Different leakage models are then obtained by putting restrictions on the joint distribution (X, Z) . We refer the reader to §6 for concrete examples of what the distribution X is in applications.

⁹The version of the Chernoff bound that we use here states that $\mathbb{P}[W < (1 - c)\mathbb{E}[W]] \leq \exp(-c^2\mathbb{E}[W]/2)$ for $0 \leq c \leq 1$, provided W is a sum of independent random variables in $\{0, 1\}$.

4.1 Bounded Leakage

A first natural restriction is to simply assume an upper bound $\ell \in \mathbb{N}$ on the total length of the leakage. This yields the so-called Bounded Leakage Model, which was formalized for the first time by Dziembowski and Pietrzak [DP08].

Definition 4 (Bounded leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is an ℓ -bounded leakage function for X if $\mathcal{Z} \subseteq \{0, 1\}^\ell$. For fixed X , we denote the set of all its ℓ -bounded leakage functions by $\text{Bounded}_\ell(X)$.*

4.2 Noisy Leakage

A considerable drawback of the Bounded Leakage Model is that physical leakage is rarely of bounded length. The Noisy Leakage Model overcomes this limitation by assuming that the length of the leakage is unbounded but somewhat *noisy*.

There are different ways from the literature how to measure the noisiness of the leakage. A first way, considered for the first time by Naor and Segev [NS09, NS12], is to assume that the leakage drops the min-entropy of X by at most $\ell \in \mathbb{R}_{>0}$ bits. We will refer to this model as the ME-Noisy Leakage Model.

Definition 5 (ME-noisy leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is an ℓ -ME-noisy leakage function for X if, denoting $Z = f(X)$, we have*

$$\tilde{\mathbb{H}}_\infty(X|Z) \geq \mathbb{H}_\infty(X) - \ell.$$

For fixed X , we denote the set of all its ℓ -ME-noisy leakage functions by $\text{Noisy}_{\infty, \ell}(X)$.

Dodis *et al.* [DHLW10] considered a slight variant of the above definition where the min-entropy drop is measured w.r.t. the uniform distribution U over \mathcal{X} (rather than X itself). We will refer to this model as the U-Noisy Leakage Model.

Definition 6 (U-noisy leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is an ℓ -U-noisy leakage function for X if it holds that*

$$\tilde{\mathbb{H}}_\infty(U|f(U)) \geq \mathbb{H}_\infty(U) - \ell,$$

where U denotes the uniform distribution over \mathcal{X} . For fixed X , we denote the set of all its ℓ -U-noisy leakage functions by $\text{UNoisy}_{\infty, \ell}(X)$.

A second way to measure noisiness is to assume that the leakage only implies a bounded bias in the distribution X , which is formally defined as distributions P_{XZ} and $P_X \otimes P_Z$ being close according to some distance when seen as real-valued vectors. Prouff and Rivain [PR13] were the first to consider this restriction using the Euclidean norm (i.e., the ℓ_2 -norm), whereas Duc, Dziembowski and Faust [DDF14, DDF19] used the statistical distance (i.e., the ℓ_1 -norm). We recall the latter definition below, which we will refer to as the SD-Noisy Leakage Model.

Definition 7 (SD-noisy leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is an η -SD-noisy leakage function for X if, denoting $Z = f(X)$, it holds that*

$$\Delta(P_{XZ}; P_X \otimes P_Z) \leq \eta,$$

where $P_X \otimes P_Z$ denotes the product distribution of X and Z . For fixed X , we denote the set of all its η -SD-noisy leakage functions by $\text{Noisy}_{\Delta, \eta}(X)$.

Duc, Dziembowski, and Faust [DDF14, DDF19] considered only a restricted subset of SD-noisy leakage, which we call *DDF-noisy leakage*. We define this notion below.

Definition 8 (DDF-noisy leakage). *Given a random variable $X = (X_1, X_2, \dots, X_n)$ over \mathcal{X}^n , we say a randomized function $f : \mathcal{X}^n \rightarrow \mathcal{Z}^n$ is an η -DDF-noisy leakage function for X if $f = (f_1, f_2, \dots, f_n)$ for functions $f_i : \mathcal{X} \rightarrow \mathcal{Z}$ such that $f_i \in \text{Noisy}_{\Delta, \eta}(X_i)$ for $i = 1, 2, \dots, n$. For a given X , we define the set of all its η -DDF-noisy leakage functions by $\text{DDFNoisy}_{\Delta, \eta}(X)$.*

By the triangle inequality, it is easy to see that

$$\text{DDFNoisy}_{\Delta, \eta}(X) \subseteq \text{Noisy}_{\Delta, \eta \cdot n}(X)$$

for every η and $X \in \mathcal{X}^n$. However, as we shall see, DDF-noisy leakage behaves quite differently than SD-noisy leakage in general. In fact, we show in §5.4 that essentially no non-trivial simulation theorem exists for SD-noisy leakage. In contrast, as it was shown in [DDF14, DDF19], it is possible to simulate DDF-noisy leakage with very low error given access to one query of a special type of bounded leakage called *threshold probing leakage*. Another reason for why this holds, which we will discuss in §5.5 in more detail, is that DDF-noisy leakage is a subset of U-noisy leakage (the same is not true of SD-noisy leakage). Since, as we shall show, U-noisy leakage can be simulated with very good parameters from bounded leakage via our general framework, the same automatically holds for DDF-noisy leakage.

Alternatively, as suggested by Prest *et al.* [PGMP19], we can measure the noisiness of the leakage by looking at the mutual information between X and Z . We can define the mutual information between X and Z as $I(X; Z) = D_{\text{KL}}(P_{XZ} \| P_X \otimes P_Z)$, where $D_{\text{KL}}(P \| P') = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{P'(x)}\right)$ is the Kullback-Leibler divergence between P and P' .

Definition 9 (MI-noisy leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is an η -MI-noisy leakage function for X if, denoting $Z = f(X)$, it holds that*

$$I(X; Z) \leq \eta.$$

For fixed X , we denote the set of all its η -MI-noisy leakage functions by $\text{MINoisy}_{\eta}(X)$.

The well-known Pinsker inequality allows us to relate MI-noisy leakage to SD-noisy leakage.

Lemma 5 (Pinsker inequality). *For arbitrary distributions P and P' over a set \mathcal{X} it holds that*

$$\Delta(P; P') \leq \sqrt{2 \cdot D_{\text{KL}}(P \| P')}.$$

As an immediate corollary of Lemma 5, we obtain the following result (which was observed also in [PGMP19]).

Corollary 1. *For any $\eta > 0$ and X we have*

$$\text{MINoisy}_{\eta}(X) \subseteq \text{Noisy}_{\Delta, \sqrt{2\eta}}(X).$$

Proof. Fix $Z = f(X)$ for $f \in \text{MINoisy}_{\eta}(X)$. The desired statement follows by noting that $I(X; Z) = D_{\text{KL}}(P_{XZ} \| P_X \otimes P_Z)$. Therefore, the Pinsker inequality implies that

$$\Delta(P_{XZ}; P_X \otimes P_Z) \leq \sqrt{2 \cdot I(X; Z)} \leq \sqrt{2\eta}.$$

□

4.3 Dense Leakage

Next, we introduce a new leakage model which we dub the Dense Leakage Model. This model intuitively says that the distribution of $Z|X = x$ is approximately dense in the distribution of Z for a large fraction of x 's. Looking ahead, dense leakage will serve as a powerful abstraction to relate different leakage models.

Definition 10 (Dense leakage). *Given a random variable X over \mathcal{X} , we say a randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$ is a (p, γ, δ) -dense leakage function for X if, denoting $Z = f(X)$, there exists a set $\mathcal{T} \subseteq \mathcal{X}$ with $P_X(\mathcal{T}) \geq 1 - p$ such that $P_{Z|X=x}$ is (γ, δ) -dense in P_Z for all $x \in \mathcal{T}$. For fixed X , we denote the set of all its (p, γ, δ) -dense leakage functions by $\text{Dense}_{p, \gamma, \delta}(X)$.*

We give a simple example of a dense leakage function to help the reader get comfortable with the notion. Let f be the function that upon input a source $X \in \{0, 1\}^n$ outputs the first ℓ bits of X , and assume X is uniform. Then, the function f is $(1, 1, 2^{-\ell})$ -dense leakage for X . Indeed, let Z be the random variable corresponding to the output of f , and, for any $x \in \{0, 1\}^n$, parse $x = z||y$ where $|z| = \ell$. Then, for any z' , the probability $P_{Z|X=x}(z')$ is equal to 1 when $z = z'$ and 0 otherwise. On the other hand, $P_Z(z')$ equals $2^{-\ell}$ which implies that $\delta = 2^{-\ell}$.

4.4 The Simulation Paradigm

Finally, we define the simulation paradigm which allows to draw connections between different leakage models. Intuitively, for any random variable X , we will say that a leakage family $\mathcal{F}(X)$ is simulatable from another leakage family $\mathcal{G}(X)$ if for all functions $f \in \mathcal{F}(X)$ there exists a simulator Sim_f which can generate \tilde{Z} such that (X, Z) and (X, \tilde{Z}) are statistically close, using a single sample $g(X)$ for some function $g \in \mathcal{G}(X)$.

Definition 11 (Leakage simulation). *Given a random variable X and two leakage families $\mathcal{F}(X)$ and $\mathcal{G}(X)$, we say $\mathcal{F}(X)$ is ε -simulatable from $\mathcal{G}(X)$ if for all $f \in \mathcal{F}(X)$ there is a (possibly inefficient) randomized algorithm Sim_f such that the following holds*

$$(X, Z) \approx_\varepsilon (X, \text{Sim}_f^{\text{Leak}(X, \cdot)}),$$

where $Z = f(X)$ and the oracle $\text{Leak}(X, \cdot)$ accepts a single query $g \in \mathcal{G}(X)$ and outputs $g(X)$.

Remark 2 (On the simulator). Note that since the simulator Sim_f knows the distribution P_X of X and the leakage function f , it also knows the joint distribution $P_{X, Z}$ where $Z = f(X)$. We will use this fact to design our leakage simulators. We will also sometimes think of the simulator Sim_f as two machines with a shared random tape, where the first machine outputs the description of a leakage function $g \in \mathcal{G}(X)$, while the second machine outputs the simulated leakage \tilde{Z} given the value $g(X)$.

5 Relating Different Leakage Models

In this section, we show both implications and separations between the leakage models defined in §4. In a nutshell, our implications show that all the noisy-leakage models from §4 can be simulated by bounded leakage with good parameters. We achieve this in two main steps: First, we prove that dense leakage can be simulated by bounded leakage with good parameters. Second, we show that dense leakage contains the other leakage models we have previously defined. Combining the two steps above, we conclude that many different leakage models can be simulated by bounded leakage with good parameters. To complement these results, our

separations show that the containment of the different leakage models in dense leakage are essentially the best we can hope for in general.

The simulation theorem for the case of ME-noisy leakage only holds for certain distributions X , which are nevertheless the most relevant in applications. In particular, we will require to assume that the random variable X is semi-flat, as formally defined below.

Definition 12 (Semi-flat distribution). *We say that X is α -semi-flat if for all $x, x' \in \text{supp}(X)$ we have*

$$P_X(x) \leq 2^\alpha \cdot P_X(x').$$

5.1 Simulating Dense Leakage with Bounded Leakage

The following theorem states that one dense leakage query can be simulated with one bounded leakage query to within small statistical error. The efficiency of the simulator and the bounded leakage function is essentially governed by the density parameter δ .

Theorem 3. *For arbitrary X , and for any $\varepsilon > 0$, the set of dense leakages $\text{Dense}_{p,\gamma,\delta}(X)$ is $(\varepsilon + \varepsilon^{1/4\delta} + \gamma + p)$ -simulatable from $\text{Bounded}_\ell(X)$ with*

$$\ell = 1 + \log\left(\frac{2 \log(1/\varepsilon)}{(1-\gamma)^{2\delta}}\right) = \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2.$$

Proof. Fix any $f \in \text{Dense}_{p,\gamma,\delta}(X)$. By hypothesis, there is a set $\mathcal{T} \subseteq \mathcal{X}$ such that $P_X(\mathcal{T}) \geq 1-p$ and $P_{Z|X=x}$ is (γ, δ) -dense in P_Z for all $x \in \mathcal{T}$. Thus, we may assume that $X \in \mathcal{T}$ by adding p to the simulation error.

We consider the simulator Sim_f which, given the distribution P_{XZ} , samples $s^* = \frac{2 \log(1/\varepsilon)}{(1-\gamma)^\delta}$ i.i.d. samples $\mathbf{z} = (z_1, z_2, \dots, z_{s^*})$ from P_Z . Then, Sim_f makes a query to $Z' = g_{\mathbf{z}}(X) \in \text{Bounded}_\ell(X)$, where $\ell = 1 + \log s^*$ and $g_{\mathbf{z}} : \mathcal{X} \rightarrow \{0, 1\}^\ell$ on input $x \in \mathcal{T}$ runs the rejection sampling algorithm from the proof of Lemma 4 to sample from $P_{Z|X=x}$ to within statistical error $\varepsilon + \varepsilon^{1/4\delta} + \gamma$ using the s^* i.i.d. samples (z_1, \dots, z_{s^*}) from P_Z , and outputs the index $i \leq s^*$ such that z_i is output by the rejection sampling algorithm, or $s^* + 1$ if this algorithm outputs \perp . Finally, if $I = g_{\mathbf{z}}(X) \leq s^*$, then Sim_f outputs z_I , and otherwise it outputs \perp . Let \tilde{Z} the random variable corresponding to the output of the simulator. Summing up the different simulation errors, Lemma 4 guarantees that

$$(X, Z) \approx_{\varepsilon + \varepsilon^{1/4\delta} + \gamma + p} (X, \tilde{Z}),$$

which completes the proof. □

Remark 3 (On useful parameters). The statement of Theorem 3 is most useful when ε , γ , and p are negligible in the security parameter, so as to obtain negligible simulation error. The parameter δ essentially dictates the number of bits of bounded leakage required to simulate a given class of dense leakages. Indeed, it is usually the case that $\log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right)$ is much smaller than $\log(1/\delta)$.

Remark 4 (On efficiency of the simulation). The complexity of the simulator from Theorem 3 is dominated by the complexity of computing the distributions P_Z (possible with knowledge of P_X and f) and $P_{Z|X=x}$ (possible with knowledge of X and f), and of sampling both the z_i according to P_Z and the decision in each step of rejection sampling. If these steps can be implemented in polynomial time with respect to some parameter of interest, then the simulator is efficient.

5.2 Min-Entropy-Noisy Leakage is Dense Leakage

The following theorem states that all ME-noisy leakage is also dense leakage for semi-flat distributions. Looking ahead, we will later establish that the semi-flatness condition is necessary.

Theorem 4. *Suppose X is α -semi-flat. Then, for every $\beta > 0$ and $\ell > 0$, and for $p = 2^{-\beta/2}$, $\gamma = 2^{-\beta/2}$ and $\delta = 2^{-(\ell+\beta+\alpha)}$, we have*

$$\text{Noisy}_{\infty,\ell}(X) \subseteq \text{Dense}_{p,\gamma,\delta}(X).$$

Proof. Fix P_{XZ} such that $\tilde{\mathbb{H}}_{\infty}(X|Z) \geq \mathbb{H}_{\infty}(X) - \ell$. Then, by Lemma 2, with probability at least $1 - 2^{-\beta}$ over the fixing $Z = z$ it holds that

$$\mathbb{H}_{\infty}(X|Z = z) \geq \mathbb{H}_{\infty}(X) - \ell - \beta. \quad (6)$$

Let \mathcal{S} denote the set of such fixings z . By a simple averaging argument, there exists a set $\mathcal{T} \subseteq \mathcal{X}$ with $P_X(\mathcal{T}) \geq 1 - 2^{-\beta/2}$ such that $P_{Z|X=x}(\mathcal{S}) \geq 1 - 2^{-\beta/2}$ for all $x \in \mathcal{T}$. To see this, it suffices to note that since $P_Z(\mathcal{S}) \geq 1 - \varepsilon$, with probability at least $1 - \sqrt{\varepsilon}$ over x we must have $P_{Z|X=x}(\mathcal{S}) \geq 1 - \sqrt{\varepsilon}$. Suppose not. Then, with probability at least $\sqrt{\varepsilon}$ over x we have $P_{Z|X=x}(\mathcal{S}) < 1 - \sqrt{\varepsilon}$, and thus $P_Z(\mathcal{S}) < (1 - \sqrt{\varepsilon}) \cdot 1 + \sqrt{\varepsilon} \cdot (1 - \sqrt{\varepsilon}) = 1 - \varepsilon$, a contradiction.

We now show that for $x \in \mathcal{T}$ it is the case that $P_{Z|X=x}$ is $(\gamma = 2^{-\beta/2}, \delta = 2^{\ell+\beta+\alpha})$ -dense in P_Z by considering \mathcal{S} . Coupled with the properties of \mathcal{T} , this yields the desired result. For all $z \in \mathcal{S}$ and x we have

$$\begin{aligned} P_{X|Z=z}(x) &\leq 2^{-(\mathbb{H}_{\infty}(X) - \ell - \beta)} \\ &= 2^{\ell+\beta} \cdot \max_{x'} P_X(x') \\ &\leq 2^{\ell+\beta+\alpha} \cdot P_X(x), \end{aligned} \quad (7)$$

where, in the last inequality, we use the hypothesis that X is α -semi-flat. By noting that $\frac{P_{X|Z=z}(x)}{P_X(x)} = \frac{P_{Z|X=x}(z)}{P_Z(z)}$, we can rewrite Eq. (7) as

$$P_{Z|X=x}(z) \leq 2^{\ell+\beta+\alpha} \cdot P_Z(z).$$

If $x \in \mathcal{T}$, this implies that $P_{Z|X=x}$ is (γ, δ) -dense in P_Z for $\delta = 2^{-(\ell+\beta+\alpha)}$, as desired. \square

Combining Theorem 3 and Theorem 4, we immediately obtain the following corollary.

Corollary 2. *If X is α -semi-flat, then, for any $\beta > 0$ and $\varepsilon > 0$, the set of leakages $\text{Noisy}_{\infty,\ell}(X)$ is $(\varepsilon + \varepsilon^{2^{\ell+\beta+\alpha-2}} + 2^{-\beta/2+1})$ -simulatable from $\text{Bounded}_{\ell'}(X)$ with*

$$\ell' = \ell + \beta + \alpha + \log \log(1/\varepsilon) + 2 \log \left(\frac{1}{1 - 2^{-\beta/2}} \right) + 2.$$

The remark below says that there is a natural tradeoff between the simulation error in the above corollary and the leakage bound.

Remark 5 (Trading simulation error with ME-noisy leakage). *By choosing $\varepsilon = 2^{-\lambda}$ and $\beta = 2 + \log^2(\lambda)$ in Corollary 2, we can obtain negligible simulation error $\varepsilon' = \lambda^{-\omega(1)}$ with leakage¹⁰ $\ell' = \ell + O(\log^2(\lambda) + \alpha)$. By choosing $\beta = \lambda$, we can instead obtain a much smaller simulation error of $\varepsilon' = 2^{-\Omega(\lambda)}$ with larger leakage $\ell' = \ell + O(\lambda + \alpha)$.*

¹⁰In fact, we can push the leakage bound down to $\ell' = \ell + O(\log \log(\lambda) \log(\lambda) + \alpha)$ or even $\ell' = \ell + O(\log^*(\lambda) \log(\lambda) + \alpha)$, while still obtaining negligible simulation error.

Near-optimality of simulation theorem for ME-noisy leakage. We now show that our simulation result for ME-noisy leakage (Corollary 2) is essentially optimal. More precisely, we obtain the following result.

Theorem 5. *For every n and $\alpha, \ell > 0$ such that $\ell + \alpha < n - 2$ there exists an $(\alpha + 1)$ -semi-flat random variable X and $f \in \text{Noisy}_{\infty, \ell+2}(X)$ such that simulating $f(X)$ with error less than $1/4$ requires one ℓ' -bounded leakage query for $\ell' \geq \ell + \alpha - 1$.*

Essentially, Theorem 5 states that $\ell + \alpha - O(1)$ bits of bounded leakage are required to simulate ℓ -ME-noisy leakage from an α -semi-flat random variable X with useful simulation error. Our simulation theorem from Corollary 2 complements this negative result, showing that $\ell' \approx \ell + \alpha$ bits of bounded leakage are enough even with *negligible* simulation error.

Proof. Fix $\alpha, \ell > 0$ such that $\ell + \alpha < n - 2$. Consider X with distribution P_X satisfying

$$P_X(x) = \begin{cases} 2^{\alpha-n}, & \text{if } x = 0^n, \\ \frac{1-2^{\alpha-n}}{2^n-1}, & \text{otherwise.} \end{cases}$$

Observe that X is $(\alpha + 1)$ -semi-flat for n large enough, since

$$2^{\alpha+1} \cdot \frac{1-2^{\alpha-n}}{2^n-1} \geq 2(2^{\alpha-n} - 2^{2(\alpha-n)}) \geq 2^{\alpha-n}$$

by the constraint on α . Then, we define $Z = f(X)$ for the leakage function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\ell+\alpha}$ specified as follows: $f(0^n)$ is uniformly distributed over $\{0, 1\}^{\ell+\alpha}$, and $f(x) = (x_1, x_2, \dots, x_{\ell+\alpha})$ otherwise.

Our first goal is to show that $f \in \text{Noisy}_{\infty, \ell+2}(X)$. To begin, it is clear that $\mathbb{H}_{\infty}(X) = n - \alpha$. Therefore, the result follows if we show that

$$\mathbb{H}_{\infty}(X|Z = z) \geq n - \alpha - \ell - 2 \tag{8}$$

for all z . On the one hand, for all z we have

$$\begin{aligned} P_Z(z) &\geq \frac{2^{\alpha-n}}{2^{\ell+\alpha}} + \frac{1-2^{\alpha-n}}{2^n-1} \cdot (2^{n-(\alpha+\ell)} - 1) \\ &\geq 2^{-(n+\ell)} + \frac{1-2^{\alpha-n}}{2^n} \cdot (2^{n-(\alpha+\ell)} - 1) \\ &\geq 2^{-(\ell+\alpha+2)}. \end{aligned}$$

On the other hand, for all x and z we have

$$P_{XZ}(x, z) = \begin{cases} 2^{-(n+\ell)}, & \text{if } x = 0^n, \\ \frac{1-2^{\alpha-n}}{2^n-1}, & \text{if } x \neq 0^n \text{ and } z = (x_1, x_2, \dots, x_{\ell+\alpha}), \\ 0, & \text{otherwise.} \end{cases}$$

As a result, we conclude that

$$P_{X|Z=z}(x) = \frac{P_{XZ}(x, z)}{P_Z(z)} \leq \frac{1-2^{\alpha-n}}{2^n-1} \cdot 2^{\ell+\alpha+2} \leq 2^{\ell+\alpha+2-n},$$

for all x and z , which leads to Eq. (8), as desired.

It remains to see that simulating $f(X)$ with error less than $1/4$ requires one query of ℓ' -bounded leakage from X for $\ell' = \ell + \alpha - 2$. This holds because, conditioning on $X \neq 0^n$, we have that X is uniform over $\{0, 1\}^n \setminus \{0^n\}$ and $f(X) = (X_1, X_2, \dots, X_{\ell+\alpha})$. Therefore, under the conditioning above, an arbitrary simulator can only guess $f(X)$ with probability at most $1/2$ from any ℓ' -bounded leakage query $g(X)$ with $\ell' \leq \ell + \alpha - 1$. Since the event $X \neq 0^n$ holds with probability $1 - 2^{\alpha-n} \geq 3/4$, the simulation error is at least $1/2 - 1/4 = 1/4$, as desired. \square

Necessity of the semi-flatness assumption in Corollary 2. Theorem 5 implies that assuming α -semi-flatness of X is necessary to obtain a non-trivial simulation theorem for ME-noisy leakage, even when we are attempting to simulate only 0-ME-noisy leakage functions. Indeed, setting $\ell = 0$ and $\alpha = n - 3$ in Theorem 5, we conclude that there exists a random variable X along with an associated 0-ME-noisy-leakage function $f \in \text{Noisy}_{\infty,0}(X)$ that requires $n - O(1)$ bits of bounded leakage from X in order to be simulated with error less than $1/4$.

5.3 Uniform-Noisy Leakage is Also Dense Leakage

There is a known connection between U-noisy and ME-noisy leakage, i.e., every U-noisy leakage function is also a ME-noisy leakage function by itself.

Lemma 6 ([DHLW10]). *Given any randomized function $f : \mathcal{X} \rightarrow \mathcal{Z}$, if it holds that*

$$\tilde{\mathbb{H}}_{\infty}(U|f(U)) \geq \mathbb{H}_{\infty}(U) - \ell,$$

then for any X over \mathcal{X} it is the case that

$$\tilde{\mathbb{H}}_{\infty}(X|f(X)) \geq \mathbb{H}_{\infty}(X) - \ell.$$

In particular, this implies that $\text{UNoisy}_{\infty,\ell}(X) \subseteq \text{Noisy}_{\infty,\ell}(X)$.

We remark that there also exist some X and a leakage function f such that $f \in \text{Noisy}_{\infty,\ell}(X)$ but $f \notin \text{UNoisy}_{\infty,\ell}(X)$ (such an example is provided in [DHLW10]). This shows that the containment of U-noisy leakage in ME-noisy leakage may be strict for some X .

Although Lemma 6 immediately yields an analogue of Corollary 2 for U-noisy leakage, we can obtain a better result by arguing directly that every U-noisy leakage function is also a dense leakage function for arbitrary X , i.e., without requiring that X be semi-flat. Our result is stated formally in the next theorem.

Theorem 6. *For every $\beta > 0$ and arbitrary X , we have*

$$\text{UNoisy}_{\infty,\ell}(X) \subseteq \text{Dense}_{p,\gamma,\delta}(X).$$

where $p = 2^{-\beta/2}$, $\gamma = 2^{-\beta/2}$ and $\delta = 2^{-(\ell+\beta)}$.

Proof. Fix $Z = f(X)$ for randomized $f : \mathcal{X} \rightarrow \mathcal{Z}$ such that $f \in \text{UNoisy}_{\infty,\ell}(X)$. Our goal is to show that there exists a set $\mathcal{T} \subseteq \mathcal{X}$ with $P_X(\mathcal{T}) \geq 1 - p$ such that for every $x \in \mathcal{T}$ there exists a set $\mathcal{S} \subseteq \mathcal{Z}$ satisfying $P_{Z|X=x}(\mathcal{S}), P_Z(\mathcal{S}) \geq 1 - \gamma$, and

$$P_{Z|X=x}(z) \leq \frac{P_Z(z)}{\delta} \tag{9}$$

for all $z \in \mathcal{S}$, with p , γ , and δ as in the theorem statement.

Let U denote a uniform random variable over \mathcal{X} . Then, by hypothesis we have

$$\tilde{\mathbb{H}}_{\infty}(U|f(U)) \geq \mathbb{H}_{\infty}(U) - \ell,$$

or, equivalently,

$$2^{\ell} \geq \mathbb{E}_{y \leftarrow f(U)} \left[\max_{x \in \mathcal{X}} \frac{P_{U|f(U)=z}(x)}{P_U(x)} \right]. \tag{10}$$

Then, we have

$$\mathbb{E}_{y \leftarrow f(U)} \left[\max_{x \in \mathcal{X}} \frac{P_{U|f(U)=z}(x)}{P_U(x)} \right] = \sum_{z \in \mathcal{Z}} P_{f(U)}(z) \cdot \left(\max_{x \in \mathcal{X}} \frac{P_{U|f(U)=z}(x)}{P_U(x)} \right)$$

$$\begin{aligned}
&= \sum_{z \in \mathcal{Z}} \max_{x \in \mathcal{X}} P_{f(U)|U=x}(z) \\
&\geq \sum_{z \in \mathcal{Z}} \max_{x \in \text{supp}(X)} P_{f(U)|U=x}(z) \\
&= \sum_{z \in \mathcal{Z}} \max_{x \in \text{supp}(X)} P_{Z|X=x}(z) \\
&= \mathbb{E}_{z \leftarrow Z} \left[\max_{x \in \text{supp}(X)} \frac{P_{Z|X=x}(z)}{P_Z(z)} \right]. \tag{11}
\end{aligned}$$

Combining Eq. (10) and Eq. (11), we conclude that

$$2^\ell \geq \mathbb{E}_{z \leftarrow Z} \left[\max_{x \in \text{supp}(X)} \frac{P_{Z|X=x}(z)}{P_Z(z)} \right]. \tag{12}$$

Via a simple averaging argument, the upper bound in Eq. (12) implies that for every $\beta > 0$ there exists a set \mathcal{S} such that $P_Z(\mathcal{S}) \geq 1 - 2^{-\beta}$ and

$$\frac{P_{Z|X=x}(z)}{P_Z(z)} \leq 2^{\ell+\beta}$$

for all $z \in \mathcal{S}$ and $x \in \text{supp}(X)$. In turn, another averaging argument implies that there exists a set $\mathcal{T} \subseteq \mathcal{X}$ such that $P_X(\mathcal{T}) \geq 1 - 2^{-\beta/2}$ and for every $x \in \mathcal{T}$ we have $P_{Z|X=x}(\mathcal{S}), P_Z(\mathcal{S}) \geq 1 - 2^{-\beta/2}$. This implies the desired result with $p = \gamma = 2^{-\beta/2}$ and $\delta = 2^{-(\ell+\beta)}$. \square

Combining Theorem 3 and Theorem 6 immediately yields the following result.

Corollary 3. *For every X and every $\beta > 0$ and $\varepsilon > 0$, the set of leakages $\text{UNoisy}_{\infty, \ell}(X)$ is $(\varepsilon + \varepsilon^{2^{\ell+\beta-2}} + 2^{-\beta/2+1})$ -simulatable from $\text{Bounded}_{\ell'}(X)$ with*

$$\ell' = \ell + \beta + \log \log(1/\varepsilon) + 2 \log \left(\frac{1}{1 - 2^{-\beta/2}} \right) + 2.$$

The remark below says that there is a natural tradeoff between the simulation error in the above corollary and the leakage bound.

Remark 6 (Trading simulation error with U-noisy leakage). *By choosing $\varepsilon = 2^{-\lambda}$ and $\beta = 2 + \log^2(\lambda)$ in Corollary 3, we can obtain negligible simulation error $\varepsilon' = \lambda^{-\omega(1)}$ with leakage $\ell' = \ell + O(\log^2(\lambda))$. By choosing $\beta = \lambda$, we can instead obtain a much smaller simulation error of $\varepsilon' = 2^{-\Omega(\lambda)}$ with larger leakage $\ell' = \ell + O(\lambda)$.*

Near-optimality of simulation theorem for U-noisy leakage. We now show that in order to simulate arbitrary ℓ -U-noisy leakage from X uniformly distributed over $\{0, 1\}^n$ with simulation error less than $1/2$, we need access to one query of ℓ' -bounded leakage from X for $\ell' \geq \ell - 1$. As we shall see, this result implies that our simulation theorem from Corollary 3 is nearly optimal.

Consider the following example: Let X be uniformly distributed over $\{0, 1\}^n$ and consider the *deterministic* function $f : \{0, 1\}^n \rightarrow \{0, 1\}^\ell$ defined as $f(x) = (x_1, x_2, \dots, x_\ell)$. Then, it is clear that $f \in \text{UNoisy}_{\infty, \ell}(X)$, and that $f(X)$ is uniformly distributed over $\{0, 1\}^\ell$. If $g \in \text{Bounded}_{\ell-1}(X)$, it then follows by Lemma 1 that $\tilde{\mathbb{H}}_\infty(f(X)|g(X)) \geq 1$, and hence we cannot guess $f(X)$ correctly from $g(X)$ with probability more than $1/2$. Finally, it is also clear

that $f \in \text{Dense}_{p=0, \gamma=0, \delta=2^{-\ell}}$. Indeed, if $Z = f(X)$, then we have $P_Z(z) = 2^{-\ell}$ for all $z \in \{0, 1\}^\ell$, and hence trivially it holds that

$$P_{Z|X=x}(z) \leq \frac{P_Z(z)}{\delta} = 1$$

for all x and z , when $\delta = 2^{-\ell}$. Summarizing the discussion above leads to the following result.

Theorem 7. *For X uniform over $\{0, 1\}^n$ and every $\ell \geq 1$ there exists $f \in \text{UNoisy}_{\infty, \ell}(X) \subseteq \text{Noisy}_{\infty, \ell}(X)$ such that $f(X)$ cannot be simulated with error less than $1/2$ by one $(\ell - 1)$ -bounded leakage query to X . Moreover, it also holds that $f \in \text{Dense}_{p=0, \gamma=0, \delta=2^{-\ell}}(X)$.*

Comparing Theorem 7 with Corollary 3, we see that our simulation theorem for U-noisy leakage is nearly optimal with respect to the bounded leakage parameter, since we only require approximately ℓ bits of bounded leakage to simulate U-noisy leakage for uniform X . Furthermore, we can achieve this result with negligible simulation error.

5.4 SD-Noisy and MI-Noisy Leakage are also Dense Leakage

We now proceed to relate SD-noisy leakage and dense leakage.

Theorem 8. *For any $\gamma > 0$ and arbitrary X , we have*

$$\text{Noisy}_{\Delta, \eta}(X) \subseteq \text{Dense}_{p, \gamma, \delta}(X)$$

with $p = 2\eta/\gamma$ and $\delta = 1/2$.

Proof. Fix $Z = f(X)$ for $f \in \text{Noisy}_{\Delta, \eta}(X)$. Then, we have

$$\mathbb{E}_{(x, z) \sim P_X \otimes P_Z} \left[\frac{|P_{XZ}(x, z) - P_X(x)P_Z(z)|}{P_X(x)P_Z(z)} \right] = \sum_{x \in \mathcal{X}, z \in \mathcal{Z}} |P_{XZ}(x, z) - P_X(x)P_Z(z)| \leq 2\eta.$$

By Markov's inequality, we conclude that there exists a set $\mathcal{W} \subseteq \mathcal{X} \times \mathcal{Z}$ such that $P_{X \otimes Z}(\mathcal{W}) \geq 1 - 2\eta$ we have

$$\frac{|P_{XZ}(x, z) - P_X(x)P_Z(z)|}{P_X(x)P_Z(z)} \leq 1$$

for all $(x, z) \in \mathcal{W}$. Then, this implies that

$$P_{XZ}(x, z) \leq 2P_X(x)P_Z(z),$$

and hence

$$P_{Z|X=x}(z) \leq 2P_Z(z)$$

for all $(x, z) \in \mathcal{W}$. Letting $W_x = \{z \in \mathcal{Z} : (x, z) \in \mathcal{W}\}$, an averaging argument implies that, for arbitrary $\gamma > 0$, there exists a set $\mathcal{T} \subseteq \mathcal{X}$ with $P_X(\mathcal{T}) \geq 1 - 2\eta/\gamma = 1 - p$ such that $P_Z(W_x) \geq 1 - \gamma$ for all $x \in \mathcal{T}$. Indeed, if this is not the case, then

$$P_{X \otimes Z}(\mathcal{W}) = \sum_{x \in \mathcal{X}} P_Z(W_x) < 2\eta/\gamma \cdot (1 - \gamma) + 1 - 2\eta/\gamma = 1 - 2\eta,$$

a contradiction. □

By combining Corollary 1 and Theorem 8, we immediately obtain an analogous result for MI-noisy leakage.

Theorem 9. *For any $\gamma > 0$ and arbitrary X , we have*

$$\text{MINoisy}_{\eta}(X) \subseteq \text{Dense}_{p, \gamma, \delta}(X)$$

with $p = \sqrt{8\eta}/\gamma$ and $\delta = 1/2$.

Near-optimality of trivial simulator for SD-noisy leakage. While in principle one can combine Theorem 8 and Theorem 9 with Theorem 3 in order to obtain simulation theorems for SD-noisy and MI-noisy leakage from bounded leakage, it turns out that these simulation theorems would be trivial in the sense that the same statements can be obtained without even considering reductions to bounded leakage.

In particular, consider the following trivial simulator Sim_f for SD-noisy leakage: Given some P_X and a function $f \in \text{Noisy}_{\Delta, \eta}(X)$, the simulator Sim_f samples \tilde{X} according to P_X and outputs $\tilde{Z} = f(\tilde{X})$ without querying any bounded leakage function. By the definition of statistical distance and the fact that $f \in \text{Noisy}_{\Delta, \eta}(X)$, we have

$$(X, Z) \approx_{\eta} (X, \tilde{Z}),$$

meaning that every η -SD-noisy leakage function f can be trivially simulated with statistical error η even from the trivial family of 0-bounded leakage functions. By Pinsker's inequality, the above trivial simulation strategy also shows that for arbitrary X every leakage function $f \in \text{MINoisy}_{\eta}(X)$ can be trivially simulated from 0-bounded leakage with error at most $\sqrt{2\eta}$.

Below, we prove that the trivial simulator for SD-Noise leakage is essentially optimal, in the sense that *any* simulator for SD-noisy leakage with access to one query of ℓ -bounded leakage for any $\ell < n$ must have error $\eta/2$. We have the following theorem.

Theorem 10. *For every $\eta \in (0, 1)$, there exist $X \in \{0, 1\}^n$ and $f \in \text{Noisy}_{\Delta, \eta}(X)$ such that every simulator for f with respect to ℓ -bounded leakage has error at least $\eta/2$ for any $\ell < n$.*

Proof. Consider $X \in \{0, 1\}^n$ and a function f inducing uniform correlated leakage $Z = f(X) \in \{0, 1\}^n$ satisfying

$$P_{X|Z=z}(x) = \begin{cases} \eta, & \text{if } x = z, \\ \frac{1-\eta}{2^n-1}, & \text{otherwise,} \end{cases} \quad (13)$$

where $\eta \in (0, 1)$. Given Eq. (13), it follows easily that X is also uniformly distributed over $\{0, 1\}^n$, and that

$$\Delta(P_{XZ}; P_X \otimes P_Z) \leq \eta.$$

Therefore, we have that $f \in \text{Noisy}_{\Delta, \eta}(X)$.

Consider now an arbitrary simulator Sim_f with access to one query of ℓ -bounded leakage for some $\ell < n$. We argue that Sim_f must have error at least $\eta/2$. Indeed, note that X is still uniformly distributed over $\{0, 1\}^n$ when conditioned on the event $X = Z$. Therefore, under this conditioning, the simulator Sim_f can only guess Z with probability at most $1/2$ if it only has access to one query of ℓ -bounded leakage from X for $\ell < n$. Since $\mathbb{P}[X = Z] = \eta$, the observation above implies that the error of Sim_f is at least $\eta/2$, as desired. \square

Combining the trivial simulator for η -SD-noisy leakage and Theorem 10, we conclude that it is impossible to obtain a general simulation theorem for SD-noisy leakage from bounded leakage that performs somewhat better than the trivial simulator. Indeed, it is even impossible to improve upon the error of the trivial simulator by more than a $1/2$ multiplicative factor.

We remark that Theorem 10 does not contradict the simulation theorems of Duc, Dziembowski, and Faust [DDF14, DDF19]. The reason for this, as we discussed in §4.2, is that the set of leakage functions considered in [DDF14, DDF19], which we termed DDF-noisy leakage, is a very restricted subset of the set of SD-noisy leakage functions of a random variable X . In fact, we show in §5.5 that DDF-noisy leakage is a subset of U-noisy leakage. This means that, via our general simulation theorem for dense leakage, and in contrast with general SD-noisy leakage, DDF-noisy leakage can be simulated in a non-trivial way with small simulation error and good parameters from bounded leakage.

Impossibility of low-error simulation of MI-noisy leakage. Via Pinsker's inequality, we know that the trivial simulator achieves simulation error $\sqrt{2\eta}$ for η -MI-noisy leakage. Therefore, it is natural to wonder whether it is possible to do significantly better than this. In particular, in line with min-entropy-noisy and uniform-noisy leakages, is it possible to design a simulator with negligible simulation error when η is non-negligible?

We show that the answer to the question above is negative even when the simulator has access to one ℓ -bounded leakage query for *any* $\ell < n$. As a corollary, and similarly to the case of SD-noisy leakage, we also conclude that one cannot do significantly better than the trivial simulator. More precisely, we have the following result.

Theorem 11. *There exists X uniform over $\{0, 1\}^n$ and a function $f \in \text{MINoisy}_\eta(X)$ such that every simulator which makes one ℓ -bounded leakage query for any $\ell < n$ has simulation error at least $\frac{\eta}{2n}$. In particular, if η is non-negligible in n , then so is the simulation error.*

Proof. Consider X uniform over $\{0, 1\}^n$ and the randomized leakage function f satisfying $f(x) = x$ with probability $\eta/n < 1$, and $f(x) = \perp$ otherwise. We begin by showing that $f \in \text{MINoisy}_\eta(X)$. Letting $Z = f(X)$, we have

$$I(X; Z) = H(X) - H(X|Z) = n - H(X|Z) = n - P_Z(\perp) \cdot n = n - \eta,$$

as desired. Consider an arbitrary simulator for X and Z with access to one query of ℓ -bounded leakage with $\ell < n$. Since the simulator can only guess X with probability at most $1/2$ and $Z = X$ holds with probability η/n , we conclude that the simulation error is at least $\frac{\eta}{2n}$. \square

5.5 DDF-Noisy Leakage is Uniform-Noisy Leakage

We begin by showing that DDF-noisy leakage is a subset of U-noisy leakage with good parameters, and hence of dense leakage as well. Then, we discuss the consequences of this fact.

Theorem 12. *For every $X \in \mathcal{X}^n$ and $\eta \leq 1/|\mathcal{X}|$, it holds that*

$$\text{Noisy}_{\text{DDF}, \eta}(X) \subseteq \text{UNoisy}_{\infty, \ell}(X)$$

for $\ell = n \log(1 + \eta(|\mathcal{X}| - 1)|\mathcal{X}|)$. In particular, when $\mathcal{X} = \{0, 1\}$ we have $\ell = n \log(1 + 2\eta) \leq 2\eta \cdot n$.

Proof. Fix some X distributed over \mathcal{X}^n and some $f \in \text{Noisy}_{\text{DDF}, \eta}(X)$. It was shown in [DDF14, Lemma 3] that there exists a possibly randomized function g such that $(X, f(X))$ is distributed exactly like $(X, g(\hat{X}))$, where \hat{X} is obtained from X by independently setting each $\hat{X}_i = X_i$ with probability $\delta = \eta \cdot |\mathcal{X}|$, and $\hat{X}_i = \perp$ otherwise.¹¹

Given the above, it suffices to show that \hat{X} is ℓ -U-noisy leakage from X with $\ell = n \log(1 + \eta(|\mathcal{X}| - 1)|\mathcal{X}|)$. Let U be uniformly distributed over \mathcal{X}^n , \hat{U} be the corresponding corrupted version of U obtained by independently setting each $\hat{U}_i = U_i$ with probability δ and $\hat{U}_i = \perp$ otherwise, and $w(\hat{u})$ denote the number of entries of \hat{u} not equal to \perp . Note that $w(\hat{U})$ follows a binomial distribution with n trials and success probability $\delta = \eta \cdot |\mathcal{X}|$. Then, we have

$$\begin{aligned} \tilde{\mathbb{H}}_\infty(U|\hat{U}) &= -\log\left(\mathbb{E}_{\hat{u} \leftarrow \hat{U}}\left[2^{-\mathbb{H}_\infty(U|\hat{U}=\hat{u})}\right]\right) \\ &= -\log\left(\mathbb{E}_{\hat{u} \leftarrow \hat{U}}\left[2^{-(n-w(\hat{u}))\log|\mathcal{X}|}\right]\right) \\ &= -n \log\left(\delta + (1-\delta) \cdot \frac{1}{|\mathcal{X}|}\right) \end{aligned}$$

¹¹This random variable \hat{X} is also called *random-probing leakage* from X in the literature.

$$= \mathbb{H}_\infty(U) - n \log(1 + \eta(|\mathcal{X}| - 1)|\mathcal{X}|),$$

where the third equality follows by noting that $Z = n - w(\hat{U})$ is distributed according to a binomial distribution with n trials and success probability $1 - \delta$. Then, the moment-generating function of Z is given by

$$\mathbb{E}[e^{tZ}] = (\delta + (1 - \delta)e^t)^n$$

for all $t \in \mathbb{R}$. Setting $t = -\ln 2 \cdot \log |\mathcal{X}|$ yields the desired result. Finally, the fourth equality holds by noting that $\mathbb{H}_\infty(U) = n \log |\mathcal{X}|$ and $\delta = \eta|\mathcal{X}|$. \square

Recalling Corollary 3, which gives a nearly optimal simulation theorem for U-noisy leakage from bounded leakage, via Theorem 12 we conclude that we can simulate all η -DDF-noisy leakage with roughly ℓ bits of bounded leakage, for ℓ as in Theorem 12, with negligible simulation error independent of η , which may even be constant. More precisely, we have the following result when $\mathcal{X} = \{0, 1\}$.

Corollary 4. *For every $X \in \{0, 1\}^n$, $\eta \leq 1/2$, $\beta > 0$, and $\varepsilon > 0$, the set of leakages $\text{DDFNoisy}_{\Delta, \eta}(X)$ is $(\varepsilon + \varepsilon^{2^{\ell+\beta-2}} + 2^{-\beta/2+1})$ -simulatable from $\text{Bounded}_{\ell'}(X)$ with*

$$\ell' = \ell + \beta + \log \log(1/\varepsilon) + 2 \log \left(\frac{1}{1 - 2^{-\beta/2}} \right) + 2$$

and $\ell = n \log(1 + 2\eta)$.

It is interesting to contrast this result with Theorem 10, which states that we cannot simulate general η -SD-noisy leakage with error less than $\eta/2$. We remark that the simulation result we obtain for DDF-noisy leakage here is not new. Such a simulation result for DDF-noisy leakage was derived in [DDF14, DDF19] via a reduction to a restricted subset of bounded leakage called *threshold probing leakage*. We chose to nevertheless include the results of this section to showcase the generality of our approach and emphasize the differences between general SD-noisy leakage and DDF-noisy leakage.

5.6 Missing Separations between Types of Leakage

With the final goal of having a clear picture of the inclusions and separations between different leakage models (as depicted in Figure 1), we showcase here the missing separations between the different sets of leakages considered above. These correspond to the relationship between SD-noisy leakage and other leakage models.

Separation between SD-noisy and ME-noisy leakages. We begin by showing that there are SD-noisy leakages with very good parameters which are only ME-noisy leakages with poor parameters. As we shall see, the leakage that we consider to obtain this separation can also be used to obtain a lower bound on the error required to simulated SD-noisy leakage with bounded leakage and to separate SD-noisy and MI-noisy leakages. Consider X and $Z = f(X)$ uniformly distributed over $\{0, 1\}^n$ as defined in Eq. (13) for $\eta \in (0, 1)$. As discussed in Section 5.4, from Eq. (13) it follows easily that X is also uniformly distributed over $\{0, 1\}^n$, and that

$$\Delta(P_{XZ}; P_X \otimes P_Z) \leq \eta.$$

Therefore, we have $f \in \text{Noisy}_{\Delta, \eta}(X)$. However, on the other hand we have $\mathbb{H}_\infty(X|Z = z) = \log(1/\eta)$ for every $z \in \{0, 1\}^n$, and thus

$$\tilde{\mathbb{H}}_\infty(X|Z) = \log(1/\eta).$$

This means that $f \notin \text{Noisy}_{\infty,\ell}(X)$ for any $\ell < n - \log(1/\eta)$. We summarize the above in the following result.

Theorem 13. *For every $\eta \in (0, 1)$ and n , there exists a function f such that for X uniform over $\{0, 1\}^n$ we have $f \in \text{Noisy}_{\Delta,\eta}(X)$ but $f \notin \text{Noisy}_{\infty,\ell}(X)$ for any $\ell < n - \log(1/\eta)$.*

Separation between SD-noisy and MI-noisy leakages, and a counter-example to a reverse Pinsker inequality for approximate density. The same joint distribution P_{XZ} defined in Eq. (13) can be used to give a separation between SD-noisy and MI-noisy leakages whenever η is not a small function of n (e.g., if η is an arbitrary constant independent of n). Namely, we have

$$\begin{aligned} I(X; Z) &= H(X) - H(X|Z) \\ &= n - H(X|Z = 0^n) \\ &= n + \eta \log \eta + (1 - \eta)(\log(1 - \eta) - \log(2^n - 1)), \end{aligned}$$

where in the second equality we used the fact that X is uniform. If $\eta \geq 1/n^\alpha$ for some $\alpha < 1$, then we conclude that $I(X; Z) \geq (1 - o(1))\eta n \geq (1 - o(1))n^{1-\alpha}$ as $n \rightarrow \infty$. On the other hand, as discussed above, we have $\Delta(P_{XZ}; P_X \otimes P_Z) \leq \eta$. We summarize the discussion above in the following theorem.

Theorem 14. *If $\eta \geq 1/n^\alpha$ for any constant $\alpha > 0$, then for n large enough there exists a function f such that for X uniform over $\{0, 1\}^n$ it holds that $f \in \text{Noisy}_{\Delta,\eta}(X)$ but $f \notin \text{MINoisy}_{\eta'}(X)$ for any $\eta' < \frac{\eta n}{2}$.*

Interestingly, the example above also shows that a certain *reverse Pinsker inequality* cannot be extended to more general pairs of distributions. We proceed to explain this in more detail. Given that the Pinsker inequality (Lemma 5) holds for all distributions $P = P_{XZ}$ and $P' = P_X \otimes P_Z$, it is natural to wonder under which conditions on P and P' we can have a reverse Pinsker inequality of the form

$$I(X; Z) = D_{\text{KL}}(P_{XZ} \| P_X \otimes P_Z) \leq C \cdot \Delta(P_{XZ}; P_X \otimes P_Z)$$

for some absolute leading constant $C > 0$. A more detailed treatment of such inequalities can be found in [Sas15]. Verdú [Ver14, Theorem 7] showed that such a reverse Pinsker inequality holds in particular whenever P_{XZ} is δ -dense in $P_X \otimes P_Z$,¹² in which case one obtains

$$I(X; Z) \leq C_\delta \cdot \Delta(P_{XZ}; P_X \otimes P_Z) \tag{14}$$

for $C_\delta = \frac{\log(1/\delta)}{2(1-\delta)}$. Observe that the distribution P_{XZ} with $Z = f(X)$ defined in Eq. (13) is such that

$$f \in \text{Dense}_{p=\sqrt{\eta}, \gamma=\sqrt{\eta}, \delta=1/2}(X)$$

by Theorem 8, where η can be made arbitrarily small. Therefore, this property and Theorem 14 show that Eq. (14) cannot possibly hold with *any* leading constant C if we replace the hypothesis that P_{XZ} is δ -dense in $P_X \otimes P_Z$ by the slightly weaker hypothesis that $f \in \text{Dense}_{p,\gamma,\delta}(X)$, i.e., that P_{XZ} is *approximately* δ -dense in $P_X \otimes P_Z$ for even very small errors p and γ .

¹²Note that this condition is equivalent to assuming that $P_{Z|X=x}$ is δ -dense in P_Z for every x .

6 Applications

In this section we show that our results have interesting implications for so-called leakage-resilient cryptography. In particular, we will show that many cryptographic primitives that have been shown to be resilient to bounded leakage are also resilient to different forms of noisy leakage, with only a small loss in parameters.

6.1 Secret Sharing with Local Leakage Resilience

A secret sharing scheme consists of n players, or parties, and two polynomial-time algorithms $\Sigma = (\text{Share}, \text{Rec})$ specified as follows:

- The randomized sharing algorithm $\text{Share}(y)$ takes as input a message $y \in \{0, 1\}^m$, and outputs n shares x_1, \dots, x_n , with $x_i \in \{0, 1\}^{s_i}$ for each $i \in [n]$.
- The deterministic reconstruction algorithm $\text{Rec}((x_i)_{i \in \mathcal{I}})$ takes as input a certain subset \mathcal{I} of shares $(x_i)_{i \in \mathcal{I}}$ and outputs a value in $\{0, 1\}^m$.

Intuitively, in a *threshold* secret sharing scheme, we want that, once fixed a threshold $t \leq n$, only the subsets of $[n]$ of at least t players are allowed to reconstruct the message, while any subset of at most $t - 1$ players cannot obtain any information about the message from their shares. Usually, we refer to this kind of secret sharing scheme as t -out-of- n secret sharing.

Definition 13 (Threshold secret sharing). *Let $t, n \in \mathbb{N}$ and $\varepsilon \in [0, 1]$ be parameters. We say that $\Sigma = (\text{Share}, \text{Rec})$ is a t -out-of- n ε -statistical secret sharing scheme if the following two properties hold.*

- **Correctness:** *for all $y \in \{0, 1\}^m$ and all $\mathcal{I} \subseteq [n]$ such that $|\mathcal{I}| \geq t$, it holds that $\text{Rec}(\text{Share}(y)_{\mathcal{I}}) = y$ with probability 1 over the randomness of Share .*
- **Privacy:** *for all $y_0, y_1 \in \{0, 1\}^m$ and all $\mathcal{U} \subseteq [n]$ such that $|\mathcal{U}| < t$, it holds that $\Delta(\text{Share}(y_0), \text{Share}(y_1)) \leq \varepsilon$.*

In the special case of $\varepsilon = 0$ (i.e., $\text{Share}(y_0)$ and $\text{Share}(y_1)$ are identically distributed), we speak of t -out-of- n perfect secret sharing.

Local leakage resilience. Next, we consider the following kind of leakage attack: after seeing an unauthorized subset of shares, the adversary performs one query of leakage from all the shares independently.

Definition 14 (Local leakage-resilient secret sharing). *Let $t, n, u \in \mathbb{N}$ be parameters such that $u < t \leq n$, and let $\Sigma = (\text{Share}, \text{Rec})$ be a t -out-of- n secret sharing scheme. We say that Σ is a (p, γ, δ) -dense u -local ε -leakage-resilient secret sharing scheme (or $(u, p, \gamma, \delta, \varepsilon)$ -DLLR-SS for short) if for all messages $y_0, y_1 \in \{0, 1\}^m$, all unauthorized subsets $\mathcal{U} \subseteq [n]$ such that $|\mathcal{U}| \leq u$, and every tuple of leakage functions (f_1, \dots, f_n) such that $f_i : \{0, 1\}^{s_i} \rightarrow \{0, 1\}^*$ is (p, γ, δ) -dense for all $i \in [n]$, we have*

$$\Delta((X_{\mathcal{U}}^0, (f_i(X_{\mathcal{U}}^0, X_i^0))_{i \in [n]}), (X_{\mathcal{U}}^1, (f_i(X_{\mathcal{U}}^1, X_i^1))_{i \in [n]})) \leq \varepsilon,$$

where $(X_1^b, \dots, X_n^b) = \text{Share}(y_b)$ for all $b \in \{0, 1\}$.

Moreover, in case the functions f_i in the above definition are:

- ℓ -bounded leakage functions, we say that Σ is ℓ -bounded u -local ε -leakage-resilient (or (u, ℓ, ε) -BLLR-SS);
- ℓ -ME-noisy leakage functions, we say that Σ is ℓ -min-entropy-noisy u -local ε -leakage-resilient (or (u, ℓ, ε) -ME-NLLR-SS);
- ℓ -U-noisy leakage functions, we say that Σ is ℓ -uniform-noisy u -local ε -leakage-resilient (or (u, ℓ, ε) -U-NLLR-SS);

The theorem below says that any bounded leakage-resilient secret sharing scheme is also secure in the presence of dense leakage.

Theorem 15. *Any (u, ℓ, ε) -BLLR-SS is also a $(u, p, \gamma, \delta, \varepsilon')$ -DLLR-SS so long as*

$$\begin{aligned} \ell &= \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2 \\ \varepsilon' &= (2n+1)\varepsilon + 2n\varepsilon^{1/4\delta} + 2n\gamma + 2np. \end{aligned}$$

Proof. Fix any unauthorized set \mathcal{U} such that $|\mathcal{U}| \leq u$, n arbitrary (p, γ, δ) -dense leakage functions f_1, \dots, f_n and any $y_0, y_1 \in \{0, 1\}^m$. For simplicity and ease of notation, let $\hat{\varepsilon} = \varepsilon + \varepsilon^{1/4\delta} + \gamma + p$ and wlog. assume that f_i does not output anything for all $i \in \mathcal{U}$. By Theorem 3, there exist n randomized simulators Sim_{f_i} such that for all $b \in \{0, 1\}$,

$$\begin{aligned} &(X_{\mathcal{U}}^b, f_1(X_1^b), f_2(X_2^b), \dots, f_n(X_n^b)) \\ &\approx_{\hat{\varepsilon}} (X_{\mathcal{U}}^b, \text{Sim}_{f_1}^{\text{Leak}(X_1^b, \cdot)}, f_2(X_2^b), \dots, f_n(X_n^b)) \\ &\approx_{\hat{\varepsilon}} \dots \\ &\approx_{\hat{\varepsilon}} (X_{\mathcal{U}}^b, \text{Sim}_{f_1}^{\text{Leak}(X_1^b, \cdot)}, \text{Sim}_{f_2}^{\text{Leak}(X_2^b, \cdot)}, \dots, \text{Sim}_{f_n}^{\text{Leak}(X_n^b, \cdot)}), \end{aligned} \quad (15)$$

where, for all $i \in [n]$, X_i^b is the random variable associated to the i -th share of a secret sharing of y_b and Sim_{f_i} asks to its oracle a single query consisting of a function $f'_i : \{0, 1\}^{s_i} \rightarrow \{0, 1\}^{\ell}$, with $\ell = \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2$.

Next, we claim that for every sequence of (p, γ, δ) -dense leakage functions $f_i : \{0, 1\}^{s_i} \rightarrow \{0, 1\}^*$ and for all messages $y_0, y_1 \in \{0, 1\}^m$,

$$(X_{\mathcal{U}}^0, \text{Sim}_{f_1}^{\text{Leak}(X_1^0, \cdot)}, \dots, \text{Sim}_{f_n}^{\text{Leak}(X_n^0, \cdot)}) \approx_{\varepsilon} (X_{\mathcal{U}}^1, \text{Sim}_{f_1}^{\text{Leak}(X_1^1, \cdot)}, \dots, \text{Sim}_{f_n}^{\text{Leak}(X_n^1, \cdot)}) \quad (16)$$

By contradiction, assume that there exists a sequence of (p, γ, δ) -dense leakage functions f_1, \dots, f_n , a pair of messages $y_0, y_1 \in \{0, 1\}^m$, and an unbounded distinguisher D such that

$$\begin{aligned} &\mathbb{P}\left[D(X_{\mathcal{U}}^0, \text{Sim}_{f_1}^{\text{Leak}(X_1^0, \cdot)}, \dots, \text{Sim}_{f_n}^{\text{Leak}(X_n^0, \cdot)}) = 1\right] \\ &\quad - \mathbb{P}\left[D(X_{\mathcal{U}}^1, \text{Sim}_{f_1}^{\text{Leak}(X_1^1, \cdot)}, \dots, \text{Sim}_{f_n}^{\text{Leak}(X_n^1, \cdot)}) = 1\right] > \varepsilon. \end{aligned}$$

For all $i \in [n]$, let $f'_i : \{0, 1\}^{s_i} \rightarrow \{0, 1\}^{\ell}$ be the leakage function output by Sim_{f_i} and denote by \tilde{Z}_i the output of the respective simulator. Consider the distinguisher D' that upon receiving $(X_{\mathcal{U}}^b, (f'_i(X_{\mathcal{U}}^b, X_i^b))_{i \in [n]})$ returns the same as D upon input $(X_{\mathcal{U}}^b, (\tilde{Z}_1, \dots, \tilde{Z}_n))$. Then,

$$\begin{aligned} &\mathbb{P}\left[D'(X_{\mathcal{U}}^0, f'_1(X_{\mathcal{U}}^0, X_1^0), \dots, f'_n(X_{\mathcal{U}}^0, X_n^0)) = 1\right] \\ &\quad - \mathbb{P}\left[D'(X_{\mathcal{U}}^1, f'_1(X_{\mathcal{U}}^1, X_1^1), \dots, f'_n(X_{\mathcal{U}}^1, X_n^1)) = 1\right] > \varepsilon. \end{aligned}$$

The above contradicts the fact that Σ is a (u, ℓ, ε) -BLLR-SS scheme. The theorem now follows by combining Eq. (15) and Eq. (16) with the triangle inequality. \square

Next, using the connection between ME-noisy and U-noisy leakage with dense leakage established in §5, we obtain the following corollary.

Corollary 5. *Any (u, ℓ', ε') -BLLR-SS is also an:*

(i) (u, ℓ, ε) -ME-NLLR-SS so long as $\ell = \ell' - 2\log(1/\varepsilon') - \alpha - \log\log(1/\varepsilon') - 1$ and $\varepsilon = (6n+1)\varepsilon'$, and assuming that $(X_1, \dots, X_n) = \text{Share}(y)$ is such that X_i is α -semi-flat for all $i \in [n]$.

(ii) (u, ℓ, ε) -U-NLLR-SS so long as $\ell = \ell' - 2\log(1/\varepsilon') - \log\log(1/\varepsilon') - 1$ and $\varepsilon = (6n+1)\varepsilon'$.

Proof. The statement follows by choosing $\beta = 2 + 2\log(1/\varepsilon')$ and $\varepsilon = \varepsilon'$ in Corollary 2 and Corollary 3. \square

As a concrete instantiation, we use the results of Benhamouda *et al.* [BDIR18], who proved that both additive and Shamir's secret sharing achieve bounded leakage resilience for certain ranges of parameters. For concreteness, we focus on additive secret sharing but similar results can be derived for Shamir's secret sharing.

In order to apply the results of Corollary 5, we need to ensure that the parameters obtained by [BDIR18] satisfy $\ell \geq 0$, i.e. we require that, simplifying the expression, $\ell' \geq -4\log(\varepsilon')$. Considering [BDIR18, Corollary 4.11], for large enough q and for $\eta > 0$ such that $\ell' = \lfloor \log(q) - \eta \rfloor$, they achieve $\varepsilon' \leq 2^{\ell' - c(n-u-2)/2q^2}$ for $c = \pi^2(2^{2\eta} - 1)/(6\ln(2))$, that is:

$$\log(\varepsilon') \leq \ell' - c(n-u-2)/2q^2.$$

By asking that $c(n-u-2)/2q^2 \geq -5\log(\varepsilon')$, we obtain that $\ell' \geq -4\log(\varepsilon')$ and we are able to apply Corollary 5.

Fix ℓ^* and let $\eta = \log(q) - \ell^*$. Then, we are asking that

$$\begin{aligned} -5\log(\varepsilon') &\leq \frac{\pi^2(2^{2\eta} - 1)}{6\ln(2)} \frac{n-u-2}{2q^2} \\ &= \frac{\pi^2(q^2 \cdot 2^{\ell^*} - 1)}{6\ln(2)} \frac{n-u-2}{2q^2} \\ &= \frac{\pi^2}{12\ln(2)} (n-u-2) \left(2^{\ell^*} - \frac{1}{q^2} \right) \end{aligned}$$

Finally, by letting $\varepsilon = (6n+1)\varepsilon'$, our goal becomes

$$-5\log(\varepsilon) \leq \frac{\pi^2}{12\ln(2)} (n-u-2) \left(2^{\ell^*} - \frac{1}{q^2} \right) - 5\log(6n+1)$$

and, recalling that $\ell' \approx \ell^*$ (since $\ell' = \lfloor \ell^* \rfloor$), we obtain $\ell' \geq 4\log(6n+1) - 4\log(\varepsilon)$ whenever $n^4(n-u) = \Omega(\log(1/\varepsilon)/\varepsilon)$.

By applying Corollary 5 to the above results, we obtain the following.

Corollary 6. *For all $u, n \in \mathbb{N}$, with $u < n$, the n -out-of- n additive secret sharing scheme over \mathbb{F}_q , where q is a large prime, is an (u, ℓ, ε) -ME-NLLR-SS (and thus an (u, ℓ, ε) -U-NLLR-SS) so long as $\ell = O(\log(n) - \log(\varepsilon))$ and $n^4(n-u) = O(\log(1/\varepsilon)/\varepsilon)$*

In particular, the above corollary includes the following cases:

- we obtain ε -security for an arbitrarily small constant ε against constant leakage whenever an appropriate (constant) number of (honest) parties is involved;
- we obtain $\frac{1}{\lambda}$ -security against $O(\log(\lambda))$ bits of leakage whenever $n^4(n-u) = \Omega(\lambda \cdot \log(\lambda))$.

We stress that while the above parameters are somewhat limited (as, e.g., a super-polynomial number of honest parties is required in order to achieve negligible security), this limitation is inherited from the analysis of additive secret sharing done in [BDIR18].

6.2 Leakage Resilience in the Floppy Model

Next, we show our results have applications to constructing so-called continuously leakage-resilient primitives in the floppy model (also called the model of invisible key updates [ADW09]). For concreteness, we focus on the case of leakage-resilient one-way functions and discuss applications to other primitives at the end of this section.

A one-way function with domain \mathcal{X} and range \mathcal{Y} in the floppy model consists of a tuple of polynomial-time algorithms $\Pi = (\text{KGen}, \text{Sample}, \text{Eval}, \text{Update})$ specified as follows:

- The randomized key generation algorithm $\text{KGen}(1^\lambda)$ takes as input the security parameter $\lambda \in \mathbb{N}$ and outputs public parameters ρ and update key τ .
- The randomized sampling algorithm $\text{Sample}(\rho)$ takes as input the public parameters ρ and outputs a value $x \in \mathcal{X}$.
- The deterministic evaluation algorithm $\text{Eval}(\rho, x)$ takes as input the public parameters ρ and a value x , and outputs a value $y \in \mathcal{Y}$.
- The randomized update algorithm $\text{Update}(\rho, \tau, x)$ takes as input the update key τ and a value x , and outputs a value $x' \in \mathcal{X}$.

Correctness requires that for all $\lambda \in \mathbb{N}$, for all $(\rho, \tau) \in \text{KGen}(1^\lambda)$, and for all $x \in \mathcal{X}$, it holds that $\text{Eval}(\rho, \text{Update}(\rho, \tau, x)) = \text{Eval}(\rho, x)$ with probability one over the randomness of Update . As for security, we intuitively require that no computationally bounded attacker can find a valid pre-image x^* of a target value y obtained by evaluating the one-way function on a randomly chosen pre-image x , even if the attacker can obtain some leakage on x . After each leakage query, the update key is used to obtain a fresh new x which is used to answer the next leakage query. Agrawal *et al.* [ADVW13] formalized this notion in the setting of bounded leakage. Below, we extend their definition to the noisy-leakage setting.

Definition 15 (Continuously leakage-resilient one-way function). *We say that $\Pi = (\text{KGen}, \text{Sample}, \text{Update}, \text{Eval})$ is a continuously ℓ -min-entropy-noisy leakage-resilient one-way function in the floppy model (ℓ -ME-NLR-OWF, for short) with domain \mathcal{X} and range \mathcal{Y} , if for all PPT adversaries \mathbf{A} there exists a negligible function $\nu : \mathbb{N} \rightarrow [0, 1]$ such that*

$$\mathbb{P} \left[\text{Eval}(\rho, x^*) = y : \begin{array}{l} (\rho, \tau) \leftarrow_{\$} \text{KGen}(1^\lambda); x \leftarrow_{\$} \text{Sample}(\rho) \\ y = \text{Eval}(\rho, x); x^* \leftarrow_{\$} \mathbf{A}^{\text{NLeak}_{\rho, \tau}(x, \cdot)}(\rho, y) \end{array} \right] \leq \nu(\lambda),$$

where oracle $\text{NLeak}_{\rho, \tau}(x, \cdot)$ upon input (the description of) function $f : \mathcal{X} \rightarrow \{0, 1\}^*$ returns $f(x)$ and updates x to $\text{Update}(\rho, \tau, x)$, and where \mathbf{A} only queries its oracles with functions f that are ℓ -ME-noisy for the distribution $X|Y, R$ such that X is uniform over \mathcal{X} and $Y = \text{Eval}(R, X)$ for R being the distribution of the public parameters output by KGen .

Moreover, in case the attacker in the above definition is restricted to only output functions f that are ℓ -U-noisy leakage functions, we say that Π is a continuously ℓ -uniform-noisy leakage-resilient one-way function (or ℓ -U-NLR-OWF).

Let $\text{GroupGen}(1^\lambda)$ be a randomized algorithm outputting the description of a cyclic group \mathbb{G} with generator g and prime order $q = q(\lambda)$. Consider the following construction.

KGen(1^λ): Run $(\mathbb{G}, g, q) \leftarrow_{\$} \text{GroupGen}(1^\lambda)$, pick a random vector $\tau = (\tau_1, \dots, \tau_n) \leftarrow_{\$} \mathbb{Z}_q^n$ and output public parameters $\rho = (\mathbb{G}, g, g_1 = g^{\tau_1}, \dots, g_n = g^{\tau_n}, q)$ and update key τ .

Sample(ρ): Output $x \leftarrow_{\$} \mathbb{Z}_q^n$.

Eval(ρ, \mathbf{x}): Parse $\mathbf{x} = (x_1, \dots, x_n)$ and output $y = \prod_{i=1}^n g_i^{x_i}$.

Update(ρ, τ, \mathbf{x}): Pick $\sigma \leftarrow \ker(\tau)$ and output $\mathbf{x} + \sigma$.

Agrawal *et al.* [ADVW13] proved that the above construction is secure under the discrete log assumption (cf. §2.3) in the setting of bounded leakage (i.e., in case the attacker of Definition 15 only outputs ℓ -bounded leakage functions). The theorem below states that the same construction is secure in the noisy-leakage setting as well.

Theorem 16. *Assuming the DL assumption holds for GroupGen, the above cryptosystem Π is an ℓ -ME-NLR-OWF (and thus an ℓ -U-NLR-OWF) with domain \mathbb{Z}_q^n and range \mathbb{G} , so long as $\ell = (n - 3) \log(q) - \omega(\log^2(\lambda))$.*

The proof of Theorem 16 relies on a noisy variant of the subspace hiding lemma [BKKV10, LLW11, ADVW13], which was used by several previous works to construct leakage-resilient primitives. The lemma intuitively says that it is hard to distinguish (noisy) leakage on a random (affine) subspace of the space generated by a random matrix \mathbf{A} from (noisy) leakage on a uniformly random matrix \mathbf{U} , even if \mathbf{A} becomes public *after* the leakage occurs.

Lemma 7 (Dual affine subspace hiding with ME-noisy leakage). *Let $n \geq m \geq d \geq u$ be integers. Let $\mathcal{W} \subseteq \mathbb{Z}_q^n$ be a fixed subspace of dimension m , let $\mathbf{B} \in \mathbb{Z}_q^{n \times u}$ be an arbitrary matrix, and let $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ be any ℓ -ME-noisy function. For randomly sampled $\mathbf{A} \leftarrow \mathcal{W}^d$ (interpreted as an $n \times d$ matrix), $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$, $\mathbf{U} \leftarrow \mathcal{W}^u$ (interpreted as an $n \times u$ matrix), we have*

$$(f(\mathbf{AV} + \mathbf{B}), \mathbf{A}) \approx (f(\mathbf{U}), \mathbf{A})$$

so long as $(d - u) \log(q) - \ell = \omega(\log^2(\lambda))$, $n = \text{poly}(\lambda)$, and $q = \lambda^{\omega(1)}$.

Proof. The proof is by reduction to the lemma below, which is the equivalent of Lemma 7 for the case where the leakage function outputs at most ℓ bits.

Lemma 8 (Dual affine subspace hiding with bounded leakage [ADVW13]). *Let $n \geq m \geq d \geq u$ be integers. Let $\mathcal{W} \subseteq \mathbb{Z}_q^n$ be a fixed subspace of dimension m , let $\mathbf{B} \in \mathbb{Z}_q^{n \times u}$ be an arbitrary matrix, and let $f : \{0, 1\}^* \rightarrow \{0, 1\}^\ell$ be any function with ℓ -bit output. For randomly sampled $\mathbf{A} \leftarrow \mathcal{W}^d$ (interpreted as an $n \times d$ matrix), $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$, $\mathbf{U} \leftarrow \mathcal{W}^u$ (interpreted as an $n \times u$ matrix), we have*

$$(f(\mathbf{AV} + \mathbf{B}), \mathbf{A}) \approx (f(\mathbf{U}), \mathbf{A})$$

so long as $(d - u) \log(q) - \ell = \omega(\log(\lambda))$, $n = \text{poly}(\lambda)$, and $q = \lambda^{\omega(1)}$.

By setting $\alpha = 0$ (as $\mathbf{AV} + \mathbf{B}$ is uniform over an affine subspace of \mathbb{Z}_q^n), $\beta = 2 + \log^2(\lambda)$, and $\varepsilon = 2^{-\lambda}$ in Corollary 2, for every ℓ -ME-noisy leakage function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ there exists a simulator Sim_f such that for randomly sampled $\mathbf{A} \leftarrow \mathcal{W}^d$, $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$ and $\mathbf{U} \leftarrow \mathcal{W}^u$ we have

$$\begin{aligned} (f(\mathbf{AV} + \mathbf{B}), \mathbf{A}) &\approx (\text{Sim}_f^{\text{Leak}(\mathbf{AV} + \mathbf{B}, \cdot)}, \mathbf{A}) \\ (f(\mathbf{U}), \mathbf{A}) &\approx (\text{Sim}_f^{\text{Leak}(\mathbf{U}, \cdot)}, \mathbf{A}), \end{aligned}$$

where Sim_f asks to its oracle a single query consisting of a function $f' : \{0, 1\}^* \rightarrow \{0, 1\}^{\ell'}$ with $\ell' = \ell + O(\log^2(\lambda))$ bits of output, and where the symbol \approx hides a negligible statistical distance.

Next, we claim that for every ℓ -ME-noisy leakage function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$, and for randomly sampled $\mathbf{A} \leftarrow \mathcal{W}^d$, $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$ and $\mathbf{U} \leftarrow \mathcal{W}^u$ we have

$$(\text{Sim}_f^{\text{Leak}(\mathbf{AV} + \mathbf{B}, \cdot)}, \mathbf{A}) \approx (\text{Sim}_f^{\text{Leak}(\mathbf{U}, \cdot)}, \mathbf{A}).$$

By contradiction, assume that there exists an ℓ -ME-noisy leakage function $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ and an unbounded distinguisher D such that for randomly sampled $\mathbf{A} \leftarrow \mathcal{W}^d$, $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$ and $\mathbf{U} \leftarrow \mathcal{W}^u$ we have

$$\mathbb{P}\left[D(\text{Sim}_f^{\text{Leak}(\mathbf{AV} + \mathbf{B}, \cdot)}, \mathbf{A}) = 1\right] - \mathbb{P}\left[D(\text{Sim}_f^{\text{Leak}(\mathbf{U}, \cdot)}, \mathbf{A}) = 1\right] \geq 1/\text{poly}(\lambda).$$

Let $f' : \{0, 1\}^* \rightarrow \{0, 1\}^{\ell'}$ be the leakage function output by Sim_f , and denote by \tilde{Z} the output of the simulator. Consider the distinguisher D' that upon receiving $(f'(\mathbf{T}), \mathbf{A})$, where \mathbf{T} either equals $\mathbf{AV} + \mathbf{B}$ or \mathbf{U} , returns the same as D upon input (\tilde{Z}, \mathbf{A}) . Then, we have

$$\mathbb{P}\left[D'(f'(\mathbf{AV} + \mathbf{B}), \mathbf{A}) = 1\right] - \mathbb{P}\left[D'(f'(\mathbf{U}), \mathbf{A}) = 1\right] \geq 1/\text{poly}(\lambda).$$

Since $(d-u)\log(q) - \ell' = (d-u)\log(q) - \ell - O(\log^2(\lambda)) = \omega(\log^2(\lambda))$, $n = \text{poly}(\lambda)$ and $q = \lambda^{\omega(1)}$, the above contradicts Lemma 8 and thus finishes the proof. \square

Proof of Theorem 16. Let $p(\lambda) \in \text{poly}(\lambda)$ be the number of leakage queries asked by attacker A in an execution of the experiment defining security of Π . We consider a sequence of games, as described below.

Game $G_0(\lambda)$: This is the original experiment defining security of Π . Here, the challenger samples $\rho = (\mathbb{G}, g, g_1, \dots, g_n, q)$ and $\tau \leftarrow \mathbb{Z}_q^n$, picks $\mathbf{x}_1 \leftarrow \mathbb{Z}_q^n$, computes $y = \text{Eval}(\rho, \mathbf{x}_1)$ and sends y to A . Hence, upon input the i -th leakage query f_i from A , the challenger returns $f_i(\mathbf{x}_i)$ and lets $\mathbf{x}_{i+1} = \mathbf{x}_i + \sigma_i$ for random $\sigma_i \leftarrow \ker(\tau)$.

Game $G_{1,j}(\lambda)$: The challenger picks a random subspace $\mathcal{S} \subseteq \ker(\tau)$ of dimension $n - 2$ at the beginning of the experiment, and lets $\mathbf{x} = \mathbf{x}_1$. Hence, the i -th leakage query is answered as follows:

- In case $1 \leq i \leq p - j$, compute $\mathbf{x}_{i+1} = \mathbf{x} + \sigma_i$ for $\sigma_i \leftarrow \ker(\tau)$.
- In case $p - j + 1 \leq i \leq p$, compute $\mathbf{x}_{i+1} = \mathbf{x} + \sigma_i$ for $\sigma_i \leftarrow \mathcal{S}$.

Game $G_2(\lambda)$: For each leakage query $i \in [p]$, the challenger defines $\mathbf{x}_{i+1} = \mathbf{x}_i + \sigma_i$ for $\sigma_i \leftarrow \mathcal{S}$.

Notice that $G_{1,p}(\lambda)$ is identically distributed to $G_2(\lambda)$. Furthermore, $G_{1,0}(\lambda)$ is identically distributed to $G_0(\lambda)$ as in both games the vectors \mathbf{x}_i are uniformly random in the affine subspace $\{\mathbf{x}_i : g^{(\mathbf{x}_i, \tau)} = y\}$. Let E_j be the event that A wins in Game $G_{1,j}(\lambda)$ and additionally returns a vector \mathbf{x}^* such that $\mathbf{x}^* - \mathbf{x} \notin \mathcal{S}$.

Lemma 9. *For every $1 \leq j \leq p$, and all even unbounded attackers A , there exists a negligible function $\nu_{1,j} : \mathbb{N} \rightarrow [0, 1]$ such that $|\mathbb{P}[E_{j-1}] - \mathbb{P}[E_j]| \leq \nu_{1,j}(\lambda)$.*

Proof. Fix any $j \in [p]$. By contradiction, assume that there is a possibly unbounded attacker A such that $|\mathbb{P}[E_{j-1}] - \mathbb{P}[E_j]| \geq 1/\text{poly}(\lambda)$. Consider the following unbounded distinguisher D breaking the affine version of the dual subspace hiding lemma with ME-noisy leakage (cf. Lemma 7 with $\mathcal{W} = \ker(\tau)$, $\mathbf{B} = \mathbf{x}$, $\mathbf{A} = \mathcal{S}$ and parameters $d = n - 2$ and $u = 1$):

- The distinguisher picks $(\rho, \tau) \leftarrow \text{KGen}(1^\lambda)$, $\mathbf{x} \leftarrow \mathbb{Z}_q^n$, and runs $A(\rho, y)$ where $y = \prod_{i=1}^n g_i^{x_i}$.
- Upon input a leakage query f_i from A , answers as follows:
 - In case $1 \leq i \leq p - j - 1$, return $f_i(\mathbf{x}_i)$ to A and define $\mathbf{x}_{i+1} = \mathbf{x} + \sigma_i$ for randomly chosen $\sigma_i \leftarrow \ker(\tau)$.
 - In case $i = p - j + 1$, forward the leakage function f_{p-j+1} to the challenger obtaining (a description of) the subspace \mathcal{S} and a string $z \in \{0, 1\}^*$, and return z to A .

- In case $p - j + 2 \leq i \leq p$, return $f_i(\mathbf{x}_i)$ to \mathbf{A} and define $\mathbf{x}_{i+1} = \mathbf{x} + \boldsymbol{\sigma}_i$ for randomly chosen $\boldsymbol{\sigma}_i \leftarrow_{\mathcal{S}} \mathcal{S}$.

- Upon receiving \mathbf{x}^* from \mathbf{A} , output 1 if and only if event E_j happens.

We note that the simulation is perfect, in the sense that if the leakage is applied to $\mathbf{u} = \mathbf{x} + \hat{\boldsymbol{\sigma}}$ for $\hat{\boldsymbol{\sigma}} \leftarrow_{\mathcal{S}} \ker(\boldsymbol{\tau})$ the view of \mathbf{A} is identical to that in a run of game $G_{1,j}(\lambda)$, whereas if the leakage is applied to $\mathbf{x} + \hat{\boldsymbol{\sigma}}$ for $\hat{\boldsymbol{\sigma}} \leftarrow_{\mathcal{S}} \mathcal{S}$ the view of \mathbf{A} is identical to that in a run of game $G_{1,j-1}(\lambda)$. Since \mathbf{A} outputs an ℓ -ME-noisy leakage function for ℓ as in the statement of the theorem, we get that $(n - 3) \log(q) - \ell = \omega(\log^2(\lambda))$ which contradicts Lemma 7. The lemma follows. \square

Lemma 10. *Under the DL assumption, for all PPT attackers \mathbf{A} there is a negligible function $\nu_2 : \mathbb{N} \rightarrow [0, 1]$ such that $\mathbb{P}[E_p] \leq \nu_2(\lambda)$.*

Proof. The proof of the lemma is identical to [ADVW13, Claim 12] and therefore omitted. The only difference is that the attacker in our case queries the leakage oracle with ℓ -ME-noisy leakage functions (instead of leaking at most ℓ bits in each leakage round).

However, this is not a problem as the reduction can sample \mathbf{x} and moreover all the leakage queries are answered using a random $(n - 2)$ -dimensional subspace $\mathcal{S} \subseteq \ker(\boldsymbol{\tau})$. \square

By combining the above lemmas, for all PPT attackers \mathbf{A} there is a negligible function $\nu : \mathbb{N} \rightarrow [0, 1]$ such that

$$\mathbb{P}[G_0(\lambda) = 1] \leq \mathbb{P}[E_0] + 1/q \leq \mathbb{P}[E_p] + \sum_{j=1}^p |\mathbb{P}[E_{j-1}] - \mathbb{P}[E_j]| + 1/q \leq \nu(\lambda),$$

where the first inequality follows because \mathbf{A} has no information about \mathcal{S} in game $G_0(\lambda)$. This finishes the proof. \square

Other primitives. The subspace hiding lemma can be used to construct other bounded leakage-resilient primitives including: public-key encryption [BKKV10, LLW11, DLWW11, ADVW13, DFMV13, DFMV17], leakage-resilient storage [DLWW11], identity-based encryption [BKKV10], digital signatures [BKKV10, LLW11], traitor tracing [ADVW13], and non-malleable codes with split-state refresh [FNSV18]. We believe our techniques can be used to prove these constructions are in fact resilient to ME-noisy leakage.

Some of the above results rely on a dual version of the lemma which intuitively says it is hard to distinguish random vectors from the column span of a random matrix \mathbf{A} from uniformly random and independent vectors. Thanks to our result, we can directly translate this lemma in the noisy leakage setting as stated below.

Lemma 11 (Subspace hiding with ME-noisy leakage). *Let $n \geq d \geq u$, s be integers, $\mathbf{S} \in \mathbb{Z}_q^{d \times s}$ be an arbitrary (fixed and public) matrix and $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ be an arbitrary ℓ -ME-noisy leakage function. For randomly sampled $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times d}$, $\mathbf{V} \leftarrow \mathbb{Z}_q^{d \times u}$, $\mathbf{U} \leftarrow \mathbb{Z}_q^{n \times u}$, we have*

$$(f(\mathbf{A}), \mathbf{AS}, \mathbf{V}, \mathbf{AV}) \approx (f(\mathbf{A}), \mathbf{AS}, \mathbf{V}, \mathbf{U})$$

so long as $(d - s - u) \log(q) - \ell = \omega(\log^2(\lambda))$ and $n = \text{poly}(\lambda)$.

6.3 Bounded-Collusion Protocols

In this section, we deal with applications related to so-called bounded-collusion protocols (BCPs). These are interactive protocols between n parties where at each round a subset of $k < n$ parties are selected, and the output of a leakage function applied to the input of such parties is appended to the protocol's transcript.

Definition 16 (Bounded-communication BCPs). *An interactive (possibly randomized) protocol π is called an n -party r -round ℓ -bounded communication k -bounded-collusion protocol $((n, r, \ell, k)$ -BC-BCP, for short) if:*

- (i) *the n parties start the protocol with input $x_1, \dots, x_n \in \mathcal{X}$, and the transcript τ is empty at the beginning of the protocol;*
- (ii) *there is a function $\text{Next} : \{0, 1\}^* \rightarrow \binom{[n]}{k}$ taking as input a (partial) transcript τ and outputting a set $\mathcal{S} \subset [n]$ with $|\mathcal{S}| = k$ along with a function $f : \mathcal{X}^k \rightarrow \{0, 1\}^*$;*
- (iii) *at each round $j \in [r]$ with current transcript τ , the protocol runs $\text{Next}(\tau)$ obtaining (\mathcal{S}_j, f_j) and appends the message $f_j(X_{\mathcal{S}_j})$ to the current transcript τ ;*
- (iv) *the final transcript τ consists of at most $\ell \in \mathbb{N}$ bits.*

The above notion, which was introduced by Kumar, Meka, and Sahai [KMS19], interpolates nicely between the well-known *number-in-hand* (NIH) and *number-on-forehead* (NOF) models, which correspond respectively to the extreme cases $k = 1$ and $k = n - 1$. Note that the number of rounds in a BC-BCP is at most $r \leq \ell$.

Below, we generalize the definition of BCPs to settings where the round functions correspond to noisy-leakage (in particular, dense and uniform-noisy leakage) functions on the parties' inputs, and thus there is no restriction on the size of the final transcript.

Definition 17 (Dense-communication BCPs). *An interactive (possibly randomized) protocol π is called an n -party r -round (p, γ, δ) -dense communication k -bounded-collusion protocol $((n, r, p, \gamma, \delta, k)$ -DC-BCP, for short) if it satisfies the same properties as in Definition 16, except that property (iv) is replaced by*

- (iv') *for each $j \in [r]$, the function $f_j : \mathcal{X}^k \rightarrow \{0, 1\}^*$ is (p, γ, δ_j) -dense leakage for $X_{\mathcal{S}_j} | \tau_{j-1}$, where τ_j denotes the transcript up to the j -th round and $0 < \delta_j \leq 1$, and where additionally $\prod_{j=1}^r \delta_j \geq \delta$.*

Definition 18 (Noisy-communication BCPs). *An interactive (possibly randomized) protocol π is called an n -party r -round ℓ -noisy communication k -bounded-collusion protocol $((n, r, \ell, k)$ -NC-BCP, for short) if it satisfies the same properties as in Definition 16, except that property (iv) is replaced by*

- (iv'') *for each $j \in [r]$, the function $f_j : \mathcal{X}^k \rightarrow \{0, 1\}^*$ is ℓ_j - U -noisy leakage for $X_{\mathcal{S}_j}$, where $\ell_j \geq 0$ and additionally $\left\lceil \sum_{j=1}^r \ell_j \right\rceil \leq \ell$.*

Observe that the number of rounds in a DC-BCP or NC-BCP is unbounded. Also, note that property (iv'') in Definition 18 implicitly implies that the overall leakage drops the min-entropy of the uniform distribution over any subset of k inputs by at most ℓ . More formally, the final transcript τ is such that¹³ for all subsets $\mathcal{S} \in \binom{[n]}{k}$ we have

$$\tilde{\mathbb{H}}_\infty(U_{\mathcal{S}} | \pi(U_1, \dots, U_n)) \geq \mathbb{H}_\infty(U_{\mathcal{S}}) - \ell, \quad (17)$$

¹³This is because, by [DHLW10, Lemma L.3], any sequence of (adaptively chosen) functions f_1, \dots, f_r on a random variable X , such that each function f_j is ℓ_j -ME-noisy leakage for some $\ell_j \geq 0$ and where $\sum_{j=1}^r \ell_j \leq \ell$, satisfies $\tilde{\mathbb{H}}_\infty(X | f_1(X), \dots, f_r(X)) \geq \mathbb{H}_\infty(X) - \ell$. Furthermore, for the case of NC-BCPs, in the worst case all the leakage happens on the same subset \mathcal{S} of inputs.

where $U = (U_1, \dots, U_n)$ is uniform over \mathcal{X}^n and $\pi(U_1, \dots, U_n)$ denotes the distribution of the transcript τ at the end of the protocol.

Clearly, any BC-BCP is also a NC-BCP with the same leakage parameter. Below, we show that the converse is also true, in the sense that the transcript of any NC-BCP π can be simulated using the transcript of a related BC-BCP π' , up to a small statistical distance. In fact, the latter statement holds true for the more general case of DC-BCPs.

Theorem 17. *Let π be an $(n, r, p, \gamma, \delta, k)$ -DC-BCP. There exists an (n, r, ℓ', k) -BC-BCP π' such that, for any $\varepsilon > 0$, a transcript of π can be simulated within statistical distance $r \cdot (\varepsilon + \varepsilon^{1/4} + \gamma + p)$ given a transcript of π' with length $\ell' = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$.*

Proof. We start by describing protocol π' acting on a random variable $X = (X_1, \dots, X_n)$. Consider the simulator Sim_f guaranteed by Theorem 3.

- Let τ' be initially empty, and sample r independent random tapes ρ_1, \dots, ρ_r for Sim .
- At each round $j \in [r]$, the function Next' takes as input the current transcript $\tau' = z'_1 || \dots || z'_{j-1}$ and runs $\text{Next}(\tilde{\tau})$, where $\tilde{\tau} = \text{Sim}_{f_1}(z'_1; \rho_1) || \dots || \text{Sim}_{f_{j-1}}(z'_{j-1}; \rho_{j-1})$.
- Let (f_j, \mathcal{S}_j) be the j -th output of Next . Then, Next' runs Sim_{f_j} on $X_{\mathcal{S}_j} | \tilde{\tau}$ (with fixed random tape ρ_j), obtaining a leakage function $f'_j : \mathcal{X}^k \rightarrow \{0, 1\}^{\ell'_j}$, and outputs (f'_j, \mathcal{S}_j) .

Next, we claim that protocol π' has ℓ' -bounded communication for ℓ' as in the statement of the theorem. Recall that, for each $j \in [r]$, the function f_j output by Next is (p, γ, δ_j) -dense leakage for $X_{\mathcal{S}_j} | \tilde{\tau}$, with $0 < \delta_j \leq 1$. Then, by applying Theorem 3, for any $\varepsilon > 0$ we get that $\ell'_j = \log(1/\delta_j) + \log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2$. Hence, the final transcript τ' has size at most $\ell' = \sum_{j=1}^r \ell'_j = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$, which is the bound in the statement of the theorem.

It remains to prove that we can simulate a transcript of π given a transcript of π' . Consider the simulator that, after running π' with random tapes ρ_1, \dots, ρ_r , obtains the transcript $\tau' = z'_1 || \dots || z'_r$ and simply outputs the simulated transcript $\tilde{\tau} = \text{Sim}_{f_1}(z'_1; \rho_1) || \dots || \text{Sim}_{f_r}(z'_r; \rho_r)$. By a hybrid argument, Theorem 3 implies that the transcript $\tilde{\tau}$ is within statistical distance at most $r \cdot (\varepsilon + \gamma + p) + \sum_{j=1}^r \varepsilon^{1/4\delta_j} \leq r \cdot (\varepsilon + \varepsilon^{1/4} + \gamma + \delta)$ from the transcript τ obtained by running π . This finishes the proof. \square

Theorem 18. *Let π be an (n, r, ℓ, k) -NC-BCP. There exists an (n, r, ℓ', k) -BC-BCP π' such that, for any $0 < \delta < 1$, a transcript of π can be simulated within statistical distance $r \cdot 3\delta$ given a transcript of π' with length $\ell' \leq \ell + r \cdot (6 + 2 \log(1/\delta) + \log \log(1/\delta))$.*

Proof. We use the same protocol π' as in the proof of Theorem 17, with the only difference that at each round $j \in [r]$ we now run the simulator Sim_{f_j} guaranteed by Corollary 3 on $X_{\mathcal{S}_j} | \tilde{\tau}$.

Then, we claim that protocol π' has ℓ' -bounded communication for ℓ' as in the statement of the theorem. Recall that, for each $j \in [r]$, the function f_j output by Next is ℓ_j -U-noisy for $X_{\mathcal{S}_j}$, with $\ell_j \geq 0$ and $\ell \geq \left\lceil \sum_{j=1}^r \ell_j \right\rceil$. By Theorem 6, we conclude that f_j is ℓ_j -U-noisy for $X_{\mathcal{S}_j} | \tilde{\tau}$ as well. Then, by setting $\beta = 2 + 2 \log(1/\delta)$ in Corollary 3, for any $0 < \delta < 1$, we get that $\ell'_j = \ell_j + 4 + 2 \log(2/(2 - \delta)) + 2 \log(1/\delta) + \log \log(1/\delta)$. Hence, the final transcript τ' has size at most $\ell' = \sum_{j=1}^r \ell'_j \leq \ell + r \cdot (4 + 2 \log(2/(2 - \delta)) + 2 \log(1/\delta) + \log \log(1/\delta))$, which implies the bound in the statement of the theorem.

Finally, by applying the same hybrid argument as in the proof of Theorem 17, we obtain that the transcript $\tilde{\tau}$ is within statistical distance at most $r \cdot 3\delta$ from the transcript τ obtained by running π . This finishes the proof. \square

Next, we show that Theorem 17 and Theorem 18 have applications to communication complexity lower bounds, and to constructing cryptographic primitives with adaptive noisy-leakage resilience (i.e., where leakage resilience is modeled either as a NC-BCP or as a DC-BCP).

6.3.1 Communication complexity lower bounds

We say that an (n, r, ℓ, k) -BCP π (with either bounded or noisy communication) ε -computes a (deterministic) Boolean function $\phi : \mathcal{X}^n \rightarrow \{0, 1\}$, if there exists an unbounded predictor P that, after running a BCP protocol π on the parties' inputs yielding a final transcript τ , outputs $\phi(X_1, \dots, X_n)$ with probability at least $1/2 + \varepsilon$ (over the randomness of $(X_i)_{i \in [n]}$, π and P). The theorem below says that for any NC-BCP π that computes a Boolean function ϕ there is a BC-BCP π' that computes the same function with roughly the same probability, where the size ℓ' of a transcript of π' is related to the leakage parameter ℓ of π .

Corollary 7. *Let π be any (n, r, ℓ, k) -NC-BCP that ε -computes a Boolean function ϕ . Then, there exists an (n, r, ℓ', k) -BC-BCP π' that ε' -computes ϕ so long as*

$$\ell' \leq \ell + r \cdot (6 + 2 \log(6r/\varepsilon) + \log \log(6r/\varepsilon)) \quad \text{and} \quad \varepsilon' = \varepsilon/2.$$

Proof. Consider the predictor P' that simply runs P upon input the transcript $\tilde{\tau}$ obtained using the protocol π' from Theorem 18. By setting $\delta = \varepsilon/(6r)$, it follows that the transcript of π' has size at most $\ell' \leq \ell + r \cdot (6 + 2 \log(6r/\varepsilon) + \log \log(6r/\varepsilon))$ and furthermore the simulated transcript $\tilde{\tau}$ of P is up to statistical distance at most $r \cdot \varepsilon/(2r) = \varepsilon/2$ from the transcript τ obtained by running π . Thus, P' computes ϕ with probability at least $1/2 + \varepsilon - \varepsilon/2 = 1/2 + \varepsilon/2$. \square

The above corollary can be used to translate known lower bounds in communication complexity for BC-BCPs to the more general setting of NC-BCPs.¹⁴ Note that a lower bound on the communication complexity of BC-BCPs does not necessarily imply a lower bound on the noisiness of NC-BCPs, as the fact that the transcript must consist of *at least* ℓ bits does not say anything about how each round function reveals on the players' inputs.

Remark 7 (On lower bounds on the leakage parameters of NC-BCPs). It may seem that a lower bound on the parameter ℓ of NC-BCPs does not necessarily mean that any protocol must reveal a lot of information on the parties' inputs, as the actual min-entropy drop in Eq. (17) could be much smaller¹⁵ than ℓ . Nevertheless, we observe that the definition of NC-BCP implies that there must exist an index $j^* \in [r]$ such that, say, $\ell_{j^*} \geq \frac{\ell-1}{r}$. This is because, if $\ell_j < \frac{\ell-1}{r}$ for all $j \in [r]$, then $\lceil \sum_{j=1}^r \ell_j \rceil \leq \ell - 1$. In this light, the corollaries below still say that, for certain Boolean functions, a transcript must necessarily reveal enough information about the inputs so long as the number of rounds is not too large.

Corollary 8. *Let $\phi_{\text{gip}} : (\{0, 1\}^m)^n \rightarrow \{0, 1\}$ be the generalized inner product function such that $\phi_{\text{gip}}(x_1, \dots, x_n) = 1$ if and only if the number of positions where all the x_i 's have 1 is odd. Then, every $(n, r, \ell, n-1)$ -NC-BCP that ε -computes ϕ_{gip} must satisfy $\ell + r \log(r) = \Omega(m/(2^n \cdot \log(2/\varepsilon)))$.*

Proof. Chung [Chu90] proved that every $(n, r, \ell', n-1)$ -BC-BCP that ε' -computes ϕ_{gip} must satisfy $\ell' = \Omega(m/2^n + \log(\varepsilon'))$. However, by Corollary 7, every $(n, r, \ell, n-1)$ -NC-BCP that ε -computes ϕ_{gip} can be transformed into an $(n, r, \ell', n-1)$ -BC-BCP that ε' -computes ϕ_{gip} with $\ell' \leq \ell + r \cdot (6 + 3 \log(6r/\varepsilon)) \leq 17 \cdot (\ell + r \log(r)) \cdot \log(2/\varepsilon)$ and $\varepsilon' = \varepsilon/2$. Thus, we get that $(\ell + r \log(r)) \cdot \log(2/\varepsilon) = \Omega(m/2^n + \log(\varepsilon/2))$. The statement follows. \square

¹⁴In fact, using Theorem 17, we could also derive lower bounds on DC-BCPs. However, we stick to the setting of NC-BCPs for simplicity.

¹⁵For instance, take $k = 1$ and consider the functions f_1, \dots, f_n that always reveal the first bit of X_1 . Then, $\ell = \sum_{j=1}^n \ell_j = n$, but $\mathbb{H}_\infty(U_1 | \pi(U_1, \dots, U_n)) = \mathbb{H}_\infty(U_1) - 1$.

Corollary 9. Let $\phi_{\text{qr}} : (\{0, 1\}^m)^n \rightarrow \{0, 1\}$ be the function such that $\phi_{\text{qr}}(x_1, \dots, x_n) = 1$ if and only if $\sum_{i=1}^n x_i$ is a quadratic residue mod q , where q is an m -bit prime number. Then, every $(n, r, \ell, n-1)$ -NC-BCP that ε -computes ϕ_{qr} must satisfy $\ell + r \log(r) = \Omega(m/(2^n \cdot \log(2/\varepsilon)))$.

Proof. Babai, Nisant and Szegedy [BNS92] proved that every $(n, r, \ell', n-1)$ -BC-BCP that ε' -computes ϕ_{qr} must satisfy $\ell' = \Omega(m/2^n + \log(\varepsilon'))$. However, by Corollary 7, every $(n, r, \ell, n-1)$ -NC-BCP that ε -computes ϕ_{qr} can be transformed into an $(n, r, \ell', n-1)$ -BC-BCP that ε' -computes ϕ_{qr} with $\ell' \leq \ell + r \cdot (6 + 3 \log(6r/\varepsilon)) \leq 17 \cdot (\ell + r \log(r)) \cdot \log(2/\varepsilon)$ and $\varepsilon' = \varepsilon/2$. Thus, we get that $(\ell + r \log(r)) \cdot \log(2/\varepsilon) = \Omega(m/2^n + \log(\varepsilon/2))$. The statement follows. \square

Corollary 10. There exists an explicit function $\phi : (\{0, 1\}^m)^n \rightarrow \{0, 1\}$ such that every (n, r, ℓ, k) -NC-BCP that ε -computes ϕ must satisfy

$$\ell + r \log(r) = \Omega\left(\frac{1}{\log(2/\varepsilon)} \cdot (m + \log(\varepsilon/2))^{\frac{\log(n/k)}{1/\log(n/k)+1}}\right).$$

Proof. Kumar, Meka, and Zuckerman [KMZ20], as well as Chattopadhyay, Goodman, Goyal and Li [CGGL20], independently proved that there exists a Boolean function ϕ (based on the Bourgain extractor [Bou05]) such that every (n, r, ℓ', k) -BC-BCP that ε' -computes ϕ must satisfy $\ell' = \Omega((m + \log(\varepsilon'))^{\frac{\log(n/k)}{\log(n/k)+1}})$. However, by Corollary 7, every (n, r, ℓ, k) -NC-BCP that ε -computes ϕ can be transformed into an (n, r, ℓ', k) -BC-BCP that ε' -computes ϕ with $\ell' \leq \ell + r \cdot (6 + 3 \log(6r/\varepsilon)) \leq 17 \cdot (\ell + r \log(r)) \cdot \log(2/\varepsilon)$ and $\varepsilon' = \varepsilon/2$. Thus, we get that $(\ell + r \log(r)) \cdot \log(2/\varepsilon) = \Omega((m + \log(\varepsilon/2))^{\frac{\log(n/k)}{\log(n/k)+1}})$. The statement follows. \square

To get an idea of the parameters, the above corollaries imply that whenever $n = O(\log \lambda)$, $\varepsilon = 2 \cdot 2^{-\lambda}$ and $r = O(\ell)$, any BCP with r rounds that computes either ϕ_{gip} or ϕ_{qr} with probability better than $2 \cdot 2^{-\lambda}$ must reduce the (average) min-entropy of the (uniform) distribution on the parties' inputs by at least $\Omega(\sqrt{m}/(\lambda \log(\lambda)))$ bits. Similarly, whenever $k \leq c \cdot n$ for some constant $c < 1$, there exists an explicit function ϕ such that any BCP with r rounds that computes ϕ with probability better than $2 \cdot 2^{-\lambda}$ must reduce the (average) min-entropy of the (uniform) distribution on the parties' inputs by at least $(m/\lambda)^{\Omega(1)}$.

6.3.2 BCP Leakage Resilience

Finally, we show how to lift bounded-leakage resilience to dense-leakage and uniform-noisy-leakage resilience in applications where the leakage itself is modelled as a BCP protocol. For concreteness, we focus again on secret sharing schemes and refer the reader to §A.2 and §A.3 of the appendix for additional examples (in particular, extractors for cylinder-intersection sources [KMS19] and symmetric non-interactive key exchange [LMQW20]).

Let $\Sigma = (\text{Share}, \text{Rec})$ be a secret sharing scheme as defined in §6.1. The definition below captures security of Σ in the presence of an adversary leaking information jointly from subsets of the shares of size $k < n$, where both the leakage functions and the subsets of shares are chosen adaptively. For simplicity, we focus on threshold secret sharing but our treatment can be generalized to arbitrary access structures.

Definition 19 (Secret sharing with BCP leakage resilience). Let $t, n, \ell \in \mathbb{N}, \varepsilon \in [0, 1]$ be parameters. A t -out-of- n secret sharing scheme $(\text{Share}, \text{Rec})$ is a k -joint r -adaptive (p, γ, δ) -dense ε -leakage-resilient secret sharing scheme, $(k, r, p, \gamma, \delta, \varepsilon)$ -JA-DLR-SS for short, if for all messages $y_0, y_1 \in \{0, 1\}^m$ and all $(n, r, p, \gamma, \delta, k)$ -DC-BCP π we have

$$\pi(X_1^{(0)}, \dots, X_n^{(0)}) \approx_\varepsilon \pi(X_1^{(1)}, \dots, X_n^{(1)}),$$

where $(X_1^{(b)}, \dots, X_n^{(b)}) = \text{Share}(y_b)$ is the distribution of the shares of message $y_b \in \{0, 1\}^m$ for all $b \in \{0, 1\}$.

Moreover, in case the protocol π in the above definition is an:

- (n, r, ℓ, k) -NC-BCP, we say that Σ is k -joint r -adaptive ℓ -noisy ε -leakage-resilient (or $(k, r, \ell, \varepsilon)$ -JA-NLR-SS);
- (n, r, ℓ, k) -BC-BCP, we say that Σ is k -joint r -adaptive ℓ -bounded ε -leakage-resilient (or $(k, r, \ell, \varepsilon)$ -JA-BLR-SS).

Corollary 11. *Every $(k, r, \ell, \varepsilon)$ -JA-BLR-SS scheme Σ is also a $(k, r, p, \gamma, \delta, \varepsilon')$ -JA-DLR-SS so long as $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$ and $\varepsilon' = \varepsilon + 2r \cdot (\varepsilon + \varepsilon^{1/4} + \gamma + p)$.*

Proof. For all $b \in \{0, 1\}$, by applying Theorem 17, we can replace the transcript $\pi(X_1^{(b)}, \dots, X_n^{(b)})$ of any $(n, r, p, \gamma, \delta, k)$ -DC-BCP with a simulated transcript $\tilde{\pi}(x_1^{(b)}, \dots, X_n^{(b)})$ obtained using the transcript $\pi'(x_1^{(b)}, \dots, X_n^{(b)})$ of a (n, r, ℓ, k) -BC-BCP with $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$. Hence, we have

$$\begin{aligned} \pi(X_1^{(0)}, \dots, X_n^{(0)}) &\approx_{\tilde{\varepsilon}} \tilde{\pi}(X_1^{(0)}, \dots, X_n^{(0)}) \\ &\approx_{\varepsilon} \tilde{\pi}(X_1^{(1)}, \dots, X_n^{(1)}) \\ &\approx_{\tilde{\varepsilon}} \pi(X_1^{(1)}, \dots, X_n^{(1)}), \end{aligned}$$

where $\tilde{\varepsilon} = \varepsilon + \varepsilon^{1/4} + \gamma + p$ and the second equation follows by the fact that Σ is a $(k, r, p, \gamma, \delta, \varepsilon)$ -JA-DLR-SS scheme. The statement now follows by the triangle inequality. \square

Corollary 12. *Every $(k, r, \ell', \varepsilon')$ -JA-BLR-SS scheme Σ is also a $(k, r, \ell, \varepsilon)$ -JA-NLR-SS scheme so long as $\ell' = \ell + r \cdot O(\log(r/\varepsilon))$ and $\varepsilon = 3\varepsilon'$.*

Proof. For all $b \in \{0, 1\}$, by setting $\delta = \varepsilon'/(3r)$ in Theorem 18, we can replace the transcript $\pi(X_1^{(b)}, \dots, X_n^{(b)})$ of any (n, r, ℓ, k) -NC-BCP with a simulated transcript $\tilde{\pi}(X_1^{(b)}, \dots, X_n^{(b)})$ obtained using the transcript $\pi'(X_1^{(b)}, \dots, X_n^{(b)})$ of a (n, r, ℓ', k) -BC-BCP with $\ell' \leq \ell + r \cdot (6 + 2 \log(3r/\varepsilon') + \log \log(3r/\varepsilon')) = \ell + r \cdot (O(\log(r/\varepsilon)))$. Hence, we have

$$\begin{aligned} \pi(X_1^{(0)}, \dots, X_n^{(0)}) &\approx_{\varepsilon'} \tilde{\pi}(X_1^{(0)}, \dots, X_n^{(0)}) \\ &\approx_{\varepsilon'} \tilde{\pi}(X_1^{(1)}, \dots, X_n^{(1)}) \\ &\approx_{\varepsilon'} \pi(X_1^{(1)}, \dots, X_n^{(1)}), \end{aligned}$$

where the second equation follows by the fact that Σ is a $(k, r, \ell', \varepsilon')$ -JA-BLR-SS scheme. The statement now follows by the triangle inequality. \square

Explicit constructions of secret sharing schemes with BCP leakage resilience in the bounded leakage setting can be built for any leakage bound ℓ and any $\varepsilon > 0$ from n -party functions with large NOF complexity with collusion bound $k = O(\log(n))$ [KMS19] (for arbitrary access structures) and $k = O(t/\log(t))$ [KMZ20] (for threshold access structures). By the above corollaries, these schemes are also directly secure in the settings of dense and U-noisy leakage.

7 Conclusions and Open Problems

We have shown that a single query of *noisy* leakage can be simulated in the information-theoretic setting using a single query of *bounded* leakage, up to a small statistical distance and at the price of a slight loss in the leakage parameter. The latter holds true for a fairly general class of noisy leakage (which we introduce) dubbed *dense* leakage. Importantly, dense leakage captures many already existing noisy-leakage models including those where the noisiness of the leakage is measured using the conditional average min-entropy [NS09, DHLW10, NS12], the statistical distance [DDF14, DDF19], or the mutual information [PGMP19]. For some of these models, our simulation theorems require additional assumptions on the input distribution or only hold for certain range of parameters, but in each case we show this is the best one can hope for.

The above result has applications to leakage-resilient cryptography, where we can reduce noisy-leakage resilience to bounded-leakage resilience in a black-box way. Interestingly, for some applications, the latter holds true even in the *computational* setting. Additionally, we have shown that our simulation theorems yield new lower bounds in communication complexity.

Several interesting open questions remain. We list some of them below:

- Can we prove that other families of noisy leakage (e.g., hard-to-invert leakage [DKL09]) fall within the class of dense leakage (or directly admit simulation theorems with good parameters from bounded leakage)?
- Can we make the simulator efficient for certain families of noisy leakage? The latter would allow to lift bounded-leakage resilience to noisy-leakage resilience for *all* computationally-secure applications, and for statistically-secure applications with simulation-based security in which the running time of the simulator needs to be polynomial in the running time of the adversary (such as leakage-tolerant MPC [BCH12]).
- Can we generalize Theorem 18 to a more general setting where the leakage parameter ℓ of NC-BCPs measures the worst-case average min-entropy drop w.r.t. the final transcript of the protocol (instead of being the summation over the worst-case min-entropy drops of each round function in isolation)? The latter would allow to strengthen the lower bounds in §6.3.1, as well as the security of the applications in §6.3.2, §A.2 and §A.3.

Acknowledgments

We thank François-Xavier Standaert for bringing [DFS15a, FH15] to our attention.

References

- [AARR03] Dakshi Agrawal, Bruce Archambeault, Josyula R. Rao, and Pankaj Rohatgi. The EM side-channel(s). In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *CHES 2002*, volume 2523 of *LNCS*, pages 29–45. Springer, Heidelberg, August 2003.
- [ADKO15] Divesh Aggarwal, Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski. Leakage-resilient non-malleable codes. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 398–426. Springer, Heidelberg, March 2015.

- [ADN⁺19] Divesh Aggarwal, Ivan Damgård, Jesper Buus Nielsen, Maciej Obremski, Erick Purwanto, João Ribeiro, and Mark Simkin. Stronger leakage-resilient and non-malleable secret sharing schemes for general access structures. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 510–539. Springer, Heidelberg, August 2019.
- [ADVW13] Shweta Agrawal, Yevgeniy Dodis, Vinod Vaikuntanathan, and Daniel Wichs. On continual leakage of discrete log representations. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part II*, volume 8270 of *LNCS*, pages 401–420. Springer, Heidelberg, December 2013.
- [ADW09] Joël Alwen, Yevgeniy Dodis, and Daniel Wichs. Leakage-resilient public-key cryptography in the bounded-retrieval model. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 36–54. Springer, Heidelberg, August 2009.
- [AJS17] Prabhanjan Ananth, Abhishek Jain, and Amit Sahai. Indistinguishability obfuscation for turing machines: Constant overhead and amortization. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part II*, volume 10402 of *LNCS*, pages 252–279. Springer, Heidelberg, August 2017.
- [AKO18] Divesh Aggarwal, Tomasz Kazana, and Maciej Obremski. Leakage-resilient algebraic manipulation detection codes with optimal parameters. In *IEEE International Symposium on Information Theory*, pages 1131–1135, 2018.
- [AS14] Hadi Ahmadi and Reihaneh Safavi-Naini. Detection of algebraic manipulation in the presence of leakage. In Carles Padró, editor, *ICITS 13*, volume 8317 of *LNCS*, pages 238–258. Springer, Heidelberg, 2014.
- [BCH12] Nir Bitansky, Ran Canetti, and Shai Halevi. Leakage-tolerant interactive protocols. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 266–284. Springer, Heidelberg, March 2012.
- [BDIR18] Fabrice Benhamouda, Akshay Degwekar, Yuval Ishai, and Tal Rabin. On the local leakage resilience of linear secret sharing schemes. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part I*, volume 10991 of *LNCS*, pages 531–561. Springer, Heidelberg, August 2018.
- [BFO⁺20] Gianluca Brian, Antonio Faonio, Maciej Obremski, Mark Simkin, and Daniele Venturi. Non-malleable secret sharing against bounded joint-tampering attacks in the plain model. Cryptology ePrint Archive, Report 2020/725, 2020. <https://eprint.iacr.org/2020/725>.
- [BKKV10] Zvika Brakerski, Yael Tauman Kalai, Jonathan Katz, and Vinod Vaikuntanathan. Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage. In *51st FOCS*, pages 501–510. IEEE Computer Society Press, October 2010.
- [BNS92] László Babai, Noam Nisan, and Mario Szegedy. Multiparty protocols, pseudorandom generators for logspace, and time-space trade-offs. *J. Comput. Syst. Sci.*, 45(2):204–232, 1992.
- [Bou05] Jean Bourgain. More on the sum-product phenomenon in prime fields and its applications. *International Journal of Number Theory*, 1(1):1–32, 2005.

- [Can00] Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067, 2000. <http://eprint.iacr.org/2000/067>.
- [CFL83] Ashok K. Chandra, Merrick L. Furst, and Richard J. Lipton. Multi-party protocols. In *15th ACM STOC*, pages 94–99. ACM Press, April 1983.
- [CFV19] Sandro Coretti, Antonio Faonio, and Daniele Venturi. Rate-optimizing compilers for continuously non-malleable codes. In Robert H. Deng, Valérie Gauthier-Umaña, Martín Ochoa, and Moti Yung, editors, *ACNS 19*, volume 11464 of *LNCS*, pages 3–23. Springer, Heidelberg, June 2019.
- [CGGL20] Eshan Chattopadhyay, Jesse Goodman, Vipul Goyal, and Xin Li. Leakage-resilient extractors and secret-sharing against bounded collusion protocols. Cryptology ePrint Archive, Report 2020/478, 2020. <https://eprint.iacr.org/2020/478>.
- [Chu90] Fan R. K. Chung. Quasi-random classes of hypergraphs. *Random Struct. Algorithms*, 1(4):363–382, 1990.
- [CJRR99] Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards sound approaches to counteract power-analysis attacks. In Michael J. Wiener, editor, *CRYPTO’99*, volume 1666 of *LNCS*, pages 398–412. Springer, Heidelberg, August 1999.
- [DDF14] Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 423–440. Springer, Heidelberg, May 2014.
- [DDF19] Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. *Journal of Cryptology*, 32(1):151–177, January 2019.
- [DDV10] Francesco Davì, Stefan Dziembowski, and Daniele Venturi. Leakage-resilient storage. In Juan A. Garay and Roberto De Prisco, editors, *SCN 10*, volume 6280 of *LNCS*, pages 121–137. Springer, Heidelberg, September 2010.
- [DFMV13] Ivan Damgård, Sebastian Faust, Pratyay Mukherjee, and Daniele Venturi. Bounded tamper resilience: How to go beyond the algebraic barrier. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part II*, volume 8270 of *LNCS*, pages 140–160. Springer, Heidelberg, December 2013.
- [DFMV17] Ivan Damgård, Sebastian Faust, Pratyay Mukherjee, and Daniele Venturi. Bounded tamper resilience: How to go beyond the algebraic barrier. *Journal of Cryptology*, 30(1):152–190, January 2017.
- [DFS15a] Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making masking security proofs concrete. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015*, pages 401–429, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.
- [DFS15b] Stefan Dziembowski, Sebastian Faust, and Maciej Skorski. Noisy leakage revisited. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 159–188. Springer, Heidelberg, April 2015.

- [DFS19] Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making masking security proofs concrete (or how to evaluate the security of any leaking device), extended version. *Journal of Cryptology*, 32(4):1263–1297, October 2019.
- [DHLW10] Yevgeniy Dodis, Kristiyan Haralambiev, Adriana López-Alt, and Daniel Wichs. Cryptography against continuous memory attacks. In *51st FOCS*, pages 511–520. IEEE Computer Society Press, October 2010.
- [DKL09] Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 621–630. ACM Press, May / June 2009.
- [DLWW11] Yevgeniy Dodis, Allison B. Lewko, Brent Waters, and Daniel Wichs. Storing secrets on continually leaky devices. In Rafail Ostrovsky, editor, *52nd FOCS*, pages 688–697. IEEE Computer Society Press, October 2011.
- [DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam D. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008.
- [DP08] Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography. In *49th FOCS*, pages 293–302. IEEE Computer Society Press, October 2008.
- [DRS04] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 523–540. Springer, Heidelberg, May 2004.
- [Dzi06a] Stefan Dziembowski. Intrusion-resilience via the bounded-storage model. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 207–224. Springer, Heidelberg, March 2006.
- [Dzi06b] Stefan Dziembowski. On forward-secure storage (extended abstract). In Cynthia Dwork, editor, *CRYPTO 2006*, volume 4117 of *LNCS*, pages 251–270. Springer, Heidelberg, August 2006.
- [Dzi10] Stefan Dziembowski. A lower bound on the key length of information-theoretic forward-secure storage schemes. In Kaoru Kurosawa, editor, *ICITS 09*, volume 5973 of *LNCS*, pages 19–26. Springer, Heidelberg, December 2010.
- [FH15] Benjamin Fuller and Ariel Hamlin. Unifying leakage classes: Simulatable leakage and pseudoentropy. In Anja Lehmann and Stefan Wolf, editors, *Information Theoretic Security*, pages 69–86, Cham, 2015. Springer International Publishing.
- [FNSV18] Antonio Faonio, Jesper Buus Nielsen, Mark Simkin, and Daniele Venturi. Continuously non-malleable codes with split-state refresh. In Bart Preneel and Frederik Vercauteren, editors, *ACNS 18*, volume 10892 of *LNCS*, pages 121–139. Springer, Heidelberg, July 2018.
- [FRR⁺10] Sebastian Faust, Tal Rabin, Leonid Reyzin, Eran Tromer, and Vinod Vaikuntanathan. Protecting circuits from leakage: the computationally-bounded and noisy cases. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 135–156. Springer, Heidelberg, May / June 2010.

- [FRR⁺14] Sebastian Faust, Tal Rabin, Leonid Reyzin, Eran Tromer, and Vinod Vaikuntanathan. Protecting circuits from computationally bounded and noisy leakage. *SIAM J. Comput.*, 43(5):1564–1614, 2014.
- [GIM⁺16] Vipul Goyal, Yuval Ishai, Hemanta K. Maji, Amit Sahai, and Alexander A. Sherstov. Bounded-communication leakage resilience via parity-resilient circuits. In Irit Dinur, editor, *57th FOCS*, pages 1–10. IEEE Computer Society Press, October 2016.
- [GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for NP. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 339–358. Springer, Heidelberg, May / June 2006.
- [HIK⁺04] Prahladh Harsha, Yuval Ishai, Joe Kilian, Kobbi Nissim, and S. Venkatesh. Communication versus computation. In Josep Díaz, Juhani Karhumäki, Arto Lepistö, and Donald Sannella, editors, *ICALP 2004*, volume 3142 of *LNCS*, pages 745–756. Springer, Heidelberg, July 2004.
- [ISW03] Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 463–481. Springer, Heidelberg, August 2003.
- [KJJ99] Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential power analysis. In Michael J. Wiener, editor, *CRYPTO’99*, volume 1666 of *LNCS*, pages 388–397. Springer, Heidelberg, August 1999.
- [KMS19] Ashutosh Kumar, Raghu Meka, and Amit Sahai. Leakage-resilient secret sharing against colluding parties. In David Zuckerman, editor, *60th FOCS*, pages 636–660. IEEE Computer Society Press, November 2019.
- [KMZ20] Ashutosh Kumar, Raghu Meka, and David Zuckerman. Bounded collusion protocols, cylinder-intersection extractors and leakage-resilient secret sharing. Cryptology ePrint Archive, Report 2020/473, 2020. <https://eprint.iacr.org/2020/473>.
- [Koc96] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Neal Kobitz, editor, *CRYPTO’96*, volume 1109 of *LNCS*, pages 104–113. Springer, Heidelberg, August 1996.
- [KR19] Yael Tauman Kalai and Leonid Reyzin. A survey of leakage-resilient cryptography. In *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pages 727–794. ACM, 2019.
- [LCG⁺20] Fuchun Lin, Mahdi Cheraghchi, Venkatesan Guruswami, Reihaneh Safavi-Naini, and Huaxiong Wang. Leakage-resilient secret sharing in non-compartmentalized models. In Yael Tauman Kalai, Adam D. Smith, and Daniel Wichs, editors, *1st Conference on Information-Theoretic Cryptography (ITC 2020)*, volume 163 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 7:1–7:24, Dagstuhl, Germany, 2020. Schloss Dagstuhl–Leibniz-Zentrum für Informatik.
- [LLW11] Allison B. Lewko, Mark Lewko, and Brent Waters. How to leak on key updates. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 725–734. ACM Press, June 2011.

- [LMQW20] Xin Li, Fermi Ma, Willy Quach, and Daniel Wichs. Leakage-resilient key exchange and two-seed extractors. Cryptology ePrint Archive, Report 2020/771, 2020. <https://eprint.iacr.org/2020/771>.
- [LSW16] Fuchun Lin, Reihaneh Safavi-Naini, and Pengwei Wang. Detecting algebraic manipulation in leaky storage systems. In Anderson C. A. Nascimento and Paulo Barreto, editors, *ICITS 16*, volume 10015 of *LNCS*, pages 129–150. Springer, Heidelberg, August 2016.
- [MR04] Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 278–296. Springer, Heidelberg, February 2004.
- [MW97] Ueli M. Maurer and Stefan Wolf. Privacy amplification secure against active adversaries. In Burton S. Kaliski Jr., editor, *CRYPTO'97*, volume 1294 of *LNCS*, pages 307–321. Springer, Heidelberg, August 1997.
- [NS09] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 18–35. Springer, Heidelberg, August 2009.
- [NS12] Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. *SIAM J. Comput.*, 41(4):772–814, 2012.
- [PGMP19] Thomas Prest, Dahmun Goudarzi, Ange Martinelli, and Alain Passelègue. Unifying leakage models on a Rényi day. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 683–712. Springer, Heidelberg, August 2019.
- [PR13] Emmanuel Prouff and Matthieu Rivain. Masking against side-channel attacks: A formal security proof. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 142–159. Springer, Heidelberg, May 2013.
- [PVZ12] Jeff M. Phillips, Elad Verbin, and Qin Zhang. Lower bounds for number-in-hand multiparty communication complexity, made easy. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 486–501, 2012.
- [Sas15] Igal Sason. On reverse Pinsker inequalities. *arXiv e-prints*, page arXiv:1503.07118, March 2015.
- [SV19] Akshayaram Srinivasan and Prashant Nalini Vasudevan. Leakage resilient secret sharing and applications. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 480–509. Springer, Heidelberg, August 2019.
- [Ver14] Sergio Verdú. Total variation distance and the distribution of relative information. In *2014 Information Theory and Applications Workshop (ITA)*, pages 1–3, 2014.
- [Yao79] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *Proceedings of the 11th Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1979, Atlanta, Georgia, USA*, pages 209–213, 1979.

A Further Applications

A.1 Forward-Secure Storage

A forward-secure storage scheme (FSS) [Dzi06b] consists of two polynomial-time algorithms $\Phi = (\text{Enc}, \text{Dec})$ specified as follows:

- The randomized encryption algorithm $\text{Enc}(\kappa, y)$ takes as input a key $\kappa \in \{0, 1\}^k$ and a message $y \in \{0, 1\}^m$, and outputs a ciphertext $x \in \{0, 1\}^n$.
- The deterministic decryption algorithm $\text{Dec}(\kappa, x)$ takes as input a key $\kappa \in \{0, 1\}^k$ and a ciphertext $x \in \{0, 1\}^n$, and outputs a value in $\{0, 1\}^m$.

We require correctness, namely for every key $\kappa \in \{0, 1\}^k$ and every message $y \in \{0, 1\}^m$, it holds that $\text{Dec}(\kappa, \text{Enc}(\kappa, y)) = y$ with probability one over the randomness of Enc .

Intuitively, FSS is a special type of secret-key encryption satisfying the following property: If the adversary has only partial information about the ciphertext $x = \text{Enc}(\kappa, y)$, represented as a leakage function $z = f(x)$, it does not learn anything about the plaintext y even when the key κ becomes public *after* the leakage occurs. The original definition of FSS was in the bounded-leakage setting, where the leakage function f outputs at most ℓ bits. The definition below generalizes this notion to the setting of dense leakage.

Definition 20 (Forward-secure storage). *We say that an FSS scheme $\Phi = (\text{Enc}, \text{Dec})$ is (p, γ, δ) -dense ε -leakage-resilient ($(p, \gamma, \delta, \varepsilon)$ -DLR-FSS for short) if for all $y_0, y_1 \in \{0, 1\}^m$, and for all (p, γ, δ) -dense leakage functions $f : \{0, 1\}^n \rightarrow \{0, 1\}^*$, we have*

$$(K, f(X_0)) \approx_\varepsilon (K, f(X_1)),$$

where K is uniform over $\{0, 1\}^k$ and $X_b \leftarrow_s \text{Enc}(K, y_b)$ for all $b \in \{0, 1\}$.

Moreover, in case the function f in the above definition is:

- an ℓ -bounded leakage function, we say Φ is ℓ -bounded ε -leakage-resilient (or (ℓ, ε) -BLR-FSS);
- an ℓ -ME-noisy leakage function, we say Φ is ℓ -min-entropy-noisy ε -leakage-resilient (or (ℓ, ε) -ME-NLR-FSS);
- an ℓ -U-noisy leakage function, we say Φ is ℓ -uniform-noisy ε -leakage-resilient (or (ℓ, ε) -U-NLR-FSS);

The theorem below says that any bounded leakage-resilient FSS is also secure in the presence of dense leakage.

Theorem 19. *Any (ℓ, ε) -BLR-FSS is also a $(p, \gamma, \delta, \varepsilon')$ -DLR-FSS so long as*

$$\begin{aligned} \ell &= \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2 \\ \varepsilon' &= 3\varepsilon + 2\varepsilon^{1/4\delta} + 2\gamma + 2p. \end{aligned}$$

Proof. Fix an arbitrary (p, γ, δ) -dense leakage function $f : \{0, 1\}^n \rightarrow \{0, 1\}^*$, and any $y_0, y_1 \in \{0, 1\}^m$. By Theorem 3, there exists a randomized simulator Sim_f such that for all $b \in \{0, 1\}$ we have

$$(K, f(X_b)) \approx_{\varepsilon + \varepsilon^{1/4\delta} + \gamma + p} (K, \text{Sim}_f^{\text{Leak}(X_b, \cdot)}), \quad (18)$$

for K uniform over $\{0, 1\}^k$ and $X_b \leftarrow \text{Enc}(K, y_b)$, where Sim_f asks to its oracle a single query consisting of a function $g : \{0, 1\}^n \rightarrow \{0, 1\}^\ell$ with $\ell = \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2$.

Next, we claim that for every (p, γ, δ) -dense leakage function $f : \{0, 1\}^n \rightarrow \{0, 1\}^*$, and for all messages $y_0, y_1 \in \{0, 1\}^m$ we have

$$(K, \text{Sim}_f^{\text{Leak}(X_0, \cdot)}) \approx_\varepsilon (K, \text{Sim}_f^{\text{Leak}(X_1, \cdot)}). \quad (19)$$

By contradiction, assume that there exists a (p, γ, δ) -dense leakage function $f : \{0, 1\}^n \rightarrow \{0, 1\}^*$, a pair of messages $y_0, y_1 \in \{0, 1\}^m$, and an unbounded distinguisher D such that

$$\mathbb{P}\left[D(K, \text{Sim}_f^{\text{Leak}(X_0, \cdot)}) = 1\right] - \mathbb{P}\left[D(K, \text{Sim}_f^{\text{Leak}(X_1, \cdot)}) = 1\right] > \varepsilon.$$

Let $g : \{0, 1\}^n \rightarrow \{0, 1\}^\ell$ be the leakage function output by Sim_f , and denote by \tilde{Z} the output of the simulator. Consider the distinguisher D' that upon receiving $(K, g(X_b))$ returns the same as D upon input (K, \tilde{Z}) . Then, we have

$$\mathbb{P}[D'(K, g(X_0)) = 1] - \mathbb{P}[D'(K, g(X_1)) = 1] > \varepsilon.$$

The above contradicts the fact that Φ is an (ℓ, ε) -BLR-FSS. The theorem now follows by combining Eq. (18) and Eq. (19) with the triangle inequality. \square

Next, using the connections between ME-noisy and U-noisy leakage with dense leakage established in §5, we obtain the following corollary.

Corollary 13. *Any (ℓ', ε') -BLR-FSS is also an:*

(i) (ℓ, ε) -ME-NLR-FSS so long as $\ell = \ell' - O(\log(1/\varepsilon') + \alpha)$ and $\varepsilon = 7\varepsilon'$, and assuming that $\text{Enc}(K, y)$ is α -semi-flat for all $y \in \{0, 1\}^m$ and with K uniform over $\{0, 1\}^k$.

(ii) (ℓ, ε) -U-NLR-FSS so long as $\ell = \ell' - O(\log(1/\varepsilon'))$ and $\varepsilon = 7\varepsilon'$.

Proof. The statement follows by choosing $\beta = 2 + 2 \log(1/\varepsilon')$ and $\varepsilon = \varepsilon'$ in Corollary 2 and Corollary 3. \square

Dziembowski [Dzi10] suggests the following construction of FSS. The encryption of $y \in \{0, 1\}^m$ is a pair of strings $x = (x_0, x_1) = (\rho, \text{Ext}(\kappa_0, \rho) \oplus \kappa_1 \oplus y) \in \{0, 1\}^{r+m}$, where $\text{Ext} : \{0, 1\}^k \times \{0, 1\}^r \rightarrow \{0, 1\}^m$ is a seeded extractor and $\kappa = (\kappa_0, \kappa_1) \in \{0, 1\}^{k+m}$ is the secret key. Note that the distribution of x is flat. Assuming that Ext is a seeded extractor with error ε for sources of min-entropy $\mathbb{H}_\infty(R) \geq r - \ell - \lambda$, then the above construction is an $(\ell, 2\varepsilon + 2^{-\lambda})$ -BLR-FSS. By the above corollary, the same scheme is also secure¹⁶ in the setting of ME-noisy (and thus U-noisy) leakage.

A.2 Extractors for Cylinder-intersection Sources

A multi-source extractor Ext is a deterministic efficient algorithm taking as input n independent m -bit sources with min-entropy at least $t \leq m$ (a.k.a. (m, t) -sources), and outputting a d -bit string which is statistically close to uniform. BCP extractors (sometimes known as extractors for cylinder-intersection sources [KMS19]) further strengthen this property by requiring the above to hold even if the distinguisher can obtain adaptive joint leakage from subsets of the sources with size at most $k < n$ (modelled as a BC-BCP). Such extractors are called strong if

¹⁶A similar result was also proven explicitly in [CFV19] for the case of ME-noisy leakage.

the latter property even holds when the extractor is further given any subset \mathcal{S} of at most k sources along with the output of the extractor.

Below, we generalize the definition of BCP strong extractors to the case of NC-BCPs and DC-BCPs.

Definition 21 (BCP strong extractor). *Let X_1, \dots, X_n be n independent (m, t) -sources. A deterministic function $\text{Ext} : (\{0, 1\}^m)^n \rightarrow \{0, 1\}^d$ is an $(m, t, n, r, p, \gamma, \delta, k)$ -DC-BCP strong extractor with error ε if for all $(n, r, p, \gamma, \delta, k)$ -DC-BCP π and for all subsets \mathcal{S} of size $|\mathcal{S}| = k$ we have*

$$(\text{Ext}(X_1, \dots, X_n), \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) \approx_\varepsilon (U_d, \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}).$$

Moreover, in case the protocol π in the above definition is an:

- (n, r, ℓ, k) -NC-BCP, we say that Ext is an (m, t, n, r, ℓ, k) -NC-BCP strong extractor;
- (n, r, ℓ, k) -BC-BCP, we say that Ext is an (m, t, n, r, ℓ, k) -BC-BCP strong extractor.

Corollary 14. *Every (m, t, n, r, ℓ, k) -BC-BCP strong extractor Ext with error ε is also an $(m, t, n, r, p, \gamma, \delta, k)$ -DC-BCP strong extractor with error ε' so long as $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$ and $\varepsilon' = \varepsilon + 2r \cdot (\varepsilon + \varepsilon^{1/4} + \gamma + p)$.*

Proof. By applying Theorem 17, we can replace the transcript $\pi(X_1, \dots, X_n)$ of any $(n, r, p, \gamma, \delta, k)$ -DC-BCP with a simulated transcript $\tilde{\pi}(X_1, \dots, X_n)$ obtained using the transcript $\pi'(X_1, \dots, X_n)$ of an (n, r, ℓ, k) -BC-BCP with $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma)) + 2)$. Hence, we have

$$\begin{aligned} (\text{Ext}(X_1, \dots, X_n), \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) &\approx_{\tilde{\varepsilon}} (\text{Ext}(X_1, \dots, X_n), \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) \\ (U_d, \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) &\approx_{\tilde{\varepsilon}} (U_d, \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}), \end{aligned}$$

where $\tilde{\varepsilon} = \varepsilon + \varepsilon^{1/4} + \gamma + p$.

Finally, by the fact that Ext is an (m, t, n, ℓ, k) -BC-BCP strong extractor with error ε , it holds that

$$(\text{Ext}(X_1, \dots, X_n), \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) \approx_\varepsilon (U_d, \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}).$$

The statement now follows by combining the above equations. \square

Corollary 15. *Every (m, t, n, r, ℓ', k) -BC-BCP strong extractor Ext with error ε' is also an (m, t, n, r, ℓ, k) -NC-BCP strong extractor with error ε so long as $\ell' = \ell + r \cdot (O(\log(r/\varepsilon)))$ and $\varepsilon = 3\varepsilon'$.*

Proof. By setting $\delta = \varepsilon'/(3r)$ in Theorem 18, we can replace the transcript $\pi(X_1, \dots, X_n)$ of any (n, r, ℓ, k) -NC-BCP with a simulated transcript $\tilde{\pi}(X_1, \dots, X_n)$ obtained using the transcript $\pi'(X_1, \dots, X_n)$ of an (n, r, ℓ', k) -BC-BCP with $\ell' \leq \ell + r \cdot (6 + 2 \log(3r/\varepsilon') + \log \log(3r/\varepsilon')) = \ell + r \cdot (O(\log(r/\varepsilon)))$. Hence, we have

$$\begin{aligned} (\text{Ext}(X_1, \dots, X_n), \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) &\approx_{\varepsilon'} (\text{Ext}(X_1, \dots, X_n), \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) \\ (U_d, \pi(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) &\approx_{\varepsilon'} (U_d, \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}). \end{aligned}$$

Finally, by the fact that Ext is an (m, t, n, ℓ', k) -BC-BCP strong extractor with error ε' , it holds that

$$(\text{Ext}(X_1, \dots, X_n), \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}) \approx_{\varepsilon'} (U_d, \tilde{\pi}(X_1, \dots, X_n), (X_i)_{i \in \mathcal{S}}).$$

The statement now follows by combining the above equations. \square

Explicit constructions of cylinder-intersection extractors are known for sources with a constant fraction of min-entropy (say $t = 0.3 \cdot m$) and collusion bound $k = n - 1$ so long as $n = O(\log m)$ [BNS92, LMQW20], and very recently even for $k = \omega(\log n)$ so long as $n \gg k$ [CGGL20, KMZ20]. By the above corollaries, these extractors are also secure in the settings of dense and U-noisy leakage.

A.3 Symmetric NIKE

A symmetric-key NIKE protocol with secret key space \mathcal{SK} , private state space \mathcal{R} , public message space \mathcal{M} and output key space \mathcal{K} consists of two algorithms $\Pi = (\text{Publish}, \text{SharedKey})$ specified as follows:

- The deterministic algorithm $\text{Publish}(sk, \rho)$ takes as input a secret key $sk \in \mathcal{SK}$, a private state $\rho \in \mathcal{R}$ and outputs a public message $\mu \in \mathcal{M}$.
- The deterministic algorithm $\text{SharedKey}(sk, \rho, \mu)$ takes as input a secret key $sk \in \mathcal{SK}$, a private state $\rho \in \mathcal{R}$, and a public message $\mu \in \mathcal{M}$, and outputs a key $\kappa \in \mathcal{K}$.

We require correctness, namely when Alice and Bob run a NIKE protocol they end up sharing the same key. More formally, for all secret keys $sk \in \mathcal{SK}$ and all private states $\rho_1, \rho_2 \in \mathcal{R}$ it holds that $\text{SharedKey}(sk, \rho_1, \mu_2) = \text{SharedKey}(sk, \rho_2, \mu_1)$ where $\mu_1 = \text{Publish}(sk, \rho_1)$ and $\mu_2 = \text{Publish}(sk, \rho_2)$.

Intuitively, a NIKE protocol is said to be leakage resilient if the shared key is statistically close to random to the eyes of a distinguisher that can independently leak information from the private state (sk, ρ_1) of Alice and from the private state (sk, ρ_2) of Bob. Depending on the leakage being bounded, noisy or dense, we obtain the following definition.

Definition 22 (Symmetric-key NIKE with BCP leakage resilience). *We say that a symmetric-key NIKE protocol $\Pi = (\text{Publish}, \text{SharedKey})$ is $(t, r, p, \gamma, \delta, \varepsilon)$ -secure against interactive dense leakage $((t, r, p, \gamma, \delta, \varepsilon)$ -DLR-NIKE, for short) if for any distribution SK such that $\mathbb{H}_\infty(SK) \geq t$ and all $(2, r, p, \gamma, \delta, 1)$ -DC-BCP π we have*

$$(M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_1) \approx_\varepsilon (M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_2),$$

where R_1, R_2 are uniform over \mathcal{R} , and where $M_1 = \text{Publish}(SK, R_1)$, $M_2 = \text{Publish}(SK, R_2)$ and $K_1 = \text{SharedKey}(SK, M_2, R_1)$, $K_2 = \text{SharedKey}(SK, M_1, R_2)$.

Moreover, in case the protocol π in the above definition is a:

- $(2, r, \ell, k)$ -NC-BCP, we say that Π is $(t, r, \ell, \varepsilon)$ -secure against interactive noisy leakage $((t, r, \ell, \varepsilon)$ -NLR-NIKE for short);
- $(2, r, \ell, k)$ -BC-BCP, we say that Π is $(t, r, \ell, \varepsilon)$ -secure against interactive bounded leakage $((t, r, \ell, \varepsilon)$ -BLR-NIKE for short);

Corollary 16. *Every $(t, r, \ell, \varepsilon)$ -BLR-NIKE Π is also a $(t, r, p, \gamma, \delta, \varepsilon)$ -DLR-NIKE so long as $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma))) + 2$ and $\varepsilon' = \varepsilon + 2r \cdot (\varepsilon + \varepsilon^{1/4} + \gamma + p)$.*

Proof. By applying Theorem 17, we can replace the transcript $\pi(X_1, \dots, X_n)$ of any $(2, r, p, \gamma, \delta, 1)$ -BC-BCP with a simulated transcript $\tilde{\pi}(X_1, \dots, X_n)$ obtained using the transcript $\pi'(X_1, \dots, X_n)$ of a $(2, r, \ell, 1)$ -BC-BCP with $\ell = \log(1/\delta) + r \cdot (\log \log(1/\varepsilon) + 2 \log(1/(1 - \gamma))) + 2$. Hence, we have

$$(M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_1) \approx_{\tilde{\varepsilon}} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_1)$$

$$(M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_2) \approx_{\tilde{\varepsilon}} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_2),$$

where $\tilde{\varepsilon} = \varepsilon + \varepsilon^{1/4} + \gamma + p$ and R_1, R_2 are uniform over \mathcal{R} , and where $M_1 = \text{Publish}(SK, R_1)$, $M_2 = \text{Publish}(SK, R_2)$ and $K_1 = \text{SharedKey}(SK, M_2, R_1)$, $K_2 = \text{SharedKey}(SK, M_1, R_2)$.

Finally, by the fact that Π is a $(t, r, \ell, \varepsilon)$ -BLR-NIKE, it holds that

$$(M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_1) \approx_{\varepsilon} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_2).$$

The statement now follows by combining the above equations. \square

Corollary 17. *Every $(t, r, \ell', \varepsilon')$ -BLR-NIKE Π is also a $(t, r, \ell, \varepsilon)$ -NLR-NIKE so long as $\ell' = \ell + r \cdot (O(\log(r/\varepsilon)))$ and $\varepsilon = 3\varepsilon'$.*

Proof. By setting $\delta = \varepsilon'/(3r)$ in Theorem 18, we can replace the transcript $\pi(X_1, \dots, X_n)$ of any $(2, r, \ell, 1)$ -NC-BCP with a simulated transcript $\tilde{\pi}(X_1, \dots, X_n)$ obtained using the transcript $\pi'(X_1, \dots, X_n)$ of a $(2, r, \ell', 1)$ -BC-BCP with $\ell' \leq \ell + r \cdot (6 + 2 \log(3r/\varepsilon') + \log \log(3r/\varepsilon')) = \ell + r \cdot (O(\log(r/\varepsilon)))$. Hence, we have

$$\begin{aligned} (M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_1) &\approx_{\varepsilon'} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_1) \\ (M_1, M_2, \pi((SK, R_1), (sk, R_2)), K_2) &\approx_{\varepsilon'} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_2), \end{aligned}$$

where R_1, R_2 are uniform over \mathcal{R} , and where $M_1 = \text{Publish}(SK, R_1)$, $M_2 = \text{Publish}(SK, R_2)$ and $K_1 = \text{SharedKey}(SK, M_2, R_1)$, $K_2 = \text{SharedKey}(SK, M_1, R_2)$.

Finally, by the fact that Π is a $(t, r, \ell', \varepsilon')$ -BLR-NIKE, it holds that

$$(M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_1) \approx_{\varepsilon'} (M_1, M_2, \tilde{\pi}((SK, R_1), (sk, R_2)), K_2).$$

The statement now follows by combining the above equations. \square

Li, Ma, Quach, and Wicks [LMQW20] showed how to construct BLR-NIKE protocols for $\ell = \Omega(m)$ and $\varepsilon = 2^{-\Omega(m)}$. By the above corollaries, these protocols are also secure in the setting of dense and U-noisy leakage.

A.4 Multi-Party Computation

We start with the notion of multi-party computation (MPC) with *passive security*. As usual we assume secure point-to-point channels between the parties. Moreover, we assume that the environment sends the inputs to the parties non-adaptively; namely, the inputs are not chosen as a function of the protocol execution, and the inputs are sent as first step of the execution of the environment. We call such environments *semi-honest*.

Additionally, we assume that the environment can receive leakage from the *private* states of all parties. More precisely, at the end of the protocol execution, the environment receives, together with the outputs of all parties, the value $g_i(st_i)$ for each $i \in [n]$, where st_i is the secret state of the party P_i . Notice that, although we import terminologies and notations from the UC framework of Canetti [Can00], we will not aim at modelling universal composability. For instance, the latter means we see the parties of the protocol as monolithic machines (i.e., they cannot span any other sub-machine).

The private state of party P_i is defined as the tuple $st_i := (\rho_i, y_1, \dots, y_r)$, where ρ_i is the random tape of the party, and y_j is the message received by the party at round $j \in [r]$. Notice that the latter includes also all messages received by all the ideal resources that the protocol needs. Also, observe that we do not include the secret input of the parties in the private states, the reason being that otherwise it would be impossible for the simulator to accomplish its task (see [BCH12]).

Real and ideal experiments. We denote by $\text{REAL}_{Z,G,\pi,g}(z)$ the random variable describing the output of the environment Z with input $z \in \{0,1\}^\lambda$, running with the protocol π and with ideal resource G , and obtaining the values $g_i(st_i)_{i \in [n]}$. We denote by $\text{IDEAL}_{Z,S,F,g}(z)$ the random variable describing the output of the environment Z with input $z \in \{0,1\}^\lambda$ and running with the simulator S and with the ideal resource F .

Definition 23 (Passively-secure leakage-resilient MPC). *A protocol π computes F using the ideal resource G with statistical passive ε -security (resp. with computational passive ε -security) in the presence of \mathcal{G} -leakage if for every semi-honest environment (resp. PPT semi-honest environment) Z and for every $g \in \mathcal{G}$, there exists a PPT simulator S such that:*

$$\{\text{REAL}_{Z,G,\pi,g}(z)\}_{z \in \{0,1\}^\lambda, \lambda \in \mathbb{N}} \approx_\varepsilon \{\text{IDEAL}_{Z,S,F,g}(z)\}_{z \in \{0,1\}^\lambda, \lambda \in \mathbb{N}}.$$

Consider the ideal resource G_{CRS} that models security in the common reference string (CRS) model. Specifically, G_{CRS} upon input 1^λ from a party P_i first checks if it is the first time that the ideal resource is called with input 1^λ . If this is the case, G_{CRS} samples a string $crs \in \{0,1\}^\lambda$ uniformly at random, returns crs to the calling party, and stores crs for future uses. Otherwise, G_{CRS} retrieves the string crs associated to 1^λ and returns it. Additionally, consider the ideal functionality F_{SFE}^ϕ of *secure function evaluation* with parameter a function ϕ represented as a circuit. The functionality takes inputs $x_1, \dots, x_n \in \{0,1\}^*$, computes $y = \phi(x_1, \dots, x_n)$, and outputs y to the parties. We are ready to state the main theorem of this section.

Theorem 20. *Let $p, \gamma, \delta \in \mathbb{R}$, and $\ell \in \mathbb{N}$ be parameters, and G be an ideal resource. Assume there exists a family of protocols $\{\pi_\phi\}_{\phi \in \{0,1\}^*}$ such that for any circuit $\phi \in \{0,1\}^*$ the protocol π_ϕ computes F_{SFE}^ϕ using the ideal resource G with statistical passive ε -security in the presence of ℓ -bounded leakage. If one-way functions exist, then there exists a family of protocols $\{\pi'_\phi\}_{\phi \in \{0,1\}^*}$ such that for any circuit $\phi \in \{0,1\}^*$ the protocol π'_ϕ computes F_{SFE}^ϕ using the ideal resources (G, G_{CRS}) with:*

- (i) *computational passive ε' -security in the presence of (p, γ, δ) -dense leakage, where $\varepsilon' = 2n(\varepsilon + \varepsilon^{1/4\delta} + \gamma + p) + \varepsilon$ given that $\ell = \log(1/\delta) + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-\gamma}\right) + 2$.*
- (ii) *computational passive ε' -security in the presence of ℓ' -U-noisy leakage, where $\varepsilon' = 2n(\varepsilon + \varepsilon^{2^\ell + \beta - 2} + 2^{-\beta/2 + 1}) + \varepsilon$ given that $\ell = \ell' + \beta + \log \log(1/\varepsilon) + 2 \log\left(\frac{1}{1-2^{-\beta/2}}\right) + 2$ for any $\beta > 0$.*

The proof of the theorem is divided into two steps. First, we define a weaker notion of security for leakage-resilient MPC, which is reminiscent of the notion of witness indistinguishability for proof systems. In more detail, our notion stipulates that real-world views of protocol executions that lead to the same output are indistinguishable, even in the presence of leakage. For this relaxed security notion, we will use our main Theorem 3 to reduce bounded-leakage resilience to dense-leakage resilience (similarly to the proof of Theorem 15). The second step of the proof of Theorem 20 shows that protocols that satisfy our relaxed notion can be used to construct protocols that are secure according to Definition 23 in the CRS model. Our approach here is similar to the classical transformation from witness indistinguishability to zero knowledge in the context of proof systems.

Definition 24 (View indistinguishability in the presence of leakage). *Let $\text{OUT}_{Z,G,\pi}(z)$ be the random variable that indicates the outputs of the parties at the end of the execution of protocol π upon input $z \in \{0,1\}^\lambda$ for the environment. We say that a protocol π computes F using the ideal resource G with ε -indistinguishability of views in the presence of \mathcal{G} -leakage if for any Z ,*

any $g \in \mathcal{G}$, and any sequences of values $\{z_\lambda\}_{\lambda \in \mathbb{N}, z_\lambda \in \{0,1\}^\lambda}$, $\{z'_\lambda\}_{\lambda \in \mathbb{N}, z'_\lambda \in \{0,1\}^\lambda}$, $\{y_\lambda\}_{\lambda \in \mathbb{N}, y_\lambda \in \{0,1\}^*}$ we have

$$\{\text{REAL}_{\mathbb{Z}, \mathcal{G}, \pi, g}(z_\lambda) \mid \text{OUT}_{\mathbb{Z}, \mathcal{G}, \pi}(z_\lambda) = y_\lambda\}_{\lambda \in \mathbb{N}} \approx_\epsilon \{\text{REAL}_{\mathbb{Z}, \mathcal{G}, \pi, g}(z'_\lambda) \mid \text{OUT}_{\mathbb{Z}, \mathcal{G}, \pi}(z'_\lambda) = y_\lambda\}_{\lambda \in \mathbb{N}}.$$

As mentioned before, we divide the proof of Theorem 20 into two lemmas.

Lemma 12. *For any ideal resources \mathbb{F} and \mathbb{G} , if π computes \mathbb{F} using the ideal resource \mathbb{G} with statistical ϵ -indistinguishability of views in the presence of ℓ -bounded leakage, then π computes \mathbb{F} using the ideal resource \mathbb{G} with statistical ϵ' -indistinguishability of views in the presence of (p, γ, δ) -dense leakage, where ℓ and ϵ' are defined as in Theorem 20.*

Proof. For the sake of simplicity fix some $\lambda \in \mathbb{N}$. By assumption, we have that for any two inputs $z \in \{0,1\}^\lambda$ and $z' \in \{0,1\}^\lambda$ with $\text{OUT}_{\mathbb{Z}, \mathcal{G}, \pi}(z) = \text{OUT}_{\mathbb{Z}, \mathcal{G}, \pi}(z')$ and any vector (g_1, \dots, g_n) of ℓ -bounded leakage functions, it holds that

$$(g_1(ST_1), \dots, g_n(ST_n)) \approx_\epsilon (g_1(ST'_1), \dots, g_n(ST'_n)),$$

where ST_i is the random variable describing the private state st_i of party $i \in [n]$ in an execution of protocol π with uniformly random tapes and input z , and ST'_i is the corresponding random variable for an executions on input z' .

Fix an arbitrary vector (h_1, \dots, h_n) of (p, γ, δ) -dense leakage functions. Theorem 3 tells us that there exists a vector of simulators $(\text{Sim}_{g_1}^{\text{Leak}(ST_1, \cdot)}, \dots, \text{Sim}_{g_n}^{\text{Leak}(ST_n, \cdot)})$, which have access to an ℓ -bounded leakage oracle Leak , where each simulator can submit a single query. For $\hat{\epsilon} = \epsilon + \epsilon^{1/4\delta} + \gamma + p$, and by applying the triangle inequality n times, we get

$$(g_1(ST_1), \dots, g_n(ST_n)) \approx_{n\hat{\epsilon}} (\text{Sim}_{g_1}^{\text{Leak}(ST_1, \cdot)}, \dots, \text{Sim}_{g_n}^{\text{Leak}(ST_n, \cdot)})$$

and since

$$(\text{Sim}_{g_1}^{\text{Leak}(ST_1, \cdot)}, \dots, \text{Sim}_{g_n}^{\text{Leak}(ST_n, \cdot)}) \approx_\epsilon (\text{Sim}_{g_1}^{\text{Leak}(ST'_1, \cdot)}, \dots, \text{Sim}_{g_n}^{\text{Leak}(ST'_n, \cdot)})$$

the theorem statement follows in a way similar to the proof of Theorem 15. \square

The following corollary is obtained by combining Corollary 3 and Lemma 12.

Corollary 18. *If π computes \mathbb{F} using the ideal resource \mathbb{G} with statistical ϵ -indistinguishability of views in the presence of ℓ -bounded leakage, then for any $\beta > 0$ we have that π computes \mathbb{F} using the ideal resource \mathbb{G} with statistical ϵ' -indistinguishability in presence of ℓ' - U -noisy leakage where $\epsilon' = 2n(\epsilon + \epsilon^{2^\ell + \beta - 2} + 2^{-\beta/2 + 1}) + \epsilon$ given that $\ell = \ell' + \beta + \log \log(1/\epsilon) + 2 \log(\frac{1}{1 - 2^{-\beta/2}}) + 2$.*

We continue with the proof of the main theorem of this section. Let f be a one-way function with domain and co-domain in $\{0,1\}^*$ that is injective from $\{0,1\}^\lambda$ to $\{0,1\}^\lambda$ for any $\lambda \in \mathbb{N}$. Given a circuit ϕ , consider the circuit ϕ_{crs} with parameter a string $crs \in \{0,1\}^\lambda$ that executes the following steps:

Circuit $\phi_{crs}(x_1, \dots, x_n)$:

- If there exists index $i \in [n]$ such that x_i can be parsed as (tp, y) and $f(tp) = crs$ then output y .
- Otherwise output $\phi(x_1, \dots, x_n)$.

Let π'_ϕ , for $\phi \in \{0, 1\}^*$, be the protocol with ideal resources \mathbf{G}_{CRS} and \mathbf{G} where the party P_i does the following:

- Call the ideal resource \mathbf{G}_{CRS} with input 1^λ and obtain crs .
- Run the protocol $\pi_{\phi_{crs}}$ with the other parties.

Lemma 13. *For any leakage family \mathcal{G} , if the protocol $\pi_{\phi_{crs}}$ computes $\mathbf{F}_{\text{SFE}}^{\phi_{crs}}$ using the ideal resource \mathbf{G} with ε -indistinguishability of views in the presence of \mathcal{G} -leakage, then the protocol π'_ϕ computes $\mathbf{F}_{\text{SFE}}^\phi$ using the ideal resources $(\mathbf{G}, \mathbf{G}_{\text{CRS}})$ with computational passive ε -security in the presence of \mathcal{G} -leakage.*

Proof. Let Z be a PPT environment and $g \in \mathcal{G}$. Our goal is to devise an appropriate simulator S . The first step of Z is to corrupt the parties and submit all inputs. With this in mind, let \mathcal{I} denote the set of the corrupted parties, and let x_1, \dots, x_n denote the inputs. The simulator S receives the inputs of the corrupted parties $(x_i)_{i \in \mathcal{I}}$ and the output y . Then, S simulates the common reference string by setting $crs = f(tp)$ for uniformly chosen $tp \in \{0, 1\}^\lambda$. To simulate the views, S runs the protocol $\pi_{\phi_{crs}}$ simulating all parties. In particular, it sets the inputs of the honest parties to the values (tp, y) , and then it sends the value $g((st_i)_{i \in [n]})$ to Z . Since Z is semi-honest, we can assume without loss of generality that Z parses the input z as $\mathcal{I}, x_1, \dots, x_n, aux$, corrupts the parties in \mathcal{I} and uses x_1, \dots, x_n as the inputs for the protocol execution. Let CRS be the random variable describing the common reference string output by the ideal resource \mathbf{G}_{CRS} . For any $crs \in \{0, 1\}^\lambda$, and for any $z = (\mathcal{I}, x_1, \dots, x_n, aux)$ let $z' = (\mathcal{I}, x'_1, \dots, x'_n, aux)$ where $x'_i = x_i$ if $i \in \mathcal{I}$, and otherwise $x'_i = (f^{-1}(crs), y)$ with $y = \phi(x_1, \dots, x_n)$. Then, we have

$$(\text{IDEAL}_{Z, S, \mathbf{F}_{\text{SFE}}^\phi}(z) | CRS = crs) \equiv \text{REAL}_{Z, \mathbf{G}, \pi_{\phi_{crs}}, g}(z').$$

Moreover, by the indistinguishability of views of $\pi_{\phi_{crs}}$, and since executions with z and z' lead to the same output, we have

$$\text{REAL}_{Z, \mathbf{G}, \pi_{\phi_{crs}}, g}(z') \approx_\varepsilon \text{REAL}_{Z, \mathbf{G}, \pi_{\phi_{crs}}, g}(z)$$

for any crs . The proof now follows by taking the average over all the assignments crs of the random variable CRS . \square

As a concrete instantiation, we use the results of Goyal, Ishai, Maji, Sahai and Sherstov [GIM⁺16], who constructed a family of 2PC protocols that are passively secure in the presence of bounded leakage. In particular, they show (see [GIM⁺16, Corollary 5]) that, for any ϕ and any λ , there exists a 2PC protocol $\pi_{\lambda, \phi}$ that computes $\mathbf{F}_{\text{SFE}}^\phi$ using the ideal resource \mathbf{G}_{OT} of oblivious transfer with statistical passive $2^{-\lambda}$ -security in the presence of λ -bounded leakage.

The following corollary is obtained by applying Theorem 20 to the protocols of [GIM⁺16], and by setting $\beta = O(\log^2 \lambda)$ and $\varepsilon = 2^{-\lambda}$, thus obtaining computational passive $\text{negl}(\lambda)$ -security in presence of $\Omega(\lambda)$ -U-noisy leakage.

Corollary 19. *If one-way functions exist, then there is a family of protocols $\{\pi'_\phi\}_{\phi \in \{0, 1\}^*}$ where for any circuit $\phi \in \{0, 1\}^*$ protocol π'_ϕ computes $\mathbf{F}_{\text{SFE}}^\phi$ using the ideal resources $(\mathbf{G}_{\text{OT}}, \mathbf{G}_{\text{CRS}})$ with computational passive $\text{negl}(\lambda)$ -security in the presence of $\Omega(\lambda)$ -U-noisy leakage.*

Following [GIM⁺16], we finally show that if we implement \mathbf{G}_{OT} using a *joint simulation secure* OT protocol, then we obtain a 2PC for $\mathbf{F}_{\text{SFE}}^\phi$ in the common reference string model that is passively secure in the presence of noisy leakage.

Definition 25. Let $F : \mathcal{X}_1 \times \mathcal{X}_2 \rightarrow \mathcal{Y}_1 \times \mathcal{Y}_2$ be a deterministic function. We say that a protocol π computes F with ϵ -joint simulation security if it correctly computes F and there exist polynomial-time simulators S_1, S_2 and a randomness distribution T shared between S_1 and S_2 such that for all inputs x_1, x_2 we have

$$(S_1(x_1, y_1, T), S_2(x_2, y_2, T)) \approx_\epsilon \text{VIEW}_\pi(x_1, x_2),$$

where $(y_1, y_2) = f(x_1, x_2)$ and $\text{VIEW}_\pi(x_1, x_2)$ is the random variable corresponding to the full view of parties P_1 and P_2 running protocol π upon inputs x_1 and x_2 .

Lemma 14. Assume that π computes F using the ideal resources G, G' with computational passive ϵ -security in the presence of ℓ - U -noisy leakage, and that π' computes G with perfect joint simulation security. Then, the protocol π^* where each invocation to G is replaced by an independent execution of π' computes F using the ideal resource G' with computational passive $\text{poly}(\epsilon)$ -security in the presence of ℓ - U -noisy leakage.

Proof. Let q the number of invocations of π' in π^* , and consider the hybrid experiment HYBRID_i that for $j \leq i$ executes π' on the j -th execution with prescribed inputs $x_1^{(j)}, x_2^{(j)}$, obtains outputs $y_1^{(j)}, y_2^{(j)}$, and then runs the simulators $S_1(x_1, y_1, T_j), S_2(x_2, y_2, T_j)$ and performs the leakage over the view created by the simulators.

Clearly, we have $\text{HYBRID}_i \approx_\epsilon \text{HYBRID}_{i+1}$. Notice that HYBRID_q is equivalent to $\text{REAL}_{Z, G, \pi, \hat{g}}$ where the leakage function \hat{g} is defined as $\hat{g} = (\hat{g}_1, \hat{g}_2)$, where \hat{g}_i has hard-coded randomness T_1, \dots, T_q and, upon input the view from P_i , runs $S_i(x_i^{(j)}, y_i^{(j)})$ for every pair $(x_i^{(j)}, y_i^{(j)})$ and then applies the leakage function g_i . Notice that if g is ℓ - U -noisy leakage, then so is \hat{g} . Indeed, let X be uniformly distributed over $\{0, 1\}^v$, where v denotes the size of a transcript of π^* , and let \hat{X} be uniformly distributed over $\{0, 1\}^{\hat{v}}$, where \hat{v} denotes the size of a transcript of π . It is clear that $\tilde{\mathbb{H}}_\infty(X|g(X)) = \tilde{\mathbb{H}}_\infty(\hat{X}|\hat{g}(\hat{X})) - (v - \hat{v})$, because the outputs of \hat{g} and g are identically distributed. Thus, if \hat{g} is not ℓ - U -noisy leakage we have $\tilde{\mathbb{H}}_\infty(X|g(X)) < \mathbb{H}_\infty(\hat{X}) - \ell + v - \hat{v} = \mathbb{H}_\infty(X) - \ell$, contradicting the fact that g is ℓ - U -noisy leakage.

Finally, we can define the simulator for π^* to be equal to the simulator for π with leakage function \hat{g} . \square

By [GIM⁺16, Theorem 9], we have a protocol for G_{OT} in the common reference string model that has $\text{negl}(\lambda)$ -joint-simulation security based on the DDH assumption. Combining this observation with the above discussion leads to the following corollary.

Corollary 20. If the DDH assumption holds, then there exists a family of protocols $\{\pi'_\phi\}_{\phi \in \{0, 1\}^*}$ where for any circuit $\phi \in \{0, 1\}^*$ the protocol π'_ϕ computes F_{SFE}^ϕ using the ideal resource G_{CRS} with computational passive $\text{negl}(\lambda)$ -security in the presence of $\Omega(\lambda)$ - U -noisy leakage.