Computational Notions of Entropy: Classical, Quantum, and Applications

Computational Notions of Entropy: Classical, Quantum, and Applications

A dissertation presented

by

Yi-Hsiu Chen

to

John A. Paulson School Of Engineering And Applied Sciences

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

in the subject of

Computer Science

Harvard University

Cambridge, Massachusetts

May 2019

Dissertation Advisor:
Professor Salil P. Vadhan

Author:
Yi-Hsiu Chen

Computational Notions of Entropy: Classical, Quantum, and Applications

Abstract

Entropy notions from information theory have many applications in cryptographic analyses and constructions, where it is most common to consider adversaries with only (polynomially) bounded computational power. Therefore, some computational relaxations of entropy, which capture some properties of entropy from the view of computationally bounded parties are also useful in cryptography and computational complexity. In this thesis, we study computational notions of entropy from several different angles.

First, in many constructions of basic cryptographic primitives, computational entropies serve as key ingredients. For instance, “next-block pseudoentropy” and “next-block accessible entropy” are used in the best known constructions of pseudorandom generators and statistically hiding commitments from one-way functions, respectively. We contribute along these lines in two aspects:

- We introduce a new notion of hardness for one-way functions called *KL-hardness*, which implies both next-block pseudoentropy and inaccessible entropy, and formalizes the duality between them. By the new notion, we also obtain a more modular and illuminating proof that one-way functions imply next-block inaccessible entropy, similar in structure to the proof that one-way functions imply next-block pseudoentropy (Vadhan and Zheng, STOC 2012).
- One common step in the constructions of basic primitives (including pseudorandom generators, statistically hiding commitments, and universal one-way hash functions) from one-way functions is *entropy flattening*, which converts an average-case entropy (e.g., Shannon) to a worst-case entropy (e.g., min-entropy). We show that any flattening

algorithm has to make $\Omega(n^2)$ queries to the function serving as the entropy sources (analogues to the one-way functions in the constructions of cryptographic primitives) where n is the input length of the functions. The result can be viewed as a step towards proving that the current best construction of pseudorandom generators from arbitrary one-way functions (Vadhan and Zheng, STOC 2012) has optimal efficiency.

Then we study the complexity of the problem "simulating auxiliary input": given a joint distribution (X, Z) on $\mathcal{X} \times \{0,1\}^\ell$ and a class of distinguishers $\mathcal{F} : \mathcal{X} \times \{0,1\}^\ell \to \{0,1\}$, construct an "efficient" simulator $\mathsf{h} : \mathcal{X} \to \{0,1\}^\ell$ such that (X, Z) and $(X, \mathsf{h}(X))$ are indistinguishable by any distinguisher $f \in \mathcal{F}$ up to advantage ε. The efficiency of h is measured by circuit size of relative to $\mathcal{F}$, the optimal complexity was known to be $\operatorname{poly}(\varepsilon^{-1}, 2^\ell)$ The existence of such a simulator implies many theorems connected to computational entropies such as the Dense Model Theorem, Impagliazzo's Hardcore Lemma, and the Leakage Chain Rule for "relaxed-HILL pseudoentropy". We improve the existing results from both directions showing a tight complexity bound $\widetilde{\Theta}(\varepsilon^{-2} \cdot 2^\ell)$ for h, which in particular improves the security analysis of some stream-cipher protocols in leakage-resilient cryptography.

Finally, we initiate the study of computational entropies in the quantum setting by proposing several quantum computational notions of entropy that generalize classical notions, studying which classical properties of computational entropies extend to the quantum case and which does not, and illustrating the application of quantum computational entropy in post-quantum cryptography. Specifically, we develop the Quantum Nonuniform Min-max Theorem to prove some properties of quantum computational entropies such as the equivalence between quantum HILL entropy and metric entropy. Notably, we also solve the problem of simulating auxiliary *quantum* input, which we further use for proving the security of Dziembowski and Pietrzak's (FOCS 2008) leakage-resilient stream-cipher against quantum adversaries with quantum leakage in the bounded-quantum-storage model.

Contents

Acknowledgments

Five years ago, I could not imagine that graduate school experience could change me so much in so many aspects. During this journey, many amazing people have helped me and shaped this unique experience. I am greatly indebted to their support and being in part of this.

Firstly, I want to thank my wife Ching-Yu who has always accompanied me in experiencing all the happiness and difficulties, and encourages me to follow my own heart. Also, I thank my parents in Taiwan for being supportive and believing in me unconditionally.

I would like to thank Tal Malkin and Moti Yung who introduced me the beauty of cryptography, and all my brilliant collaborators Rohit, Mark, Kai-Min, Mika, Thibaut, Ching-Yi, Jyun-Jie, Salil, Xiaodi, and Jiapeng. This thesis could not have been done without them, and I have gained much wisdom (both in mathematics and life) from them. I especially thank Kai-Min for hosting me in Sinica, Taiwan for a year and guiding me to the quantum world. I appreciate Boaz Barak, Salil Vadhan, and Leslie Valiant for serving on my dissertation committee, and James Mickens on my qualifying exam committee. I also thank Carol Harlow and Allison Choat for doing great jobs in administrative tasks and scheduling meetings.

Outside of work, I want to thank my brothers and all my friends, including my office mates in both 138 and 334, people in the Taiwanese association, and the members of the badminton club. They added more dimensions to my graduate school life. I thank Jarosław and Chi-Ning for trashing me in go games many times, badminton mates for making sure I am sweating, and Hsuan and Jin for hosting parties, gathering friends and cooking wonderful meals. Surely there are too many people to list. Sincere thanks go to all of you!

Finally and most importantly, I want to express my gratefulness to my advisor Salil Vadhan. Needless to say how great he is as a researcher. Additionally, he is the most incredible mentor that a student could ask for. Besides his direct guidance, I also learned a lot from his practices and demonstrations. Academically, I learned from his persistence and passion in simplifying proofs and pursuing the most fundamental ideas behind theorems. Beyond that, Salil's decency, honesty, and humility also keep inspiring me. For the next stage and the ones after, the knowledge and qualities I have gained will be everlasting.

This research was supported by NSF grant CCF-1420938 and CCF-1763299.

To my family

Chapter 1

Introduction

Entropy notions can be used to measure the uncertainty in a random variable. They play essential roles in cryptography for both upper bounds (e.g., arguing security) and lower bounds (e.g., minimum key lengths needed). Specifically, traditional entropy notions can be used to prove information-theoretic (unconditional) security properties, where adversaries are modeled with unlimited computational power. Information-theoretic security is invulnerable to future developments in computational power, algorithms, or even quantum computers. However, many interesting cryptographic objects and systems in modern cryptography are provably impossible under information-theoretic security. To bypass this barrier, in modern cryptography, it is most common to study models where adversary's computational power is bounded and base security on the hardness of some computational problems for such adversaries. For example, Goldwasser and Micali defined computational indistinguishability [GM84] as the computational analogue of statistical distance to circumvent Shanon's impossibility results on perfectly secure encryption [Sha49]. Similarly, as entropy notions describe the uncertainty of a random variable "information-theoretically", computational notions of entropy (the subjects of this dissertation) describe the uncertainty of a random variable from the eyes of computationally bounded parties.

In this thesis, we study computational notions of entropy from three different angles. First, computational entropies are the key ingredients in many constructions of cryptographic primitives. Second, the notions are connected to many results in cryptography and complexity theory, where most of them can be derived from the *Leakage Simulation Lemma*. Finally, we propose a number of computational entropy notions in the *quantum* setting and show their applications. In the rest of the chapter, we first review some of the essential ideas of computational entropies, then summarize our contributions for each of the three perspectives.

1.1 Background: Computational Notions of Entropies

Depending on purposes, there are many different kinds of computational notions of entropy, where each of them tries to catch some properties of entropy from the view of computationally bounded parties. We categorize these notions into three types:

1. *Pseudoentropy*: a distribution "behaves" like one with *higher* entropy form a view of a computationally bounded adversary.

2. *Accessible entropy*: for a given distribution X, the entropy of the distribution X' generated by a computationally bounded adversary (under the constraint that the support of X contains the support of X') is *lower* than the true entropy of X.

3. *Computational KL-divergence*: a generalization of pseudoentropy, which describes a "distance" between two distributions from a view of a computationally bounded adversary.

Now we introduce the computational entropies in more details.

1.1.1 Pseudoentropies

One of the first proposed notions of pseudoentropy is due to Yao [Yao82], which is based on efficient compression. Then, Håstad, Impagliazzo, Levin, and Luby introduced another class of pseudoentropies [HILL99] based on computational indistinguishability. We call them *HILL-type entropies*. For example, we say a distribution X has *HILL (Shannon) entropy* at least k (written $\mathrm{H}_{\mathsf{HILL\text{-}Sh}}(X) \geq k$) if there exists a distribution X' such that (1) X and X' are computationally indistinguishable, and (2) the Shannon entropy of X'

(written $\mathrm{H}_{\mathsf{Sh}}(X') \geq k$) is at least k, namely $\mathbb{E}_{x \xleftarrow{r} X'}[\log(1/\Pr[X' = x])] \geq k$.[1] The notions by Håstad et al. [HILL99] were introduced as part of their construction of pseudorandom generators (PRGs) from arbitrary one-way functions (OWFs). They also have inspired many other pseudoentropy notions. One of them is a refined notion called *next-block HILL entropy*, which was defined by [HRV10] to get simpler and more efficient constructions of PRGs from OWFs (see Section 1.2).

Another natural class of definitions of pseudoentropies called *metric-type entropies*[2] were defined by Barak, Shaltiel, and Wigderson [BSW03], where the quantifiers in the definition of HILL-type entropies are switched. That is, a distribution X has metric entropy at least k if for every polynomial-size distinguisher D, there exists a distribution X' with entropy at least k that is indistinguishable from X by D. Barak et al. also showed the equivalence (up to some parameter losses) between HILL and metric entropies. With this equivalence, metric-type entropy is a useful intermediate notion for obtaining tighter security proofs in some applications [DP08, FOR15].

Most of the pseudoentropy notions we consider in this thesis are HILL-type and metric-type entropies, both of which are based on computational indistinguishability. There are more variants of pseudoentropy that are less relevant to our works. See [HLR07] and [FR12] for more details about the definitions and the relationships between different notions.

1.1.2 Accessible entropy

On the other hand, an accessible entropy captures the forgeability of random variables by a computationally bounded adversary. A random variable could have accessible entropy lower than its true entropy. Here we provide some intuition of this type of notion through an example [HRVW09].

Let $f : \{0,1\}^{2n} \to \{0,1\}^n$ be sampled from a family of collision-resistant hash functions.

[1] Thorough the thesis, all logarithms are binary unless specified. That is, information quantities are measured in bits.

[2] The name "metric" is from considering distributions in a metric space, where the distance is defined by a family of distinguishers.

Suppose the random variable X is sampled from $\{0,1\}^{2n}$ and $Y = f(X)$. For a typical element $y \in \mathrm{Supp}(Y)$, there are 2^n possible x such that $f(x) = y$, so the entropy of X given that $Y = y$ is n. Now consider the same process executed by a computationally bounded adversary. After y is fixed, since f is collision-resistant, the adversary cannot find $x' \neq x$ such that $f(x') = f(x)$ except negligible probability. Therefore, we say the (computational) "accessible entropy" of X given Y is negligible. More specifically, we say that X has accessible entropy at most k give Y if for every computationally bounded adversary cannot generate a random variable X' with conditional entropy (given Y and the adversary's coin toss) greater than k and $\mathrm{Supp}(X) \subseteq \mathrm{Supp}(X')$.

Accessible entropy is useful as an *upper* bound on computational entropy, and is interesting when it is *smaller* than the real entropy. The gap between true entropy and accessible entropy is called *inaccessible entropy.*

The intuitive concept of inaccessible entropy was used implicitly in [Rom90] and [HNO$^+$09], which were the first construction of universal one-way hash functions (UOWHFs) and statistically-hiding commitment schemes (SHCs) from arbitrary one-way functions (OWFs), respectively. Then the inaccessible entropy was formally defined in [HRVW09, HRVW19] to improve the construction of SHC. Subsequent to that, a variant of inaccessible entropy is used to have a more efficient construction of UOWHFs.

1.1.3 Computational KL-divergence

Let X and Y be distributions on $\mathcal{X}$. The *KL-divergence* (a.k.a. *relative entropy*) from X to Y is defined as $\mathrm{D}_{\mathsf{KL}}(X \,\|\, Y) \overset{\text{def}}{=} \mathbb{E}_{x \overset{\text{r}}{\leftarrow} X}[\log(\Pr[X = x]/\Pr[Y = x])]$. It is a generalization of entropy as $\mathrm{H}_{\mathsf{Sh}}(X) = \log|\mathcal{X}| - \mathrm{D}_{\mathsf{KL}}(X \,\|\, U_{\mathcal{X}})$ where $U_{\mathcal{X}}$ is the uniform distribution over $\mathcal{X}$. For computational relaxations, we in particular consider the "worst-case" notion, *max-divergence* (also called max-relative entropy in some literatures), defined as $\mathrm{D}_{\mathsf{max}}(X \,\|\, Y) \overset{\text{def}}{=} \max_{x \in \mathcal{X}} \log(\Pr[X = x]/\Pr[Y = x])$.

Since max-divergence involves two random variables, there are more ways to define its computational relaxations than for ordinary entropy. Here we only consider the relaxations that follow the idea of defining HILL-type entropies. First, we say that the *HILL-1 max-divergence* from X to Y is small if there exists a distribution X' that is computationally indistinguishable from X, for which the max-divergence between X' and Y is small. Alternatively, the *HILL-2 max-divergence* between distributions X to Y is small if there exists a distribution Y' that is computationally indistinguishable from Y, for which the max-divergence from X to Y' is small.

With those definitions, the Dense Model Theorem [GT08, RTTV08] can be equivalently stated as small HILL-2 max-divergence implies small HILL-1 max-divergence. Another application of the computational max-divergence is in computational differential privacy [MPRV09]. The definition of differential privacy [DMNS06, DKM$^+$06] can be stated using max-divergence. That is, a mechanism M satisfies ε-differential privacy if for all data set x and x' differing only on single row, we have $\mathrm{D}_{\mathsf{max}}(M(x) \,\|\, M(x')) \leq \varepsilon$ and $\mathrm{D}_{\mathsf{max}}(M(x') \,\|\, M(x)) \leq \varepsilon$.[3] Analogously, computational differential privacy can be described in the language of computational max-divergence.

1.2 Application in Constructing Cryptographic Primitives

A one-way function (OWF) [DH76] is the most basic and unstructured object with cryptographic hardness, and is the minimal assumption for complexity-based cryptography [IL89]. Yet a rich class of cryptographic primitives such as CCA-secure symmetric encryption [Lub94, BDJR97], digital signature [DH76], pseudorandom function [GGM84], and zero-knowledge proofs for **NP** [GMW86] are implied by one-way functions [GGM86, HILL99, Rom90, GMW91, GMW87, Nao91, HNO$^+$09]. The above constructions from OWFs start with one or more of the following three basic but more structured building blocks: (1) pseudorandom generators (PRGs) [BM82, Yao82] (2) statistically hiding commitments (SHCs) [BCC88], and (3) universal one-way hash functions (UOWHFs) [NY89]. Therefore, the constructions of PRGs, SHCs, and UOWHFs from OWFs are fundamental and important research topics in complexity and

[3] In this definition, we use natural logarithms in max-divergence to match the convention in differential privacy.

cryptography.

The original constructions of those primitives [ILL89, Has90, Rom90, HNO$^+$09] are rather complicated and inefficient. Since then, a series of subsequent work improved and simplified the constructions: [Hol06, HHR06, HRV10, VZ12] for PRGs, [HRVW19] for SHCs, and [HHR$^+$10] for UOWHFs. Our contribution is in making progress along these lines of work in two aspects:

1. **Unifying computational entropies.** (§2, [ACHV19])

   In many constructions of the three basic primitives, the first step is to design a computational entropy generator G based on OWF f, which creates a gap between real entropy and some form of computational entropy. One of the keys for the improved constructions is using a "good" notion of computational entropy, which on the one hand can be obtained from one-way function efficiently, and on the other hand, characterizes the essential properties of the target primitives. For example, *next-block HILL entropy* and *inaccessible entropy* are used in recent state-of-the-art constructions of PRGs and SHCs from OWFs, respectively.

   We introduce a new notion of hardness called *KL-hardness* for search problems which on the one hand is satisfied by all one-way functions (corresponding to the generator $\mathsf{G}(x) = (f(x), x)$) and on the other hand implies both next-block HILL entropy and inaccessible entropy. Therefore our KL-hardness notion unifies those two computational entropies and sheds light on the apparent duality between them. Additionally, it provides a modular way to obtain next-block inaccessible entropy from one-way functions, which is similar to the proof that one-way functions imply next-block HILL entropy in [VZ12].

2. **Lower bound for flattening entropies.** (§3, [CGVZ18])

   One of the criteria to measure the efficiency of constructions of those basic cryptographic primitives from OWFs is the number of queries to the OWF. The most expensive step in those constructions is *entropy flattening*, which converts, an "average-type" entropy (e.g., Shannon) to a "worst-type" entropy (e.g., min- or max-entropy). In particular, this step has query complexity of $\widetilde{\Theta}(n^2)$, where n is the input length of the OWF, and this is the main efficiency bottleneck in the construction of PRGs.

We model this this problem with the notion of a *flattening algorithm.* A flattening algorithm is also used in reductions between problems complete for statistical zero-knowledge [Oka00, SV97, GSV99a, Vad99]. We show that any flattening algorithm has to make $\Omega(n^2)$ queries to the functions that provide entropy sources (analogous to the OWFs in the constructions of cryptographic primitives) where n is the input length of the source function. In particular, this result can be viewed as a step towards proving that the current best construction of pseudorandom generators from arbitrary one-way functions by Vadhan and Zheng [VZ12] has optimal efficiency.

1.3 Simulating Auxiliary Input (§4, [CCL18])

Besides their usage for constructing cryptographic primitives, computational entropies also directly connect to a number of results in cryptography and computational complexity. For instance, the security of the *leakage-resilient stream-cipher* in [DP08] relies on the *Leakage Chain Rule* (Theorem 5.5.14) for conditional HILL min-entropy [DP08, RTTV08], Impagliazzo's Hardcore Lemma [Imp95] is a special case of the equivalence between conditional HILL min-entropy and unpredictability entropy [HLR07, Zhe13], and the (complexity-theoretic version of) Dense Model Theorem [RTTV08] can be viewed as an equivalence between HILL-1 and HILL-2 relative max-entropy. In fact, all these results can be obtained from the *Leakage Simulation Lemma* [JP14].

In the leakage simulation lemma, we are given a joint distribution (X, Z) where Z is "short." The goal is to find an "efficient" randomized simulator h such that (X, Z) and $(X, \mathsf{h}(X))$ are indistinguishable by a family of distinguishers. The non-triviality comes from the efficiency requirement on h. Otherwise, one can simply hardcode the conditional distribution of Z given $X = x$ for all x.

Besides the results mentioned above, in [TTV09, JP14], they showed that the Leakage Simulation Lemma also implies the Regularity Theorem in [TTV09], Weak Szemerédi's Regularity Lemma [FK99], some connections between various zero-knowledge notions (e.g., every interactive proof system satisfies a weak notion of zero-knowledge) [CLP15].

Our results

The efficiency is measured in how much more complex the simulator h is than distinguishers in the family, which we call its *relative complexity*. Specifically, let ℓ be the bit length of Z and ε be a desired bound on the distinguishing probability. The relative complexity should be polynomial in both ε and 2^ℓ. In this paper, we achieve $\widetilde{O}(2^\ell \varepsilon^{-2})$ for the relative complexity which is better than all previous results [VZ13a, JP14, Sko16a, Sko16b]. On the other hand, we also prove that our simulator is almost optimal by proving a black-box lower bound, where black-box means the simulator can only use distinguishers in a black-box way. We also make a mild assumption that the simulator does not query the distinguisher with same input $x \in \mathrm{Supp}(X)$, which are also assumed implicitly in [LTW11, Zha11, PS16].

An implication of our upper bound result is in leakage-resilient cryptography. In particular, for analyzing the provable security level of the leakage-resilient stream-ciphers by Pietrzak [Pie09], our complexity bound is the first one that provides a non-trivial and security guarantee in some legitimate parameters.

1.4 Computational Notions in Quantum World (§5, [CCL+17])

In this thesis, we initiate the study of computational notions of entropy in the quantum setting. Some of our results consider the “post-quantum cryptography” setting where the adversary has access to a quantum computer, but the object we measure entropies on remains classical. However, we begin by investigating the more general settings where we consider the computational entropies of quantum states instead of only classical random variables.

In more detail, we begin with defining some quantum extensions of computational pseudoentropy notions and study the connections between them. Most of the pseudoentropy notions, including HILL-type, metric-type, and unpredictable entropies can be naturally extended to the quantum setting by replacing the polynomial-size circuits by polynomial-size *quantum circuits*. Then we study which classical theorems about computational entropy extend to the quantum setting. We show that metric (min-)entropy and HILL (min-)entropy are also equivalent in the quantum setting (as shown in the classical case [BSW03, RTTV08]). In the course of that, we develop the quantum analogue of the "Nonuniform Min-max Theorem" [Zhe13], which we prove using the generalization bound for Rademacher complexity. We also prove that the Leakage Simulation Lemma still holds when the "simulation part" is quantum. Based on that, we prove that the construction of leakage-resilient stream-ciphers in [DP08] is secure against a quantum adversary with logarithmic quantum storage even if some quantum information is leaked during computations. Also, it implies that the Leakage Chain Rule for relaxed-HILL entropy holds when the leakage is quantum. On the other side, we have counterexamples showing that the natural quantum versions of Dense Model Theorem do not hold in general. That implies the an inequivalence between different types of computational max-divergence notions, while the notions are equivalent in the classical case by the (classical) Dense Model Theorem.

1.5 Preliminaries

1.5.1 Notation and convention

For a random variable X over $\mathcal{X}$, $\mathrm{Supp}(X) \overset{\text{def}}{=} \{x \in \mathcal{X} : \Pr[X = x] > 0\}$ denotes the support of X. A random variable is *flat* if it is uniform over its support. Random variables will be written with uppercase letters and the associated lowercase letter and calligraphic letter represent a generic element from its support and the sample space of the random variable. For a distribution X over $\mathcal{X}$, $x \overset{\text{r}}{\leftarrow} X$ means x is a random sample drawn from X.

For a natural number n, $[n]$ denotes the set $\{1, 2, \ldots, n\}$ and U_n denotes the uniform distribution over $\{0,1\}^n$. For a finite set $\mathcal{X}$, $|\mathcal{X}|$ denotes its cardinality, and $U_{\mathcal{X}}$ denotes the uniform distribution over $\mathcal{X}$.

poly denotes the set of polynomial functions and negl the set of all negligible functions. That is $\varepsilon \in \mathrm{negl}$ if for all $p \in \mathrm{poly}$ and large enough $n \in \mathbb{N}$, $\varepsilon(n) \leq 1/p(n)$. We will sometimes abuse notations and write $\mathrm{poly}(n)$ to mean $p(n)$ for some $p \in \mathrm{poly}$ and similarly for $\mathrm{negl}(n)$.

For a function f, $\widetilde{O}(f)$ means $O(f \log^k f)$ and $\widetilde{\Omega}(f)$ means $\Omega(f/\log^k f)$ for some constant $k > 0$. PPT stands for probabilistic polynomial time and can be either in the uniform or non-uniform model of computation.

1.5.2 Information theory

Definition 1.5.1 (entropies)**.** *For a random variable X and $x \in \mathrm{Supp}(X)$, the* sample entropy *(also called surprise) of x is*

$$\mathrm{H}^*_x(X) \stackrel{\text{def}}{=} \log \frac{1}{\Pr[X = x]}\,.$$

The (Shannon) entropy of X is

$$\mathrm{H}_{\mathsf{Sh}}(X) \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{x \stackrel{r}{\leftarrow} X}\Big[\mathrm{H}^*_x(X)\Big] = \mathop{\mathbb{E}}_{x \stackrel{r}{\leftarrow} X}\left[\log \frac{1}{\Pr[X = x]}\right].$$

The min-entropy of X is

$$\mathrm{H}_{\mathsf{min}}(X) \stackrel{\text{def}}{=} \min_{x \in \mathrm{Supp}(X)} \mathrm{H}^*_x(X) = \min_{x \in \mathrm{Supp}(X)} \log \frac{1}{\Pr[X = x]}\,.$$

The max-entropy of X is

$$\mathrm{H}_{\mathsf{max}}(X) \stackrel{\text{def}}{=} \log \Big|\mathrm{Supp}(X)\Big| \leq \max_{x \in \mathrm{Supp}(X)} \mathrm{H}^*_x(X)\,,$$

where $\mathrm{Supp}(X) = \{x : \Pr[X = x] > 0\}$.

Definition 1.5.2 (conditional (average) min-entropy [DORS08])**.** *Let (X, Y) be jointed distributed random variables. The average min-entropy of X conditioned on Y is*

$$\mathrm{H}_{\mathsf{min}}(X|Y) \stackrel{\text{def}}{=} \log \frac{1}{\mathbb{E}_{y \leftarrow Y}\left[\max_x \Pr[X = x | Y = y]\right]}$$

Proposition 1.5.3 (chain rule for entropy)**.** *Let (A, X) be a pair of random variables, then* $\mathrm{H}(A, X) = \mathrm{H}(A|X) + \mathrm{H}(X)$ *and for $(a, x) \in \mathrm{Supp}(A, X)$,* $\mathrm{H}^*_{a,x}(A, X) = \mathrm{H}^*_{a,x}(A|X) + \mathrm{H}^*_x(X)$.

Definition 1.5.4 (KL-divergences)**.** *For distributions X and Y on $\mathcal{X}$, and $x \in \mathrm{Supp}(Y)$, the* sample KL-divergence *(log-probability ratio) is*

$$\mathrm{D}^*_x(X \,\|\, Y) \stackrel{\text{def}}{=} \log \frac{\Pr[X = x]}{\Pr[Y = x]}\,.$$

The KL-divergence (also called relative entropy) from X to Y is

$$\mathrm{D}_{\mathsf{KL}}(X \,\|\, Y) \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{x \xleftarrow{r} X}[\mathrm{D}^*_x(X \,\|\, Y)] = \mathop{\mathbb{E}}_{x \xleftarrow{r} X}\left[\log \frac{\Pr[X=x]}{\Pr[Y=x]}\right].$$

The max divergence (also called max-relative entropy) between X and Y is

$$\mathrm{D}_{\mathsf{max}}(X \,\|\, Y) \stackrel{\text{def}}{=} \max_{x \in \mathrm{Supp}(X)} \mathrm{D}^*_x(X \,\|\, Y) = \max_{x \in \mathrm{Supp}(X)} \log \frac{\Pr[X=x]}{\Pr[Y=x]}.$$

Proposition 1.5.5. *Let X be a distribution on $\{0,1\}^n$. Then*

$$\mathrm{H}_{\mathsf{Sh}}(X) = n - \mathrm{D}_{\mathsf{KL}}(X \,\|\, U_n)$$

$$\mathrm{H}_{\mathsf{min}}(X) = n - \mathrm{D}_{\mathsf{max}}(X \,\|\, U_n)$$

1.5.3 Cryptography

Definition 1.5.6 (statistical distance)**.** *Let Random variable X_1 and X_2 be two random variables on $\mathcal{X}$. The statistical distance (a.k.a. total variation) is*

$$\mathrm{d}_{\mathsf{TV}}(X_1\,,\,X_2) \stackrel{\text{def}}{=} \max_{T \subseteq \mathcal{X}} \Big|\Pr[X_1 \in T] - \Pr[X_2 \in T]\Big|.$$

*We also say X_1 and X_2 are ε-*close *if* $\mathrm{d}_{\mathsf{TV}}(X_1\,,\,X_2) \le \varepsilon$.

Definition 1.5.7 (computational indistinguishability [GM84])**.** *Let X and Y be distributions over $\{0,1\}^n$, and $f : \{0,1\}^n \to \{0,1\}$ be a distinguisher. We say X and Y are ε-(computationally) indistinguishable by f if*

$$\Big|\Pr[f(X)=1] - \Pr[f(Y)=1]\Big| \le \varepsilon.$$

More generally, X and Y are ε-indistinguishable by a family of distinguishers $\mathcal{F}$ if X and Y are ε-indistinguishable by f for all $f \in \mathcal{F}$.

One common choice for the family of distinguishers is all distinguishers with bounded circuit complexity. We say X and Y are (t,ε)-indistinguishable if X and Y are ε-indistinguishable by all distinguishers of size t.

In the asymptotic setting, let n be a security parameter, we say X and Y are com-

putationally indistinguishable if for some $t(n) = n^{\omega(1)}$ *and* $\varepsilon(n) = n^{-\omega(1)}$*,* X *and* Y *are* $(t(n), \varepsilon(n))$*-indistinguishable.*

Definition 1.5.8 (one-way function)**.** *Let* n *be a security parameter,* $t = t(n)$ *and* $\varepsilon = \varepsilon(n)$*. A function* $f : \{0,1\}^n \rightarrow \{0,1\}^n$ *is a* (t, ε)-one-way function *if:*

1. *For all time* t *randomized algorithm* A*,* $\Pr_{x \overset{r}{\leftarrow} U_n}\left[\mathsf{A}\big(f(x)\big) \in f^{-1}\big(f(x)\big)\right] \leq \varepsilon$*, where* U_n *is uniform over* $\{0,1\}^n$*.*

2. *There exists a* PPT *algorithm* B *such that* $\mathsf{B}(x, 1^n) = f(x)$ *for all* $x \in \{0,1\}^n$*.*

If f *is* $(n^c, 1/n^c)$*-one-way for every* $c \in \mathbb{N}$*, we say that* f *is* (strongly) one-way*.*

Chapter 2

Unifying Computational Entropies via Kullback–Leibler Divergence

In this chapter, we introduce a new notion of hardness called *KL-hardness* for search problems which on the one hand is satisfied by all one-way functions and on the other hand implies both next-block HILL entropy [HRV13] and inaccessible entropy [HRVW09, HV17]. Two forms of computational entropy are used in the state-of-the-art constructions of pseudorandom generators [VZ12] and statistically-hiding commitment schemes [HRVW19], respectively.

2.1 Introduction

2.1.1 One-way functions and computational entropy

One-way functions [DH76] are on one hand the minimal assumption for complexity-based cryptography [IL89], but on the other hand can be used to construct a remarkable array of cryptographic primitives, including such powerful objects as CCA-secure symmetric encryption, zero-knowledge proofs and statistical zero-knowledge arguments for all of **NP**, and secure multiparty computation with an honest majority [GGM86, GMW91, GMW87, HILL99, Rom90, Nao91, HNO^{+}09]. All of these constructions begin by converting the "raw hardness" of a one-way function (OWF) to one of the following more structured cryptographic primitives: a pseudorandom generator (PRG) [BM82, Yao82], a universal one-way hash function (UOWHF) [NY89], or a statistically hiding commitment scheme (SHC) [BCC88].

The original constructions of these three primitives from arbitrary one-way functions [HILL99, Rom90, HNO$^+$09] were all very complicated and inefficient. Over the past decade, there has been a series of simplifications and efficiency improvements to these constructions [HRVW09, HRV13, HHR$^+$10, VZ12], leading to a situation where the constructions of two of these primitives — PRGs and SHCs — share a very similar structure and seem "dual" to each other. Specifically, these constructions proceed as follows:

1. Show that every OWF $f : \{0,1\}^n \to \{0,1\}^n$ has a gap between its "real entropy" and an appropriate form of "computational entropy". Specifically, for constructing PRGs, it is shown that the function $\mathsf{G}(x) = (f(x), x_1, x_2, \ldots, x_n)$ has "next-block HILL entropy" at least $n + \omega(\log n)$ while its real entropy is $\mathrm{H}_{\mathsf{Sh}}(\mathsf{G}(U_n)) = n$ [VZ12]. For constructing SHCs, it is shown that the function $\mathsf{G}(x) = (f(x)_1, \ldots, f(x)_n, x)$ has "next-block accessible entropy" at most $n - \omega(\log n)$ while its real entropy is again $\mathrm{H}(\mathsf{G}(U_n)) = n$ [HRVW09]. Note that the differences between the two cases are whether we break x or $f(x)$ into individual bits (which matters because the "next-block" notions of computational entropy depend on the block structure) and whether the form of computational entropy is larger or smaller than the real entropy.

2. An "entropy equalization" step that converts G into a similar generator where the real entropy in each block conditioned on the prefix before it is known. This step is exactly the same in both constructions.

3. A "flattening" step that converts the (real and computational) Shannon entropy guarantees of the generator into ones on (smoothed) min-entropy and max-entropy. This step is again exactly the same in both constructions.

4. A "hashing" step where high (real or computational) min-entropy is converted to uniform (pseudo)randomness and low (real or computational) max-entropy is converted to a small-support or disjointness property. For PRGs, this step only requires randomness

extractors [HILL99, NZ96], while for SHCs it requires (information-theoretic) interactive hashing [NOVY98, DHRS07]. (Constructing full-fledged SHCs in this step also utilizes UOWHFs, which can be constructed from one-way functions [Rom90]. Without UOWFHs, we obtain a weaker binding property, which nevertheless suffices for constructing statistical zero-knowledge arguments for all of **NP**.)

This common construction template came about through a back-and-forth exchange of ideas between the two lines of work. Indeed, the uses of computational entropy notions, flattening, and hashing originate with PRGs [HILL99], whereas the ideas of using next-block notions, obtaining them from breaking $(f(x), x)$ into short blocks, and entropy equalization originate with SHCs [HRVW09]. All this leads to a feeling that the two constructions, and their underlying computational entropy notions, are “dual” to each other and should be connected at a formal level.

In this paper, we make progress on this project of unifying the notions of computational entropy, by introducing a new computational entropy notion that yields both next-block HILL entropy and next-block accessible entropy in a clean and modular fashion. It is inspired by the proof of [VZ12] that $(f(x), x_1, \dots, x_n)$ has next-block HILL entropy $n + \omega(\log n)$, which we will describe now.

2.1.2 Next-block HILL entropy from OWF

First, we review the definitions of next-block HILL entropy and next-block accessible entropy, and how they are obtained from OWFs [VZ12, HRVW09]. For succinctness, we use the notation $z_{<i} = (z_1, \dots, z_i)$.

Definition 2.1.1 (next-block HILL entropy [HRV10], informal)**.** *Let n be a security parameter, and $Z = (Z_1, \dots, Z_m)$ be a random variable distributed on strings of length* poly(n). *We say that X has* next-block HILL entropy *at least k if there is a random variable $\widetilde{Z} = (\widetilde{Z}_1, \dots, \widetilde{Z}_m)$, jointly distributed with X, such that*

1. *For all $i \in [m]$, $(Z_{<i}, Z_i)$ is computationally indistinguishable from $(Z_{<i}, \widetilde{Z}_i)$.*

2. $\sum_{i=1}^{m} \mathrm{H}_{\mathsf{Sh}}(\widetilde{Z}_i | Z_{<i}) \geq k$.

It was conjectured in [HRV10] and proven in [VZ12] that next-block HILL entropy can be obtained from any OWF by breaking its input into bits:

Theorem 2.1.2 ([VZ12], informal)**.** *Let $f : \{0,1\}^n \to \{0,1\}^n$ be a one-way function, X be uniformly distributed on $\{0,1\}^n$, and $X = (X_1, \ldots, X_m)$ be a partition of X into m blocks where block lengths are $O(\log n)$. Then $(f(X), X_1, \ldots, X_m)$ has next-block HILL entropy at least $n + \omega(\log n)$.*

The intuition behind Theorem 2.1.2 is that since X is hard to sample given $f(X)$, then it should have some extra computational entropy given $f(X)$. This intuition is formalized using the following notion of being "hard to simulate":[1]

Definition 2.1.3 (KL-hard for simulating)**.** *Let n be a security parameter, and (Y, X) be a pair of random variables, jointly distributed over strings of length* poly(n)*. We say that X is* Δ-KL-hard for simulating *given Y if for all probabilistic polynomial-time S, we have*

$$\mathrm{D}_{\mathsf{KL}}(Y, X \,\|\, Y, \mathsf{S}(Y)) \geq \Delta\,.$$

That is, it is hard for any efficient adversary S to simulate the conditional distribution of X given Y, even approximately.

The first step of the proof of Theorem 2.1.2 is to show that X is $\omega(\log n)$-KL-hard for simulating given $f(X)$. Next, X is broken into short blocks ($X = (X_1, \ldots, X_m)$), and the sum of "next-block KL-hardness for simulating" is preserved. That is, for all $i \in [n]$ X_i is Δ_i-KL-hard to sample given $f(X), X_{<i}$, where $\sum_i \Delta_i = \omega(\log n)$. Finally, they showed that the KL-hardness of X for simulating is equivalent to the gap between conditional HILL entropy and real conditional entropy when the length of X is $O(\log n)$.

We remark that breaking X into short blocks is necessary for showing the equivalence between the KL-hardness for simulating and conditional HILL entropy. Indeed, one cannot

[1] In [VZ12], it was called "KL-hard for sampling". Here we emphasize that the algorithm is simulating the randomness used by the one-way function.

expect any X' with $\mathrm{H}_{\mathsf{Sh}}(X'|f(X))$ noticeably larger than $\mathrm{H}_{\mathsf{Sh}}(X|f(X))$ and $(X', f(X))$ is computationally indistinguishable from $(X, f(X))$. An algorithm can easily distinguish them by checking whether f maps the first block to the second block.

2.1.3 Next-block accessible entropy from OWF

We say generator $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_1, \ldots, \widetilde{\mathsf{G}}_m)$ which takes a sequence of uniformly random string $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$ is online if for all $i \in [m]$, $\widetilde{\mathsf{G}}_i(\widetilde{R}) = \widetilde{\mathsf{G}}_i(\widetilde{R}_{\leq i})$ only depends on the first i random strings of $\widetilde{R}$.

Definition 2.1.4 (next-block inaccessible entropy, informal)**.** *Let n be a security parameter, and $Z = (Z_1, \ldots, Z_m)$ be a random variable distributed on strings of length* $\mathrm{poly}(n)$*. We say that Z has* next-block accessible entropy *at most k if for all online generator $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_1, \ldots, \widetilde{\mathsf{G}}_m)$ such that* $\mathrm{Supp}(\widetilde{\mathsf{G}}(\widetilde{R})) \subseteq \mathrm{Supp}(Z)$*, we have*

$$\sum_{i=1}^{m} \mathrm{H}_{\mathsf{Sh}}\left(\widetilde{\mathsf{G}}_i(\widetilde{R}_{\leq i}) \;\middle|\; \widetilde{R}_{<i}\right) \leq k\,,$$

where $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$ is uniformly distributed.

The accessible entropy adversary $\widetilde{\mathsf{G}}$ is trying to *generate* the random variables Z_i conditioned on the history rather than recognize them. Note that we condition on not only the previous blocks $(\widetilde{Z}_1, \ldots, \widetilde{Z}_{i-1})$, but also the coin tosses $(\widetilde{R}_1, \ldots, \widetilde{R}_{i-1})$ used previously.

Similarly to next-block HILL entropy (Theorem 2.1.2), it is known that one-wayness implies next-block inaccessible entropy.

Theorem 2.1.5 ([HRVW09])**.** *Let $f : \{0,1\}^n \to \{0,1\}^n$ be a one-way function, X be uniformly distributed in $\{0,1\}^n$, and $(Y_1, \ldots, Y_m)$ be a partition of $Y = f(X)$ into blocks of length $O(\log n)$. Then $(Y_1, \ldots, Y_m, X)$ has next-block accessible entropy at most $n - \omega(\log n)$.*

Unfortunately, the existing proof of Theorem 2.1.5 is not modular like that of Theorem 2.1.2 outlined above. In particular, it does not isolate the step of relating one-wayness to an entropic hardness or the significance of having short blocks. Another unsatisfactory aspect is that when the random variable Z is not uniform on its support, there can be an adversary $\widetilde{\mathsf{G}}$ achieving accessible entropy even *higher* than $\mathrm{H}_{\mathsf{Sh}}(Z)$, for example by making $\widetilde{Z}$ uniform on $\mathrm{Supp}(Z)$.

2.1.4 Our unified notion — KL-hardness

We remedy the above issues by proposing a new, more general notion of KL-hardness. This notion provides a unified way to capture the hardness inside the jointly distributed random variables $\mathsf{G}(X) = (f(X), X) = (Y, X)$, which allows us to obtain both next-block HILL entropy and next-block inaccessible entropy.

In KL-hardness for simulating, the hardness is characterized by the KL divergence from the true distribution (Y, X) to $(Y, \mathsf{S}(Y))$, which produced by "simulator" S. On the other hand, for next-block accessible entropy, we would like to capture the hardness in approximating the joint distribution (Y, X). That is, we ask a generator $\widetilde{\mathsf{G}}$ to output $\big(\widetilde{Y}, \widetilde{X}\big) = \widetilde{\mathsf{G}}(\widetilde{R}) = \big(\widetilde{\mathsf{G}}_Y(\widetilde{R}), \widetilde{\mathsf{G}}_X(\widetilde{R})\big)$ such that $\mathrm{Supp}\big((\widetilde{Y}, \widetilde{X})\big) \subseteq \mathrm{Supp}\big((Y, X)\big)$ and $\big(\widetilde{Y}, \widetilde{X}\big)$ is "close" to (Y, X). We combine both ideas by looking at the "distance" between distributions inferred from $\widetilde{\mathsf{G}}$ and S. Similar to the definition of accessible entropy, where the measurement of accessible entropy is conditioned on the coin tosses of the generator, we ask S to output the randomness of $\widetilde{\mathsf{G}}$ instead of X (which can be seen as the randomness of G where $\mathsf{G}(x) = (f(x), x)$ as above). In our KL-hardness definition, the adversary gets to choose both $\widetilde{\mathsf{G}}$ and S to reduce the KL-divergence.

Definition 2.1.6 (KL-hard, informal version of Definition 2.3.4)**.** *Let n be a security parameter, and (Y, X) be a pair of random variables jointly distributed over strings of length* $\mathrm{poly}(n)$*. We say that (Y, X) is* Δ-KL-hard *if for all probabilistic* $\mathrm{poly}(n)$*-time algorithms* $\widetilde{\mathsf{G}} = \big(\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_x\big)$ *and* S *such that* $\mathrm{Supp}\big(\widetilde{\mathsf{G}}(\widetilde{R})\big) \subseteq \mathrm{Supp}\big((Y, X)\big)$*, we have*

$$\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{R} \,\Big\|\, Y, \mathsf{S}(Y)\Big) \geq \Delta\,,$$

where $\widetilde{R}$ denotes uniformly distributed coin tosses for $\widetilde{\mathsf{G}}$.

Similar to KL-hardness for simulating, one can show that if $f : \{0,1\}^n \to \{0,1\}^n$ is a OWF, then $(f(X), X)$ is $\omega(n)$-KL-hard (Theorem 2.3.5) with a one-line calculation.

The KL-hardness measures on both how well $\widetilde{\mathsf{G}}_y$ approximates the distribution of Y and how well S simulates the corresponding coin tosses of $\widetilde{\mathsf{G}}_y$. Potentially, there is a trade-off between the two criteria. In fact, we will see that one of each hardness leads to HILL entropy or accessible entropy.

First, to focus on the hardness in the simulator S, we can fix $\widetilde{\mathsf{G}}(\widetilde{R}) = \mathsf{G}(\widetilde{R})$ to be "honest" by mimicking G. Then the definition reduces to KL-hardness for simulating (Definition 2.1.3). Thus, (Y, X) being Δ-KL-hard implies X is Δ-KL-hard for simulating given Y, and then the gap between next-block HILL entropy and true entropy follows as [VZ12] have shown.

On the other hand, to focus on the hardness in the generator $\widetilde{\mathsf{G}}$, a natural simulator S on input y simply keeps guessing the coin tosses r for $\widetilde{\mathsf{G}}$ until $\widetilde{\mathsf{G}}_y(r) = y$, and outputs r. The simulator outputs $\perp$ if it fails to find correct coin tosses. It can be shown that $\mathrm{D}_{\mathsf{KL}}(\widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{R} \,\|\, Y, \mathsf{S}(Y)) \approx \mathrm{D}_{\mathsf{KL}}(\widetilde{\mathsf{G}}_y(\widetilde{R}) \,\|\, Y)$ if $\mathsf{S}(Y)$ outputs $\perp$ with small probability.

Unfortunately, the probability that the simulator succeeds can be exponentially small in general. Therefore, we break Y into short blocks $Y = (Y_1, \ldots, Y_m)$ and consider $\widetilde{\mathsf{G}}$ to be an online generator as in the definition of accessible entropy. Then we can obtain a simulator for $\widetilde{\mathsf{G}}$ in an "online fashion" as well: it guesses the coin toss $\widetilde{r}_i$ one at a time by matching $\widetilde{\mathsf{G}}_i(\widetilde{r}_{\leq i})$ to $\widetilde{Y}_i$, so it only fails with negligible probability in each step. As before, if it fails in guessing within polynomial trials, it outputs $\perp$ for all remaining $\widetilde{r}_i$s (See Algorithm 2.4.1 for the formal definition of the simulator). We denote such a simulator as $\mathsf{Sim}^{\widetilde{\mathsf{G}}}$ and define *next-block KL-hardness for generating* as follows. (Note that the adversary can only choose $\widetilde{\mathsf{G}}$ but $\mathsf{Sim}^{\widetilde{\mathsf{G}}}(Y)$ also depends on $\widetilde{\mathsf{G}}$.)

Definition 2.1.7 (next-block KL-hard for generating, informal version of Definition 2.4.1)**.** *Let n be a security parameter, and $(Y_1, \ldots, Y_m, X)$ be a random variables, jointly distributed over strings of length* $\mathrm{poly}(n)$*. We say that $(Y_1, \ldots, Y_m, X)$ is* next-block Δ-KL-hard for generating *if for all probabilistic polynomial-time online generator $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_{y_1}, \ldots, \widetilde{\mathsf{G}}_{y_m}, \widetilde{\mathsf{G}}_x)$ such that* $\mathrm{Supp}(\widetilde{\mathsf{G}}) \subseteq \mathrm{Supp}((Y_1, \cdots, Y_m, X))$*, we have*

$$\mathrm{D}_{\mathsf{KL}}\left(\widetilde{Y}_1, \ldots, \widetilde{Y}_m, \widetilde{R} \,\middle\|\, Y_1, \ldots, Y_m, \mathsf{Sim}^{\widetilde{\mathsf{G}}}(Y)\right) \geq \Delta\,,$$

where $\widetilde{R}$ is uniformly distributed and $\widetilde{Y}_i = \widetilde{\mathsf{G}}_{y_i}(\widetilde{R}_{\leq i})$ for all $i \in [m]$.

It can be shown that if (Y, X) is Δ-KL-hard and we break $Y = (Y_1, \ldots, Y_m)$ into blocks of length $O(\log n)$, then $(Y_1, \ldots, Y_m, X)$ is next-block Δ-KL-hard for generating. Once we have the next-block KL-hardness for generating, the next step is to deduce the next-block inaccessible entropy among $(Y_1, \ldots, Y_m, X)$ from "next-block KL-hardness for generating". In fact, we obtain the slightly more general notion, which we call "next-block inaccessible relative entropy":

Definition 2.1.8 ((next-block) inaccessible relative entropy, informal version of Definition 2.4.4)**.** *Let n be a security parameter, and $Z = (Z_1, \ldots, Z_m)$ be a random variable distributed on strings of length* $\mathrm{poly}(n)$. *We say that Z has* next-block-inaccessible relative entropy *at least Δ if for all probabilistic polynomial-time online generator $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_{z_1}, \ldots, \widetilde{\mathsf{G}}_{z_m})$ such that* $\mathrm{Supp}(\widetilde{\mathsf{G}}) \subseteq \mathrm{Supp}(Z)$*, we have*

$$\sum_{i=1}^{m} \mathrm{D}_{\mathsf{KL}}\left(\widetilde{Z}_i | \widetilde{R}_{<i}, \widetilde{Z}_{<i} \,\middle\|\, Z_i | R_{<i}, Z_{<i}\right) \geq \Delta\,,$$

where $\widetilde{R}$ is uniformly distributed, $\widetilde{Z}_i = \widetilde{\mathsf{G}}_{z_i}(\widetilde{R}_{\leq i})$ and $R = (R_1, \ldots, R_m)$ is a dummy random variable independent of Z.

A nice property of the definition of next-block inaccessible relative entropy compared to next-block inaccessible entropy is that it is meaningful even for non-flat random variables, as KL-divergence is always nonnegative. Moreover, for flat random variables (which is the case for $Z = (f(X), X)$), both definitions are equivalent. Intuitively, this is an analogue of the equality $\mathrm{H}_{\mathsf{Sh}}(\widetilde{Z}) = \mathrm{H}_{\mathsf{Sh}}(Z) - \mathrm{D}_{\mathsf{KL}}(\widetilde{Z} \,\|\, Z)$ when Z is flat and $\mathrm{Supp}(\widetilde{Z}) \subseteq \mathrm{Supp}(Z)$.

With these new notions, we obtain a new, more modular proof of Theorem 2.1.5, which outlined as:

$$\begin{aligned}
& f \text{ is a one-way function} \\
\Rightarrow\ & (f(X), X) \text{ is } \omega(\log n)\text{-KL-hard} \\
\Rightarrow\ & (f(X)_1, \ldots, f(X)_n, X) \text{ is next-block } \omega(\log n)\text{-KL-hard for generating} \\
\Rightarrow\ & (f(X)_1, \ldots, f(X)_n, X) \text{ has next-block inaccessible (relative) entropy at least } \omega(\log n)\,.
\end{aligned}$$

The reduction implicit in the second-to-last step is the same as the one in [HRVW09], but the analysis is different (In particular, [HRVW09] makes no use of KL-divergence.). Similar to the existing proof of Theorem 2.1.2 in [VZ12], this proof separates the move from one-wayness to a form of KL-hardness, the role of short blocks, and the move from KL-hardness to computational entropy. Moreover, this further illumination of and toolkit for notions of computational entropy may open the door to other applications in cryptography.

2.2 Preliminaries

2.2.1 Information theory.

Definition 2.2.1 (Conditional KL-divergence)**.** *For pairs of random variables* (A, X) *and* (B, Y)*, and* $(a, x) \in \mathrm{Supp}(A, X)$*, the* conditional sample KL-divergence *is:*

$$\mathrm{D}^*_{a,x}(A|X \,\|\, B|Y) \stackrel{\text{def}}{=} \log \frac{\Pr[A = a|X = x]}{\Pr[B = a|Y = x]} \,,$$

and the conditional KL-divergence *is:*

$$\mathrm{D}_{\mathsf{KL}}(A|X \,\|\, B|Y) \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{(a,x) \stackrel{r}{\leftarrow} (A,X)} \left[\log \frac{\Pr[A = a|X = x]}{\Pr[B = a|Y = x]} \right] .$$

Definition 2.2.2 (smooth min*-KL-divergence)**.** *For distributions* X *and* Y*, the min*-divergence*[2] *between* X *and* Y *is*

$$\mathrm{D}_{\mathsf{min}^*}(X \,\|\, Y) \stackrel{\text{def}}{=} \min_{x \in \mathrm{Supp}(Y)} \mathrm{D}^*_x(X \,\|\, Y) = \min_{x \in \mathrm{Supp}(Y)} \log \frac{\Pr[X = x]}{\Pr[Y = x]} \,.$$

For $\delta \in [0, 1]$*, we define the* δ*-smooth min*-divergence from* X *to* Y*,* $\mathrm{D}^\delta_{\mathsf{min}^*}(X \,\|\, Y)$ *to be the quantile of level* δ *of* $\mathrm{D}^*_x(X \,\|\, Y)$*. Equivalently it is the smallest* $\Delta \in \mathbb{R}$ *satisfying*

$$\Pr_{x \stackrel{r}{\leftarrow} X} \left[\mathrm{D}^*_x(X \,\|\, Y) \leq \Delta \right] \geq \delta \,,$$

[2] This is not the standard Rényi divergence D_α with $\alpha = 0$. We use min* to emphasize the difference and indicate that the minimum is taken over sample notions.

and it is characterized by

$$\mathrm{D}^{\delta}_{\mathsf{min}^*}(X \,\|\, Y) > \Delta \Leftrightarrow \Pr_{x \overset{r}{\leftarrow} X}\left[\mathrm{D}^*_x(X \,\|\, Y) \le \Delta\right] < \delta\,.$$

Proposition 2.2.3 (chain rule for KL-divergence)**.** *For pairs of random variables* (X, A) *and* (Y, B)*,*

$$\mathrm{D}_{\mathsf{KL}}(A, X \,\|\, B, Y) = \mathrm{D}_{\mathsf{KL}}(A|X \,\|\, B|Y) + \mathrm{D}_{\mathsf{KL}}(X \,\|\, Y)\,.$$

For $(a, x) \in \mathrm{Supp}(A, X)$*,*

$$\mathrm{D}^*_{a,x}(A, X \,\|\, B, Y) = \mathrm{D}^*_{a,x}(A|X \,\|\, B|Y) + \mathrm{D}^*_x(X \,\|\, Y)\,.$$

Proposition 2.2.4 (data-processing inequality)**.** *Let* (X, Y) *be a pair of random variables and let* f *be a function defined on* $\mathrm{Supp}(Y)$*, then:*

$$\mathrm{D}_{\mathsf{KL}}(X \,\|\, Y) \ge \mathrm{D}_{\mathsf{KL}}(f(X) \,\|\, f(Y))\,.$$

2.2.2 Block generators.

Definition 2.2.5 (block generator)**.** *An* m-block generator *is a function* $\mathsf{G} : \{0,1\}^s \to \prod_{i=1}^m \{0,1\}^{\ell_i}$*.* $\mathsf{G}_i(r)$ *denotes the* i*-th block of* G *on input* r *and* $|\,\mathsf{G}_i\,| = \ell_i$ *denotes the bit length of the* i*-th block.*

Definition 2.2.6 (online block generator)**.** *An* online m-block generator *is a function* $\widetilde{\mathsf{G}} : \prod_{i=1}^m \{0,1\}^{s_i} \to \prod_{i=1}^m \{0,1\}^{\ell_i}$ *such that for all* $i \in [m]$ *and* $r \in \prod_{i=1}^m \{0,1\}^{s_i}$*,* $\widetilde{\mathsf{G}}_i(r)$ *only depends on* $r_{\le i}$*. We sometimes write* $\widetilde{\mathsf{G}}_i(r_{\le i})$ *when the input blocks* $i+1, \ldots, m$ *are unspecified.*

Definition 2.2.7 (support)**.** *The* support *of a generator* G *is the support of the random variable* $\mathrm{Supp}(\mathsf{G}(R))$ *for uniform input* R*. If* $(\mathsf{G}_y, \mathsf{G}_w)$ *is an online block generator, and* Π *is a binary relation, we say that* $(\mathsf{G}_y, \mathsf{G}_w)$ *is* supported on Π *if* $\mathrm{Supp}(\mathsf{G}_y(R), \mathsf{G}_w(R)) \subseteq \Pi$.

The subscripts we use for a block generator often match the random variables they correspond to.

2.3 Search Problems and KL-hardness

In this section, we first present the classical notion of hard-on-average search problems and introduce the new notion of KL-hardness. We then relate the two notions by proving that average-case hardness implies KL-hardness.

2.3.1 Search problems

For a binary relation $\Pi \subseteq \{0,1\}^* \times \{0,1\}^*$, we write $\Pi(y, w)$ for the predicate that is true iff $(y, w) \in \Pi$ and say that w is a *witness* for the *instance* y.[3] To each relation Π, we naturally associate (1) a *search problem*: given y, find w such that $\Pi(y, w)$ or state that no such w exist and (2) the *decision problem* defined by the language $L_\Pi \overset{\text{def}}{=} \{y \in \{0,1\}^* : \exists w \in \{0,1\}^*, \Pi(y, w)\}$. **FNP** denotes the set of all relations Π computable by a polynomial time algorithm and such that there exists a polynomial p such that $\Pi(y, w) \Rightarrow |w| \leq p(|y|)$. Whenever $\Pi \in \textbf{FNP}$, the associated decision problem L_Π is in **NP**. We now define average-case hardness.

Definition 2.3.1 (distributional search problem)**.** *A* distributional search problem *is a pair* (Π, Y) *where* $\Pi \subseteq \{0,1\}^* \times \{0,1\}^*$ *is a binary relation and* Y *is a random variable supported on* L_Π*.*

The problem (Π, Y) *is* $(\mathrm{t}, \varepsilon)$-hard *if* $\Pr\left[\Pi(Y, \mathsf{A}(Y))\right] \leq \varepsilon$ *for all time* t *randomized algorithm* A*, where the probability is over the distribution of* Y *and the randomness of* A*.*

Example 2.3.2. *For* $f : \{0,1\}^n \to \{0,1\}^n$*, the problem of inverting* f *is the search problem associated with the relation* $\Pi^f \overset{\text{def}}{=} \{(f(x), x) : x \in \{0,1\}^n\}$*. If* f *is a* (t, ε)*-one-way function, then the distributional search problem* $\left(\Pi^f, f(X)\right)$ *of inverting* f *on a uniform random input* $X \in \{0,1\}^n$ *is* (t, ε)*-hard.*

Remark 2.3.3. *Consider a distributional search problem* (Π, Y)*. Without loss of generality, there exists a (possibly inefficient) two-block generator* $\mathsf{G} = (\mathsf{G}_y, \mathsf{G}_w)$ *supported on* Π *such that*

[3] We used the unconventional notation y for the instance (instead of x) because our relations will often be of the form Π^f for some function f; in this case an instance is some y in the range of f and a witness for y is any preimage $x \in f^{-1}(y)$.

$\mathsf{G}_y(R) = Y$ *for uniform input* R. *If* G_w *is polynomial-time computable, it is easy to see that the search problem* $(\Pi^{\mathsf{G}_y}, \mathsf{G}_y(R))$ *is at least as hard as* (Π, Y). *The advantage of writing the problem in this "functional" form is that the distribution* $(\mathsf{G}_1(R), R)$ *over (instance, witness) pairs is flat, which is a necessary condition to relate hardness to inaccessible entropy.*

Furthermore, if G_y *is also polynomial-time computable and* (Π, Y) *is* $(\mathrm{poly}(n), \mathrm{negl}(n))$*-hard, then* $R \mapsto \mathsf{G}_y(R)$ *is a one-way function. Combined with the previous example, we see that the existence of one-way functions is equivalent to the existence of* $(\mathrm{poly}(n), \mathrm{negl}(n))$*-hard search problems for which (instance, witness) pairs can be efficiently sampled.*

2.3.2 KL-hardness

Instead of considering an adversary directly attempting to solve a search problem (Π, Y), the adversary in the definition of KL-hardness comprises a pair of algorithm $(\widetilde{\mathsf{G}}, \mathsf{S})$ where $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ is a two-block generator outputting valid (instance, witness) pairs for Π and S is a *simulator* for $\widetilde{\mathsf{G}}$: given an instance y, the goal of S is to output randomness r for $\widetilde{\mathsf{G}}$ such that $\widetilde{\mathsf{G}}_y(r) = y$. Formally, the definition is as follows.

Definition 2.3.4 (KL-hard)**.** *Let* (Π, Y) *be a distributional search problem. We say that* (Π, Y) *is* (t, Δ)-KL-hard *if*

$$\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y\Big) > \Delta \tag{2.1}$$

for all pairs $(\widetilde{\mathsf{G}}, \mathsf{S})$ *of time* t *algorithms where* $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ *is a two-block generator supported on* Π *and* $\widetilde{R}$ *is uniform randomness for* $\widetilde{\mathsf{G}}_y$. *Similarly,* (Π, Y) *is* (t, Δ)-$\mathrm{D}^{\delta}_{\mathsf{min}^*}$-hard *if for all such algorithm pairs:*

$$\mathrm{D}^{\delta}_{\mathsf{min}^*}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y\Big) > \Delta\,.$$

Note that a pair $(\widetilde{\mathsf{G}}, \mathsf{S})$ achieves a KL-divergence of zero in Equation (2.1) if $\widetilde{\mathsf{G}}_y(R)$ has the same distribution as Y and if $\widetilde{\mathsf{G}}_y(\mathsf{S}(y)) = y$ for all $y \in \mathrm{Supp}(Y)$. In this case, we have that $\widetilde{\mathsf{G}}_w(\mathsf{S}(Y))$ is a valid witness for Y since $\widetilde{\mathsf{G}}$ is supported on Π.

More generally, the composition $\widetilde{\mathsf{G}}_w \circ \mathsf{S}$ solves the search problem (Π, Y) whenever $\widetilde{\mathsf{G}}_y(\mathsf{S}(Y)) = Y$. When the KL-divergences in Equation (2.1) are upper-bounded, we can lower bound the probability of the search problem being solved (Lemma 2.3.7). This immediately implies that hard search problems are also KL-hard.

Theorem 2.3.5. *Let* (Π, Y) *be a distributional search problem. If* (Π, Y) *is* (t, ε)*-hard, then it is* (t', Δ')*-KL-hard and* (t', Δ'')*-*$\mathrm{D}^{\delta}_{\mathsf{min}^*}$*-hard for every* $\delta \in [0,1]$ *where* $t' = \Omega(t)$,$\Delta' = \log(1/\varepsilon)$ *and* $\Delta'' = \log(1/\varepsilon) - \log(1/\delta)$.

Remark 2.3.6. *As we see, a "good" simulator* S *for a generator* $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ *is one for which* $\widetilde{\mathsf{G}}_y(\mathsf{S}(Y)) = Y$ *holds often. It will be useful in Section 2.4 to consider simulators* S *which are allowed to fail by outputting a failure string* $r \notin \mathrm{Supp}(\widetilde{R})$*, (e.g.,* $r = \perp$*) and adopt the convention that* $\widetilde{\mathsf{G}}_y(r) = \perp$ *whenever* $r \notin \mathrm{Supp}(\widetilde{R})$*. With this convention, we can without loss of generality add the requirement that* $\widetilde{\mathsf{G}}_y(\mathsf{S}(y)) = y$ *whenever* $\mathsf{S}(y) \in \mathrm{Supp}(\widetilde{R})$*: indeed,* S *can always check that it is the case and if not output a failure symbol. For such a simulator* S*, observe that for all* $r \in \mathrm{Supp}(\widetilde{R})$*, the second variable on both sides of the KL-divergences in Definition 2.3.4 is obtained by applying* $\widetilde{\mathsf{G}}_y$ *on the first variable and can thus be dropped, leading to a more concise definition of KL-hardness:* $\mathrm{D}_{\mathsf{KL}}(\widetilde{R} \,\|\, \mathsf{S}(Y)) > \Delta$*.*

Theorem 2.3.5 is an immediate consequence of the following lemma.

Lemma 2.3.7. *Let* (Π, Y) *be a distributional search problem and* $(\widetilde{\mathsf{G}}, \mathsf{S})$ *be a pair of algorithms with* $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ *a two-block generator supported on* Π*. Define the linear-time oracle algorithm* $\mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(y) \stackrel{\text{def}}{=} \widetilde{\mathsf{G}}_w(\mathsf{S}(y))$*. For* $\Delta \in \mathbb{R}^+$ *and* $\delta \in [0,1]$*:*

1. *If* $\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y\Big) \leq \Delta$ *then* $\Pr\Big[\Pi(Y, \mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(Y))\Big] \geq 1/2^{\Delta}$.
2. *If* $\mathrm{D}^{\delta}_{\mathsf{min}^*}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y\Big) \leq \Delta$ *then* $\Pr\Big[\Pi(Y, \mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(Y))\Big] \geq \delta/2^{\Delta}$.

Proof. We have:

$$\begin{aligned}
\Pr\Big[\Pi(Y, \mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(Y))\Big] &= \Pr\Big[\Pi(Y, \widetilde{\mathsf{G}}_w(\mathsf{S}(Y)))\Big] \\
&\geq \Pr\Big[\widetilde{\mathsf{G}}_y(\mathsf{S}(Y)) = Y\Big] && (\widetilde{\mathsf{G}} \text{ is supported on } \Pi) \\
&= \sum_{r \in \mathrm{Supp}(\widetilde{R})} \Pr\Big[\mathsf{S}(Y) = r \,\wedge\, Y = \widetilde{\mathsf{G}}_y(r)\Big]
\end{aligned}$$

$$= \mathop{\mathbb{E}}_{r \xleftarrow{\mathrm{r}} \widetilde{R}} \left[\frac{\Pr\left[\mathsf{S}(Y) = r \,\wedge\, Y = \widetilde{\mathsf{G}}_y(r)\right]}{\Pr\left[\widetilde{R} = r\right]} \right]$$

$$= \mathop{\mathbb{E}}_{\substack{r \xleftarrow{\mathrm{r}} \widetilde{R} \\ y \leftarrow \widetilde{\mathsf{G}}_y(r)}} \left[2^{-\,\mathrm{D}^*_{r,y}\left(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\middle\|\, \mathsf{S}(Y), Y\right)} \right].$$

Now, the first claim follows by Jensen's inequality (since $x \mapsto 2^{-x}$ is convex) and the second claim follows by Markov' inequality when considering the event that the sample-KL is smaller than Δ (which occurs with probability at least δ by assumption). □

Relation to KL-hardness for simulating. In [VZ12][4], the authors introduced the notion of KL-hardness for simulating: for jointly distributed variables (Y, W), W is hard for simulating given Y if it is hard for a polynomial time adversary to approximate—measured in KL-divergence—the conditional distribution W given Y. Formally:

Definition 2.3.8 (KL-hard for simulating, Def. 3.4 in [VZ12])**.** *Let* (Y, W) *be a pair of random variables, we say that* W *is* (t, Δ)-KL-hard to sample given Y *if for all time* t *randomized algorithm* S*, we have:*

$$\mathrm{D}_{\mathsf{KL}}(Y, W \,\|\, Y, \mathsf{S}(Y)) > \Delta\,.$$

It was shown in [VZ12] that if $f : \{0,1\}^n \to \{0,1\}^n$ is a one-way function, then $\big(f(X), X_1, \ldots, X_n\big)$ has next-bit HILL entropy for uniform $X \in \{0,1\}^n$ (Theorem 2.1.2). The first step in proving this result was to prove that X is KL-hard to simulate given $f(X)$.

We observe that when (Y, W) is of the form $\big(f(X), X\big)$ for some function $f : \{0,1\}^n \to \{0,1\}^n$ and variable X over $\{0,1\}^n$, then KL-hardness for sampling is implied by KL-hardness by simply fixing $\widetilde{\mathsf{G}}$ to be the "honest simulator" $\widetilde{\mathsf{G}}(X) = (\widetilde{\mathsf{G}}_y(X), \widetilde{\mathsf{G}}_x(X)) = \big(f(X), X\big)$. Indeed, in this case we have:

$$\mathrm{D}_{\mathsf{KL}}\Big(X, \widetilde{\mathsf{G}}_y(X) \,\Big\|\, \mathsf{S}(Y), Y\Big) = \mathrm{D}_{\mathsf{KL}}\Big(X, f(X) \,\Big\|\, \mathsf{S}(Y), Y\Big)\,.$$

Corollary 2.3.9. *Consider a function* $f : \{0,1\}^n \to \{0,1\}^n$ *and define* $\Pi^f \overset{\mathrm{def}}{=} \{(f(x), x) :$

[4] In their work, they named it KL-hardness for sampling. We call it KL-hardness for simulating as S is more specifically simulating the conditional distribution of "randomness" W.

$x \in \{0,1\}^n\}$ and $Y \overset{\text{def}}{=} f(X)$ for X uniform over $\{0,1\}^n$. If f is (t, ε)-one-way, then (Π^f, Y) is $(t', \log(1/\varepsilon))$-KL-hard and X is $(t', \log(1/\varepsilon))$-KL-hard for sampling given Y with $t' = \Omega(t)$.

Witness KL-hardness. We also introduce a relaxed notion of KL-hardness called witness-KL-hardness. In this notion, we further require $(\widetilde{\mathsf{G}}, \mathsf{S})$ to approximate the joint distribution of (instance, witness) pairs rather than only instances. For example, the problem of inverting a function f over a random input X is naturally associated with the distribution $\big(f(X), X\big)$. The relaxation in this case is analogous to the notion of *distributional one-way function* for which the adversary is required to approximate the uniform distribution over preimages.

Definition 2.3.10 (witness KL-hardness)**.** *Let Π be a binary relation and (Y, W) be a pair of random variables supported on Π. We say that (Π, Y, W) is (t, Δ)-*witness-KL-hard *if for all pairs of time t algorithms $(\widetilde{\mathsf{G}}, \mathsf{S})$ where $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ is a two-block generator supported on Π, for uniform $\widetilde{R}$,*

$$\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y, W\Big) > \Delta\,.$$

*Similarly, for $\delta \in [0,1]$, (Π, Y, W) is (t, Δ)-*witness-$\mathrm{D}^{\delta}_{\mathsf{min}^*}$-hard*, if for all such pairs,*

$$\mathrm{D}^{\delta}_{\mathsf{min}^*}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}) \,\Big\|\, \mathsf{S}(Y), Y, W\Big) > \Delta\,.$$

We introduced KL-hardness first, since it is the notion which is most directly obtained from the hardness of distribution search problems. Observe that by the data processing inequality for KL divergence (Proposition 2.2.4), dropping the third variable on both sides of the KL divergences in Definition 2.3.10 only decreases the divergences. Hence, KL-hardness implies witness-KL-hardness as stated in Theorem 2.3.11. As we will see in Section 2.4 witness-KL-hardness is the "correct" notion to obtain inaccessible entropy from: it is in fact equal to inaccessible entropy up to $1/\operatorname{poly}$ losses.

Theorem 2.3.11. *Let Π be a binary relation and (Y, W) be a pair of random variables supported on Π. If (Π, Y) is (t, ε)-hard, then (Π, Y, W) is (t', Δ')-witness-KL-hard and (t', Δ'')-witness-$\mathrm{D}^{\delta}_{\mathsf{min}^*}$-hard for every $\delta \in [0,1]$ where $t' = \Omega(t)$, $\Delta' = \log(1/\varepsilon)$ and $\Delta'' =$*

$\log(1/\varepsilon) - \log(1/\delta)$.

We cannot actually get the parameter for $\mathrm{D}_{\mathsf{min}^*}$ in Theorem 2.3.11 simply by data processing inequality. That is, it does not follow with the claimed parameters in a black-box manner from Theorem 2.3.5. However, the proof essentially identical to the one for Theorem 2.3.5 gives the result.

Proof. Let $(\widetilde{\mathsf{G}}, \mathsf{S})$ be a pair of algorithms with $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w)$ a two-block generator supported on Π. Define the linear-time oracle algorithm $\mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(y) \stackrel{\text{def}}{=} \widetilde{\mathsf{G}}_w(\mathsf{S}(y))$. Then

$$\begin{aligned}
\Pr\Big[\Pi(Y, \mathsf{A}^{\widetilde{\mathsf{G}}_w, \mathsf{S}}(Y))\Big] &= \Pr\Big[\Pi(Y, \widetilde{\mathsf{G}}_w(\mathsf{S}(Y)))\Big] \\
&\geq \Pr[\widetilde{\mathsf{G}}_y(\mathsf{S}(Y)) = Y][2] && (\widetilde{\mathsf{G}} \text{ is supported on } \Pi) \\
&= \sum_{r \in \mathrm{Supp}(\widetilde{R})} \Pr\Big[\mathsf{S}(Y) = r \wedge Y = \widetilde{\mathsf{G}}_y(r)\Big] \\
&\geq \sum_{\substack{r \in \mathrm{Supp}(\widetilde{R}) \\ w \in \mathrm{Supp}(\widetilde{\mathsf{G}}_w(\widetilde{R}))}} \Pr\Big[\mathsf{S}(Y) = r \wedge Y = \widetilde{\mathsf{G}}_y(r) \wedge W = w\Big] \\
&= \mathop{\mathbb{E}}_{\substack{r \xleftarrow{\mathrm{r}} \widetilde{R} \\ w \leftarrow \widetilde{\mathsf{G}}_w(r)}} \left[\frac{\Pr\left[\mathsf{S}(Y) = r \wedge Y = \widetilde{\mathsf{G}}_y(r) \wedge W = w\right]}{\Pr\left[\widetilde{R} = r \wedge \widetilde{\mathsf{G}}_w(r) = w\right]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{r \xleftarrow{\mathrm{r}} \widetilde{R} \\ (y,w) \leftarrow \widetilde{\mathsf{G}}(r)}} \left[2^{-\mathrm{D}^*_{r,y,w}\left(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}) \,\middle\|\, \mathsf{S}(Y), Y, W\right)}\right],
\end{aligned}$$

The witness-KL-hardness then follows by applying Jensen's inequality (since $x \mapsto 2^{-x}$ is convex) and the witness-$\mathrm{D}_{\mathsf{min}^*}$-hardness follows by Markov's inequality by considering the event that the sample-KL is smaller than Δ (this event has density at least δ). □

2.4 Inaccessible Entropy and Witness KL-hardness

In this section, we relate our notion of witness KL-hardness to the inaccessible entropy definition of [HRVW19].

In the KL-hardness definition, the adversary can choose both $\widetilde{\mathsf{G}}$ and S. Contrary to the fixing $\widetilde{\mathsf{G}}$ to be honest in the definition of KL-hardness for simulating, now we fix S to be the sampler that generate the randomness of $\widetilde{\mathsf{G}}$ by rejection sampling for defining *KL-hardness for generating.* In order to have the simulator runs in polynomial time and succeeds in finding the randomness with high probability, we restrict $\widetilde{\mathsf{G}}$ to be online and its the output blocks to be short, so the sampler S generates the randomness of the online generator $\widetilde{\mathsf{G}}$ block by block. The hardness we obtain via making $\widetilde{\mathsf{G}}$ online and fixing S called *next-block (witness) KL-hardness for generating.*

By next-block witness KL-hardness for generating, we show it implies the next-block (min*-)inaccessible relative entropy (Definition 2.4.4), which is equivalent to (min-)inaccessible entropy defined in [HRVW19] when the joint distribution is flat (Proposition 2.4.5). Together, these results provides a modular proof of that if f is a one-way function, the generator $\mathsf{G}^f(X) = (f(X)_1, \ldots, f(X)_n, X)$ has super-logarithmic inaccessible entropy.

Next-block KL-hardness for generating

Consider a binary relation Π and a pair of random variables (Y, W) supported on Π. For an online $(m+1)$-block generator $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w) = (\widetilde{\mathsf{G}}_{y_1}, \ldots, \widetilde{\mathsf{G}}_{y_m}, \widetilde{\mathsf{G}}_w)$ supported on Π, it is natural to consider the simulator $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$ that exploits the block structure of $\widetilde{\mathsf{G}}_y$: on input $Y \overset{\text{def}}{=} (Y_1, \ldots, Y_m)$, $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}(Y)$ generates randomness $\widehat{R} = (\widehat{R}_1, \ldots, \widehat{R}_m)$ block by block using rejection sampling until $\widetilde{\mathsf{G}}_i(\widehat{R}_{\leq i}) = Y_i$. The subscript T is the maximum number of attempts after which $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$ gives up and outputs $\perp$. The formal definition of $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$ is given in Algorithm 2.4.1.

```
Algorithm 2.4.1: Rejection sampling simulator $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$
Input: $y_1, \ldots, y_m \in (\{0,1\}^*)^m$
Output: $\widehat{r}_1, \ldots, \widehat{r}_m \in (\{0,1\}^v \cup \{\perp\})^m$
For $i = 1 \to m$
    1. Repeat sampling $\widetilde{r}_i \xleftarrow{r} \{0,1\}^v$ until $\widetilde{\mathsf{G}}_{y_i}(\widetilde{r}_{\leq i}) = y_i$ or $\geq T$ attempts
    2. If $\widetilde{\mathsf{G}}_{y_i}(\widetilde{r}_{\leq i}) \neq y_i$ then $\widetilde{r}_j = \perp$ for all $j \geq i$. return
```

Once we fix the simulator to be $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$ for a given $\widetilde{\mathsf{G}}$, we can define next-block KL-hardnesses for generating:

Definition 2.4.1. *Let* Π *be a binary relation,* (Y, W) *be a pair of random variables supported on* Π*. and* $Y = (Y_1, \ldots, Y_m)$*. We say that* $(Y_1, \ldots, Y_m, W)$ *is* (t, Δ, T)-next-block witness-KL-hard for generating *if for every time* t *online* $(m+1)$*-block generator* $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w) = (\widetilde{\mathsf{G}}_{y1}, \ldots, \widetilde{\mathsf{G}}_{y_m}, \widetilde{\mathsf{G}}_w)$ *supported on* $(Y_1, \ldots, Y_m, W)$*, we have*

$$\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\Big\|\, \mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}, Y, W\Big) > \Delta,$$

where $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$ *and* $\widetilde{R}_w$ *are uniformly random. Similarly,* $(Y_1, \ldots, Y_m, W)$ *is* (t, Δ, T)*-next-block witness-*$\mathrm{D}^{\delta}_{\mathsf{min}^*}$*-hard for generating if for such online generator, we have*

$$\mathrm{D}^{\delta}_{\mathsf{min}^*}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\Big\|\, \mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}, Y, W\Big) > \Delta.$$

In the definition of next-block witness-KL-hardness for generating, we consider special cases for adversarial $(\widetilde{\mathsf{G}}, \mathsf{S})$. Thus, the implication from KL-hardness to next-block witness KL-hardness for generating is straightforward.

Theorem 2.4.2. *Let* Π *be a binary relation and let* (Y, W) *be a pair of random variables supported on* Π*. Let* $Y = (Y_1, \ldots, Y_m)$*. For every* $T \leq t/m$*, if* (Π, Y, W) *is* (t, Δ)*-witness KL-hard, then* $(Y_1, \ldots, Y_m, W)$ *is* $(O(t/(mT)), \Delta, T)$*-next-block witness-KL-hard for generating. Similarly, if if* (Π, Y, W) *is* (t, Δ)*-witness* $\mathrm{D}^{\delta}_{\mathsf{min}^*}$*-hard, then* $(Y_1, \ldots, Y_m, W)$ *is* $(O(t/(mT)), \Delta, T)$*-next-block witness-*$\mathrm{D}^{\delta}_{\mathsf{min}^*}$*-hard for generating.*

Proof. In Definition 2.4.1, let the running time of the online generator $\widetilde{\mathsf{G}}$ be t'. Without loss of generality, the maximum length of an input block v for simulator $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}$ is at most t'. Therefore, the running time of the simulator is $O(mTt')$. □

Inaccessible relative entropy

We first recall the definition of inaccessible entropy from [HRVW19], slightly adapted to our notations.

Definition 2.4.3 (inaccessible entropy)**.** *Let $Z = (Z_1, \ldots, Z_m)$ be a joint distribution. We say that $(Z_1, \ldots, Z_m)$ has* t-inaccessible entropy Δ *if for all m-block online generators $\widetilde{\mathsf{G}}$ running in time t and consistent with Z:*

$$\sum_{i=1}^{m} \Big(\mathrm{H}(Z_i \mid Z_{<i}) - \mathrm{H}(\widetilde{Z}_i \mid \widetilde{R}_{<i}) \Big) > \Delta \,.$$

where $(\widetilde{Z}_1, \ldots, \widetilde{Z}_m) = \widetilde{\mathsf{G}}(\widetilde{R}_1, \ldots, \widetilde{R}_m)$ for a uniform $\widetilde{R}_{\leq m+1}$. We say that $(Z_1, \ldots, Z_m)$ has (t, δ)-min-inaccessible entropy Δ *if for all m-block online generators $\widetilde{\mathsf{G}}$ running in time t and consistent with $(Z_1, \ldots, Z_m)$:*

$$\Pr_{\substack{r_{\leq m} \overset{r}{\leftarrow} \widetilde{R}_{\leq m} \\ y_{\leq m} \leftarrow \widetilde{\mathsf{G}}(r_{\leq m})}} \left[\sum_{i=1}^{m} \Big(\mathrm{H}^*_{z_i, z_{<i}}(Z_i \mid Z_{<i}) - \mathrm{H}^*_{y_i, r_{<i}}(\widetilde{Z}_i \mid \widetilde{R}_{<i}) \Big) \leq \Delta \right] < \delta \,.$$

One unsatisfactory aspect of Definition 2.4.3 is that inaccessible entropy can be negative since the generator $\widetilde{\mathsf{G}}$ could have more entropy than $Z = (Z_1, \ldots, Z_m)$: if all the Z_i are independent biased random bits, then a generator $\widetilde{\mathsf{G}}$ outputting unbiased random bits will have negative inaccessible entropy. We introduce the notion of inaccessible relative entropy, which remedies the above issue. Also, we will soon see that this notion more directly connects to our KL-hardness for generating.

Definition 2.4.4 (inaccessible relative entropy)**.** *The joint distribution $Z = (Z_1, \ldots, Z_m)$ has* t-inaccessible relative entropy Δ*, if for every time t online m-block generator $\widetilde{\mathsf{G}}$ supported on Z, writing $\widetilde{Z} = (\widetilde{Z}_1, \ldots, \widetilde{Z}_m) \overset{\text{def}}{=} \widetilde{\mathsf{G}}(\widetilde{R})$ for uniform $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$, we have*

$$\sum_{i=1}^{m} \mathrm{D}_{\mathsf{KL}}\Big(\widetilde{Z}_i \mid \widetilde{R}_{<i}, \widetilde{Z}_{<i} \,\Big\|\, Z_i \mid R_{<i}, Z_{<i} \Big) > \Delta \,,$$

where R_i is a "dummy" random variable over the domain of $\widetilde{\mathsf{G}}_i$ and independent of Z. Similarly, for $\delta \in [0, 1]$, we say that $(Z_1, \ldots, Z_m)$ has (t, δ)-min*-inaccessible relative entropy*, if for every $\widetilde{\mathsf{G}}$ as above, we have*

$$\Pr_{r \overset{r}{\leftarrow} \widetilde{R},\, z \leftarrow \widetilde{\mathsf{G}}(r)} \left[\sum_{i=1}^{m} \mathrm{D}^*_{z_i, r_{<i}, z_{<i}}\Big(\widetilde{Z}_i \mid \widetilde{R}_{<i}, \widetilde{Z}_{<i} \,\Big\|\, Z_i \mid R_{<i}, Z_{<i} \Big) \leq \Delta \right] < \delta \,,$$

where $\widetilde{Z}, R, \widetilde{R}$ are defined as above.

In the definition, since $\widetilde{Z}_{<i}$ is a function of $\widetilde{R}_{<i}$, the first conditional distribution in the KL is effectively $\widetilde{Z}_i|\widetilde{R}_{<i}$. Similarly the second distribution is effectively $Z_i|Z_{<i}$. The extra random variables are there for syntactic consistency.

In the case where Z is a flat distribution, then no distribution with the same support can have higher entropy. Moreover, (min-)inaccessible entropy Definitions 2.4.4 and 2.4.3 coincide to (min*-)inaccessible relative entropy as stated in the following observation. For example, the distribution $Z = (f(X), X)$ for a function f and uniform input X is always a flat distribution even if f itself is not regular.

Proposition 2.4.5. *Let $Z = (Z_1, , \ldots, Z_m)$ be a flat distribution and $\widetilde{\mathsf{G}}$ be an m-block generator consistent with $Z_{\leq m}$. Then for $\widetilde{Z} = \widetilde{\mathsf{G}}(\widetilde{R})$ for uniform $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$ we have that for every $z, r \in \operatorname{Supp}(Z, R)$,*

$$\sum_{i=1}^{m} \Big(\mathrm{H}^*_{z_i, z_{<i}}(Z_i \mid Z_{<i}) - \mathrm{H}^*_{z_i, r_{<i}}(\widetilde{Z}_i \mid \widetilde{R}_{<i}) \Big) = \sum_{i=1}^{m} \mathrm{D}^*_{z_i, z_{<i}, r_{<i}} \Big(\widetilde{Z}_i \mid \widetilde{R}_{<i}, \widetilde{Z}_{<i} \Big\| Z_i \mid R_{<i}, Z_{<i} \Big) .$$

In particular, $(Z_1, \ldots, Z_m)$ has (t, δ)-min*-inaccessible relative entropy at least Δ iff it has (t, δ)-min-inaccessible entropy at least Δ; $(Z_1, \ldots, Z_m)$ has t-inaccessible relative entropy at least Δ iff it has t-inaccessible entropy at least Δ.

Proof. For the sample notions, the chain rule (Proposition 2.2.3) gives:

$$\sum_{i=1}^{m} \mathrm{H}^*_{z_i, z_{<i}}(Z_i \mid Z_{<i}) = \mathrm{H}^*_z(Z_{\leq m}) = \log |\operatorname{Supp}(Z)|$$

for all z since Z is flat. Hence:

$$\begin{aligned} \log \Big|\operatorname{Supp}(Z)\Big| - \sum_{i=1}^{m} \mathrm{H}^*_{y_i, y_{<i}} \Big(\widetilde{Z}_i \mid \widetilde{R}_{<i} \Big) &= \sum_{i=1}^{m} \Big(\mathrm{H}^*_{z_i, z_{<i}}(Z_i \mid Z_{<i}) - \mathrm{H}^*_{y_i, r_{<i}}(\widetilde{Z}_i \mid \widetilde{R}_{<i}) \Big) \\ &= \sum_{i=1}^{m} \mathrm{D}^*_{z_i, z_{<i}, r_{<i}} \Big(\widetilde{Z}_i \mid \widetilde{R}_{<i}, \widetilde{Z}_{<i} \Big\| Z_i \mid R_{<i}, Z_{<i} \Big) . \end{aligned}$$

Taking the expectation over (Z, R) on both sides yields the equivalence between inaccessible entropy and inaccessible relative entropy. □

KL-hardness to inaccessible relative entropy

The last piece of the main result of this section is to show that next-block witness-KL-hardness for generating implies inaccessible relative entropy. The "approximation error" of the pair $(\widetilde{\mathsf{G}}, \mathsf{Sim}_T^{\widetilde{\mathsf{G}}})$ for next-block witness KL-hardness for generating, namely $\mathrm{D}_{\mathsf{KL}}(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\|\, \mathsf{Sim}_T^{\widetilde{\mathsf{G}}}(Y), Y, W)$, can be decomposed into two terms:

1. How well $\widetilde{\mathsf{G}}_y$ approximates the distribution Y in an online manner.

2. The success probability of the rejection sampling procedure.

The second term can be made arbitrarily small by setting the number of trials T in $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}}$ to be a large enough multiple of $m \cdot 2^\ell$ where ℓ is the length of the blocks of $\widetilde{\mathsf{G}}_y$ (Lemma 2.4.7). This leads to a poly(m) time algorithm whenever ℓ is logarithmic in m. That is, given an online block generator $\widetilde{\mathsf{G}}$ for which $\widetilde{\mathsf{G}}_y$ has short blocks, we obtain a corresponding simulator "for free". This let us connect the definition to inaccessible relative entropy, which makes no reference to simulators.

Theorem 2.4.6. *Let Π be a binary relation and let (Y, W) be a pair of random variables supported on Π. Let $Y = (Y_1, \ldots, Y_m)$ where the bit length of Y_i is at most ℓ. Then we have:*

1. *If $(Y_1, \ldots, Y_m, W)$ is (t, Δ, T)-next-block witness-KL-hard for generating, then $(Y_1, \ldots, Y_m, W)$ has t-inaccessible relative entropy $(\Delta - m \cdot 2^\ell/(T \ln 2))$.*

2. *If $(Y_1, \ldots, Y_m, W)$ is (t, Δ, T)-next-block witness-$\mathrm{D}^{\delta}_{\mathsf{min}^*}$-hard for generating, then for every $\delta' \in [0, 1-\delta]$, $(Y_1, \ldots, Y_m, W)$ has $(t, \delta + \delta')$-min*-inaccessible relative entropy $(\Delta - m \cdot 2^\ell/(T\delta' \ln 2))$.*

Proof. We will prove by contradiction: assume there exists an online generator that breaks the conditions of having (min*-)inaccessible relative entropy, then show that the same generator also breaks the next-block witness-KL($\mathrm{D}_{\mathsf{min}^*}$)-hardness for generating.

Let $\widetilde{\mathsf{G}} = (\widetilde{\mathsf{G}}_y, \widetilde{\mathsf{G}}_w) = (\widetilde{\mathsf{G}}_{y_1}, \ldots, \widetilde{\mathsf{G}}_{y_m}, \widetilde{\mathsf{G}}_w)$ be an $(m+1)$-block online generator. Define $\widetilde{Y} \overset{\text{def}}{=} \widetilde{\mathsf{G}}_y(\widetilde{R})$ for uniform $\widetilde{R} = (\widetilde{R}_1, \ldots, \widetilde{R}_m)$ and $\widetilde{W} \overset{\text{def}}{=} \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w)$ where $\widetilde{R}_w$ is also uniform. We also define $\widehat{R} \overset{\text{def}}{=} \mathsf{Sim}_T^{\widetilde{\mathsf{G}}}(Y)$ and $\widehat{Y} \overset{\text{def}}{=} \widetilde{\mathsf{G}}(\widehat{R})$.

First, we ignore the witness block and focus on sample notions. For every $r \in \mathrm{Supp}(\widetilde{R})$ and $y \stackrel{\mathrm{def}}{=} \widetilde{\mathsf{G}}(r)$, we have

$$\begin{aligned}
&\mathrm{D}^*_{r,y}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}(Y), Y\Big) = \mathrm{D}^*_{r,y}\Big(\widetilde{R}, \widetilde{Y} \,\Big\|\, \widehat{R}, \widehat{Y}\Big) \\
&= \sum_{i=1}^{m}\Big(\mathrm{D}^*_{r,y}\Big(\widetilde{R}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{\le i} \,\Big\|\, \widehat{R}_i \mid \widehat{R}_{<i}, \widehat{Y}_{\le i}\Big) + \mathrm{D}^*_{r,y}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{<i} \,\Big\|\, \widehat{Y}_i \mid \widehat{R}_{<i}, \widehat{Y}_{<i}\Big)\Big) \\
&= \sum_{i=1}^{m} \mathrm{D}^*_{r,y}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{<i} \,\Big\|\, \widehat{Y}_i \mid \widehat{R}_{<i}, \widehat{Y}_{<i}\Big) \\
&= \sum_{i=1}^{m} \mathrm{D}^*_{r,y}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i} \,\Big\|\, \widehat{Y}_i \mid \widehat{R}_{<i}\Big),
\end{aligned}$$

The first equality is by the fact that $\widetilde{\mathsf{G}}_y(\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}(y)) = y$ whenever $\mathsf{Sim}_T^{\widetilde{\mathsf{G}}_y}(y) \neq \perp$ (See Remark 2.3.3. The penultimate equality is by definition of rejection sampling: $\widetilde{R}_i|\widetilde{R}_{<i}, \widetilde{Y}_{\le i}$ and $\widehat{R}_i|\widehat{R}_{<i}, \widehat{Y}_{\le i}$ are identical on $\mathrm{Supp}(\widetilde{R}_i)$ since conditioning on $\widehat{Y}_i = y$ implies that only non-failure cases ($\widehat{R}_i \neq \perp$) are considered. The last equality is because $\widetilde{Y}_{<i}$ (resp. $\widehat{Y}_{<i}$) is a deterministic function of $\widetilde{R}_{<i}$ (resp. $\widehat{R}_{<i}$).

We now relate $\widehat{Y}_i|\widehat{R}_{<i}$ to $Y_i|Y_{<i}$:

$$\begin{aligned}
&\Pr\Big[\widehat{Y}_i = y_i | \widehat{R}_{<i} = r_{<i}\Big] \\
&\quad = \Pr\Big[\widehat{Y}_i = y_i, Y_i = y_i | \widehat{R}_{<i} = r_{<i}\Big] && (\widehat{Y}_i = y_i \Leftrightarrow \widehat{Y}_i = y_i \wedge Y_i = y_i) \\
&\quad = \Pr\Big[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}\Big] \cdot \Pr\Big[Y_i = y_i | \widehat{R}_{<i} = r_{<i}\Big] && \text{(Bayes' Rule)} \\
&\quad = \Pr\Big[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}\Big] \cdot \Pr\Big[Y_i = y_i | Y_{<i} = y_{<i}\Big],
\end{aligned}$$

where the last equality is because when $r \in \mathrm{Supp}(\widetilde{R})$, $\widehat{R}_{<i} = r_{<i} \Rightarrow Y_{<i} = y_{<i}$ and because Y_i is independent of $\widehat{R}_{<i}$ given $Y_{<i}$ (as $\widehat{R}_{<i}$ is simply a randomized function of $Y_{<i}$). Combining

the previous two derivations and putting back the witness block we obtain

$$
\begin{aligned}
&\mathrm{D}^*_{r,y,w}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\Big\|\, \mathsf{Sim}^{\widetilde{\mathsf{G}}}_T(Y), Y, W\Big) \\
={}&\mathrm{D}^*_{r,y}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, \mathsf{Sim}^{\widetilde{\mathsf{G}}}_T(Y), Y\Big) + \mathrm{D}^*_{r,y,w}\Big(\widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \mid \widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}) \,\Big\|\, W \mid \mathsf{Sim}^{\widetilde{\mathsf{G}}}_T(Y), Y\Big) \\
={}&\sum_{i=1}^{m} \mathrm{D}^*_{r,y}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{<i} \,\Big\|\, Y_i \mid R_{<i}, Y_{<i}\Big) + \mathrm{D}^*_{r,y,w}\Big(\widetilde{W} \mid \widetilde{R}, \widetilde{Y} \,\Big\|\, W \mid R, Y\Big) \\
&+ \sum_{i=1}^{m} \log \frac{1}{\Pr[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}]}\,.
\end{aligned}
\tag{2.2}
$$

When taking the expectation the last logarithmic term by the following lemma.

Lemma 2.4.7. *Let $\widetilde{\mathsf{G}}$ be an online m-block generator, and let $L_i \overset{\text{def}}{=} 2^{|\widetilde{\mathsf{G}}_i|}$ be the size of the codomain of $\widetilde{\mathsf{G}}_i$, $i \in [m]$. Then for all $i \in [m]$, $r_{<i} \in \operatorname{Supp}(\widetilde{R}_{<i})$ and uniform $\widetilde{R}_i$:*

$$
\mathop{\mathbb{E}}_{y_i \xleftarrow{r} \widetilde{\mathsf{G}}_i(r_{<i}, \widetilde{R}_i)} \left[\log \frac{1}{\Pr[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}]} \right] \le \log\left(1 + \frac{L_i - 1}{T}\right) \le \frac{L_i}{T \ln 2}\,.
$$

Now, the first claim of the main lemma follows by taking expectations on both sides of Equation 2.2 and directly applying Lemma 2.4.7.

$$
\begin{aligned}
&\mathrm{D}_{\mathsf{KL}}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\Big\|\, \mathsf{Sim}^{\widetilde{\mathsf{G}}}_T(Y), Y, W\Big) \\
&\qquad \le \sum_{i=1}^{m} \mathrm{D}_{\mathsf{KL}}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{<i} \,\Big\|\, Y_i \mid R_{<i}, Y_{<i}\Big) + \mathrm{D}_{\mathsf{KL}}\Big(\widetilde{W} \mid \widetilde{R}, \widetilde{Y} \,\Big\|\, W \mid R, Y\Big) + \frac{m \cdot 2^{\ell}}{T \ln 2}\,.
\end{aligned}
$$

For the second claim, assume for contradiction. That is, when sampling $r \xleftarrow{r} \widetilde{R}, r_w \xleftarrow{r} \widetilde{R}_w$ and letting $y \leftarrow \widetilde{\mathsf{G}}_y(r), w \leftarrow \widetilde{\mathsf{G}}_w(r, r_w)$, the following inequality holds with probability at least $\delta + \delta'$

$$
\sum_{i=1}^{m} \mathrm{D}^*_{y,r}\Big(\widetilde{Y}_i \mid \widetilde{R}_{<i}, \widetilde{Y}_{<i} \,\Big\|\, Y_i \mid R_{<i}, Y_{<i}\Big) + \mathrm{D}^*_{y,r,w}\Big(\widetilde{W} \mid \widetilde{R}, \widetilde{Y} \,\Big\|\, W \mid R, Y\Big) \ge \Delta - \frac{m \cdot 2^{\ell}}{T \delta' \ln 2}\,.
$$

Applying Markov's inequality on Lemma 2.4.7 we have

$$
\Pr_{(y,r) \xleftarrow{r} (\widetilde{Y}, \widetilde{R})} \left[\sum_{i=1}^{m} \log \frac{1}{\Pr[\widehat{Y}_i = y_i | \widehat{R}_{<i} = r_{<i}, \widehat{Y}_{<i} = y_{<i}]} \ge \frac{m \cdot 2^{\ell}}{T \delta' \ln 2} \right] \le \delta'
$$

Combining these inequality along with Equation 2.2, we have that

$$\Pr_{\substack{r \xleftarrow{\mathrm{r}} \widetilde{R}, r_w \xleftarrow{\mathrm{r}} \widetilde{R}_w \\ y \leftarrow \widetilde{\mathsf{G}}(r), w \leftarrow \widetilde{\mathsf{G}}_w(r, r_w)}} \left[\mathrm{D}^*_{r,y,w}\Big(\widetilde{R}, \widetilde{\mathsf{G}}_y(\widetilde{R}), \widetilde{\mathsf{G}}_w(\widetilde{R}, \widetilde{R}_w) \,\Big\|\, \mathsf{Sim}^{\widetilde{\mathsf{G}}}_T(Y), Y, W\Big) \geq \Delta \right] \geq \delta\,,$$

which breaks the next-block witness-$\mathrm{D}^{\delta}_{\mathsf{min}^*}$-hardness for generating. □

Proof of Lemma 2.4.7. By definition of $\mathsf{Sim}^{\widetilde{\mathsf{G}}}_T$, we have:

$$\Pr\Big[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}\Big] = 1 - \Big(1 - \Pr[\widetilde{\mathsf{G}}_{y_i}(r_{<i}, \widetilde{R}_i) = y_i]\Big)^T\ .$$

Applying Jensen's inequality, we have:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{y_i \xleftarrow{\mathrm{r}} \widetilde{\mathsf{G}}_{y_i}(r_{<i}, \widetilde{R}_i)} \left[\log \frac{1}{\Pr\big[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}\big]} \right] \\
&\quad \leq \log \mathop{\mathbb{E}}_{y_i \xleftarrow{\mathrm{r}} \widetilde{\mathsf{G}}_{y_i}(r_{<i}, \widetilde{R}_i)} \left[\frac{1}{\Pr\big[\widehat{Y}_i = y_i | Y_i = y_i, \widehat{R}_{<i} = r_{<i}\big]} \right] \\
&\quad = \log\Big(\textstyle\sum_{y \in \mathrm{Image}\left(\widetilde{\mathsf{G}}_{y_i}(r_{<i}, \cdot)\right)} \dfrac{p_y}{1 - (1 - p_y)^T} \Big)
\end{aligned}$$

where $p_y = \Pr\big[\widetilde{\mathsf{G}}_{y_i}(r_{<i}, \widetilde{R}_i) = y\big]$. Since the function $x \mapsto x/\Big(1 - (1 - x)^T\Big)$ is convex (see Fact 2.4.8 below), the maximum of the expression inside the logarithm over probability distributions $\{p_y\}$ is achieved at the extremal points of the standard probability simplex. Namely, when all but one $p_y \to 0$ and the other one is 1. Since $\lim_{x \to 0} x/1 - (1 - x)^T = 1/T$:

$$\log\Big(\textstyle\sum_{y \in \mathrm{Image}\left(\widetilde{\mathsf{G}}_{y_i}(r_{<i}, \cdot)\right)} \dfrac{p_y}{1 - (1 - p_y)^T} \Big) \leq \log\left(1 + (L_i - 1) \cdot \frac{1}{T}\right)\ .$$

□

Fact 2.4.8. *For all $t \geq 1$, $f : x \mapsto \frac{x}{1 - (1 - x)^t}$ is convex over $[0, 1]$.*

Proof. We instead show convexity of $\tilde{f} : x \mapsto f(1 - x)$. A straightforward computation gives:

$$\tilde{f}''(x) = \frac{x^{t-2} t\big(t(1 - x)(x^t + 1) - (1 + x)(1 - x^t)\big)}{(1 - x^t)^3}$$

so that it suffices to show the non-negativity of $g(x) = t(1 - x)(x^t + 1) - (1 + x)(1 - x^t)$ over $[0, 1]$. The function g has second derivative $t(1 - x)(t^2 - 1)x^{t-2}$, which is non-negative when

$x \in [0,1]$, and thus the first derivative g' is non-decreasing. Also, the first derivative at 1 is equal to zero, so that g' is non-positive over $[0,1]$ and hence g is non-increasing over this interval. Since $g(1) = 0$, this implies that g is non-negative over $[0,1]$ and f is convex as desired. □

Remark 2.4.9. *For fixed distribution and generators, in the limit where T grows to infinity, the error term caused by the failure of rejection sampling in time T vanishes. In this case, KL-hardness implies block-KL-hardness without any loss in the hardness parameters.*

By chaining the reductions between the different notions of hardness considered in this work (witness-KL-hardness, block-KL-hardness and inaccessible entropy), we obtain a more modular proof of the theorem of Haitner et al. [HRVW19], obtaining inaccessible entropy from any one-way function.

Theorem 2.4.10. *Let n be a security parameter, $f : \{0,1\}^n \to \{0,1\}^n$ be a (t,ε)-one-way function, and X be uniform over $\{0,1\}^n$. For $\ell \in \{1,\dots,n\}$, decompose $f(X) \stackrel{\text{def}}{=} (Y_1,\dots,Y_{n/\ell})$ into blocks of length ℓ. Then:*

1. *For every $0 \le \Delta \le \log(1/\varepsilon)$, $(Y_1,\dots,Y_{n/\ell},X)$ has t'-inaccessible entropy at least $(\log(1/\varepsilon) - \Delta)$ for $t' = t/O(\frac{n^2 \cdot 2^\ell}{\Delta \ell^2})$.*

2. *For every $0 < \delta \le 1$ and $0 \le \Delta \le \log(1/\varepsilon) - \log(2/\delta)$, $(Y_1,\dots,Y_{n/\ell},X)$ has (t',δ)-min-inaccessible entropy at least $(\log(1/\varepsilon) - \log(2/\delta) - \Delta)$ for $t' = t/O(\frac{n^2 \cdot 2^\ell}{\delta\Delta\ell^2})$.*

Proof. Since f is (t,ε)-one-way, the distributional search problem $(\Pi^f, f(X))$ where $\Pi^f = \{(f(x),x) : x \in \{0,1\}^n\}$ is (t,ε)-hard.

For the first claim, clearly $(f(X),X)$ is supported on Π^f, so by Theorem 2.3.11, $(\Pi^f, f(X), X)$ is $(\Omega(t), \log(1/\varepsilon))$-witness KL-hard. Then by Theorem 2.4.2, $(Y_1,\dots,Y_{n/\ell},X)$ is $(\Omega(t\ell/nT), \log(1/\varepsilon), T)$-next-block witness-KL-hard for generating. Take $T = n \cdot 2^\ell/\ell\Delta \ln 2$ and apply Theorem 2.4.6. $(Y_1,\dots,Y_{n/\ell},X)$ has $\Omega(t \cdot \Delta\ell^2/(n^2 \cdot 2^\ell))$-inaccessible relative entropy at least $(\log(1/\varepsilon) - \Delta)$, and hence inaccessible entropy (Proposition 2.4.5).

Proving the second claim is similar. By Theorem 2.3.11, $(\Pi^f, f(X), X)$ is $(\Omega(t), \log(1/\varepsilon) - \log(2/\delta))$-witness $\mathsf{D}^{\delta/2}_{\mathsf{min}^*}$-hard. Then by Theorem 2.4.2, $(Y_1,\dots,Y_{n/\ell},X)$ is

$(\Omega(t\ell/nT), \log(1/\varepsilon), T)$-next-block witness-$\mathrm{D}^{\delta/2}_{\mathsf{min}^*}$-hard for generating. Take $T = 2n{\cdot}2^\ell/\delta\ell\Delta\ln 2$ and apply Theorem 2.4.6. $(Y_1, \ldots, Y_{n/\ell}, X)$ has $(\Omega(t \cdot \Delta\ell^2/(\delta \cdot n^2 \cdot 2^\ell)), \delta)$-min*-inaccessible relative entropy at least $(\log(1/\varepsilon) - \Delta - \log(2/\delta))$, and hence min-inaccessible entropy (Proposition 2.4.5). □

Remark 2.4.11. *For comparison, the original proof of [HRVW19] shows that for every* $0 < \delta \leq 1$*,* $(Y_1, \ldots, Y_{n/\ell}, X)$ *has* (t', δ)*-min-inaccessible entropy at least* $(\log(1/\varepsilon) - 2\log(1/\delta) - O(1))$ *for* $t' = t/\widetilde{O}(\frac{n^2 \cdot 2^\ell}{\delta\ell^2})$*, which in particular for fixed* t' *has quadratically worse dependence on* δ *in terms of the achieved inaccessible entropy:* $\log(1/\varepsilon) - 2 \cdot \log(1/\delta) - O(1)$ *rather than our* $\log(1/\varepsilon) - 1 \cdot \log(1/\delta) - O(1)$*.*

Corollary 2.4.12 ([HRVW19, Theorem 4.2])**.** *Let* n *be a security parameter,* $f : \{0,1\}^n \to \{0,1\}^n$ *be a strong one-way function, and* X *be uniform over* $\{0,1\}^n$*. Then for every* $\ell = O(\log n)$*,* $(f(X)_{1\ldots\ell}, \ldots, f(X)_{n-\ell+1\ldots n}, X)$ *has* $n^{\omega(1)}$*-inaccessible entropy* $\omega(\log n)$ *and* $(n^{\omega(1)}, \mathrm{negl}(n))$*-min-inaccessible entropy* $\omega(\log n)$*.*

Chapter 3

Entropy Flattening

We study entropy flattening: given a circuit C_X implicitly describing an n-bit source X (namely, X is the output of C_X on a uniform random input), construct another circuit C_Y describing a source Y such that (1) source Y is nearly flat (uniform on its support), and (2) the Shannon entropy of Y is monotonically related to that of X. The standard solution is to have C_Y evaluate C_X altogether $\Theta(n^2)$ times on independent inputs and concatenate the results (correctness follows from the asymptotic equipartition property). In this paper, we show that this is optimal among black-box constructions: any circuit C_Y for entropy flattening that repeatedly queries C_X as an oracle requires $\Omega(n^2)$ queries.

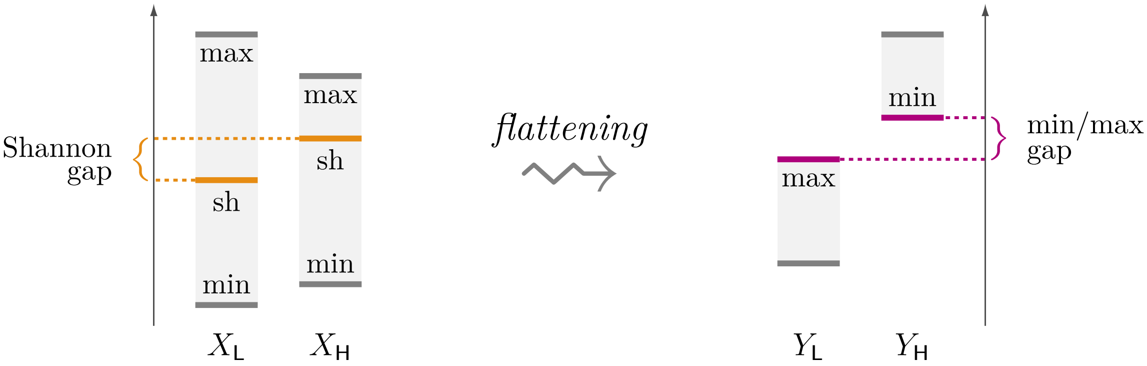
Entropy flattening is a component used in the constructions of pseudorandom generators and other cryptographic primitives from one-way functions [HILL99, Rom90, Hol06, HHR06, HRVW09, HRV13, HHR$^+$10, VZ12]. It is also used in reductions between problems complete for statistical zero-knowledge [Oka00, SV97, GSV99a, Vad99]. The $\Theta(n^2)$ query complexity is often the main efficiency bottleneck. Our lower bound hints that the current best construction of pseudorandom generator from arbitrary one-way functions by Vadhan and Zheng [VZ12] is likely to be optimal in terms of query complexity.

3.1 Introduction

We say a source X is *flat* if it is uniform over its support. Then a source X is flat iff its Shannon entropy, min-entropy, and max-entropy (written $\mathrm{H}_{\mathsf{Sh}}(X), \mathrm{H}_{\mathsf{min}}(X)$, and $\mathrm{H}_{\mathsf{max}}(X)$, respectively) are all equal.

The task. The *entropy flattening* is defined as the following task: given a circuit C_X implicitly describing an n-bit source X (namely, X is the output of C_X on a uniform random input), efficiently construct another circuit C_Y describing a "flattened" version Y of X. The goal is to have the output source Y (or a small statistical modification of it) be such that its min- and max-entropies are *monotonically* related to the Shannon entropy of X. Concretely, one interesting range of parameters is:

- if *input* sources X_{H} and X_{L} exhibit a 1-bit Shannon entropy gap, $\mathrm{H}_{\mathsf{Sh}}(X_{\mathsf{H}}) \geq \mathrm{H}_{\mathsf{Sh}}(X_{\mathsf{L}})+1$,
- then the respective *output* sources Y_{H} and Y_{L} must witness $\mathrm{H}_{\mathsf{min}}(Y_{\mathsf{H}}) \geq \mathrm{H}_{\mathsf{max}}(Y_{\mathsf{L}}) + 1$ (modulo a small modification to Y_{H} and Y_{L}).



Entropy flattening is not only an ingredient in constructions of cryptographic primitives from one-way functions as mentioned before, it is also used in reductions between problems complete for (non-interactive) statistical zero-knowledge [Oka00, SV97, GSV99a, Vad99].

A solution: repeat X. The standard strategy for entropy flattening is to construct Y as the concatenation X^q of some q i.i.d. copies of the input source X. That is, in circuit language, $\mathsf{C}_Y(x_1, \ldots, x_q) = (\mathsf{C}_X(x_1), \ldots, \mathsf{C}_X(x_q))$. The well-known *asymptotic equipartition* *property* in information theory states that X^q is ε-close[1] to having min- and max-entropies closely approximated by $q \cdot \mathrm{H}_{\mathsf{Sh}}(X)$. (It is common to say that X^q has a certain ε-*smooth* min- and max-entropy [RW04].)

Lemma 3.1.1 ([HILL99, HR11]). *Let X be an n-bit random variable. For any $q \in \mathbb{N}$ and $\varepsilon > 0$ there is an nq-bit random variable Y' that is ε-close to $Y = X^q$ such that*

$$\mathrm{H}_{\mathsf{min}}(Y'),\ \mathrm{H}_{\mathsf{max}}(Y') \ \in\ q \cdot \mathrm{H}_{\mathsf{Sh}}(X) \pm O\Big(n\sqrt{q \cdot \log(1/\varepsilon)}\Big)\,.$$

In particular, it suffices to set $q = \widetilde{\Theta}(n^2)$ in order to flatten entropies in the aforementioned interesting range of parameters (1-bit Shannon gap implies at least 1-bit min/max gap). The analysis here is also tight by a reduction to standard anti-concentration results: it is *necessary* to have $q = \Omega(n^2)$ in order for the construction $Y = X^q$ to flatten entropies.

3.1.1 Our result

We show that any *black-box* construction for entropy flattening—that is, a circuit C_Y which treats C_X as a black-box oracle—requires $\Omega(n^2)$ oracle queries to C_X. This is formalized in Theorem 3.1.2 below.

In particular, the simple "repeat-X" strategy is optimal among all black-box constructions. Besides querying C_X on independent inputs, a black-box algorithm has the freedom to perform adaptive queries, and it can produce outputs that are arbitrary functions of its query/answer execution log (rather than merely concatenating the answers). For example, this allows the use of hash functions and randomness extractors, which is indeed useful for variations of the flattening task (e.g., Lemma 3.2.2).

Query model. In our black-box model, the input source is now encoded as the output distribution of an *arbitrary* function $f : \{0,1\}^n \to \{0,1\}^m$ where $m = \Theta(n)$ (not necessarily computed by a small circuit); namely, the input source is $f(U_n)$ where U_n denotes the uniform

[1] Random variables Z_1 and Z_2 on $\mathcal{Z}$ are ε-*close* if $\mathrm{d}_{\mathsf{TV}}(Z_1\,,\,Z_2) \le \varepsilon$ where $\mathrm{d}_{\mathsf{TV}}(Z_1\,,\,Z_2)$ is the usual statistical (or total variation) distance, given by $\mathrm{d}_{\mathsf{TV}}(Z_1\,,\,Z_2) = \max_{T \subseteq \mathcal{Z}} |\Pr[Z_1 \in T] - \Pr[Z_2 \in T]|$.

distribution over n-bit strings. We consider oracle algorithms A^f that have query access to f. Given an n'-bit input w (thought of as a random seed) to A^f, the algorithm computes by repeatedly querying f (on query $x \in \{0,1\}^n$ it gets to learn $f(x)$), until it finally produces some m'-bit output string $\mathsf{A}^f(w)$. We denote by $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ the function computed by A^f. Thus $\mathsf{A}^f(U_{n'})$ is the output source.

Inputs/outputs. Our input sources come from the promise problem Entropy-Approximation; the circuit version of this problem is complete for the complexity class **prBPL** (non-interactive statistical zero-knowledge), as shown by Goldreich, Sahai, and Vadhan [GSV99a]. The Entropy-Approximation promise problem is (here $\tau \in \mathbb{N}$ is a threshold parameter):

- YES input: (f, τ) such that $\mathrm{H}_{\mathsf{Sh}}(f(U_n)) \geq \tau + 1$.
- NO input: (f, τ) such that $\mathrm{H}_{\mathsf{Sh}}(f(U_n)) \leq \tau - 1$.

The goal of a flattening algorithm A^f (which also gets τ as input, but we supress this in our notation) is to produce an output distribution that is statistically close to having high min-entropy or low max-entropy depending on whether the input source f is a YES or a NO instance. We say that A^f is an (ε, Δ)-*flattening algorithm* if (here $\kappa = \kappa(\tau)$ is a parameter that A^f gets to choose):

- (f, τ) is a YES input $\Rightarrow$ $\mathsf{A}^f(U_{n'})$ is ε-close to a distribution Z_{H} with $\mathrm{H}_{\mathsf{min}}(Z_{\mathsf{H}}) \geq \kappa + \Delta$.
- (f, τ) is a NO input $\Rightarrow$ $\mathsf{A}^f(U_{n'})$ is ε-close to a distribution Z_{L} with $\mathrm{H}_{\mathsf{max}}(Z_{\mathsf{L}}) \leq \kappa - \Delta$.

Main theorem. Our main result is the following theorem.

Theorem 3.1.2. *There exist constants $\varepsilon, \Delta > 0$ such that every (ε, Δ)-flattening algorithm for n-bit oracles f requires $\Omega(n^2)$ oracle queries.*

In fact, our proof yields an even more fine-grained lower bound. Suppose we allow ε and Δ to vary subject to $n/25 \geq \Delta \geq \log(1/\varepsilon)$. Then our lower bound becomes $\Omega(n^2 \log(1/\varepsilon))$, which is tight in both n and ε.

3.1.2 Relevance to cryptographic constructions

Many constructions of PRG [HILL99, Hol06, HRV10, VZ12], SHC [HNO+09, HRVW09, HRVW19], and UOWHF [Rom90, KK05, HHR+10] use the flattening technique to transform Shannon-like entropies to min or max-like entropies. To illustrate the usage of flattening, we take HILL's construction [HILL99] as an example, which was also the first use of flattening in complexity-based cryptography.

The first step is to show any one-way function $f : \{0,1\}^n \to \{0,1\}^m$ is also a *HILL entropy generator*. That is, $Y = f(U_n)$ is computationally indistinguishable from a random variable Y' such that $\mathrm{H}_{\mathsf{Sh}}(Y')$ is noticeably higher than $\mathrm{H}_{\mathsf{Sh}}(Y)$. In other words, for some threshold τ and a non-negligible gap parameter Δ it holds that:

1. $\mathrm{H}_{\mathsf{HILL\text{-}Sh}}(f(U_n)) \geq \tau + \Delta$, and
2. $\mathrm{H}_{\mathsf{Sh}}(f(U_n)) \leq \tau - \Delta$.

Then the recipe of flattening algorithm is applied. Specifically evaluating f on many independent inputs yields a distribution that is close to having low max-entropy yet is computationally indistinguishable from having high min-entropy. Note that even thought the flattening algorithm we stated is for real entropies, it also applies to HILL-type entropies (which can be a simple reduction.).

After flattening, universal hashing (or randomness extraction) is applied to obtain a pseudorandom generator $\mathsf{G}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$, where $\mathsf{G}^f(U_{n'})$ is computationally indistinguishable from $U_{m'}$ (i.e. indistinguishable from min-entropy at least m') yet has max-entropy at most $n' \leq m' - 1$ (due to having a seed length of n'). Note that in HILL's construction, the query complexity due to the flattening step is $\widetilde{\Theta}(n^4)$, rather than $\widetilde{\Theta}(n^2)$ in Lemma 3.1.1 since the gap Δ is $\widetilde{\Theta}(1/n)$ in this case.

A series of subsequent works [Hol06, HHR06, HRV10, VZ12] improved the efficiency of the HILL construction. The state-of-the-art constructions [HRV13, VZ12] replace the HILL entropy with the more refined pseudoentropy notion next-block HILL entropy introduced previously and thereby obtain $\Delta = \widetilde{\Theta}(1)$ in the entropy gap. In the best constructions the query complexity is $\widetilde{\Theta}(n^3)$, an extra cost of $\widetilde{\Theta}(n)$ due to the fact that how the entropy is spread out among the bits of the output of the next-block HILL entropy generator.

In the best constructions, there is still an exact cost of $\widetilde{\Theta}(n)$ due to the fact that we don't know how the entropy is spread out among the bits of the output of the next-bit HILL entropy generator f.

Overall, with the most efficient constructions to date, the pseudorandom generator makes $\widetilde{\Theta}(n^3)$ queries to the one-way function, of which a $\widetilde{\Theta}(n^2)$ factor is due to flattening. This complexity renders the constructions too inefficient for practice, and thus it is important to know whether a more efficient construction is possible.

Lower bound in constructing PRG. The work of Gennaro, Gertner, Katz, and Trevisan [GGKT05] gave the first lower bound on constructing pseudorandom generators from one-way functions. Specifically they proved that any "black-box" construction of a pseudorandom generator $\mathsf{G}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ from a one-way function $f : \{0,1\}^n \to \{0,1\}^m$ requires $\Omega((m' - n')/\log n)$ queries to f. Thus, many queries are needed to construct a pseudorandom generator with large stretch. However, their lower bound says nothing about the number of queries needed to obtain a pseudorandom generator with small stretch (i.e., where $m' = n' + O(\log n)$), and indeed it applies even to one-way permutations f, where no flattening is needed and a pseudorandom generator with small stretch can be obtained with a single query to the one-way function [GL89].

For constructing pseudorandom generators with small stretch from one-way functions, Holenstein and Sinha [HS12] proved that any black-box construction requires $\widetilde{\Omega}(n)$ queries. Their lower bound is also independent to flattening, as it applies even to *regular* one-way functions, which directly (with one query) give a separation between HILL min-entropy and max-entropy. Rather, their lower bound corresponds to the efficiency costs coming from not knowing the entropy thresholds τ (or how the entropy is spread across the bits in the case of next-bit HILL entropy).

Our lower bound for flattening (Theorem 3.1.2) can be viewed as a first-step towards proving that any black-box construction of pseudorandom generators from one-way functions requires $\widetilde{\Omega}(n^2)$ queries. One might hope to also combine this with [HS12] and obtain a lower bound of $\widetilde{\Omega}(n^3)$ queries, which would match the best-known construction of [VZ12].

Seed length. Another important and well-studied efficiency criterion for pseudorandom generator constructions is how the seed length n' of the pseudorandom generator $\mathsf{G}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ depends on the input length n of the one-way function $f : \{0,1\}^n \to \{0,1\}^m$. The standard method for flattening (Lemma 3.1.1) requires independent samples from the distribution being flattened, and thus the query complexity of flattening contributes a multiplicative factor to the seed length of the pseudorandom generator. For example, the construction of [VZ12] gives a pseudorandom generator with seed length $\widetilde{\Theta}(n^2) \cdot n = \widetilde{\Theta}(n^3)$, as $\widetilde{\Theta}(n^2)$ independent evaluations of the one-way function (or corresponding pseudoentropy generator) are used for flattening. An interesting open problem is to show that independent evaluations are indeed necessary, and extend our lower bound on query complexity to a lower bound on the input length n' of the flattening algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$. This could be a first step towards proving a superlinear lower bound on the seed length of pseudorandom generators constructed (in a black-box way) from one-way functions, a long-standing open problem. We note that the existing lower bounds on query complexity of [GGKT05, HS12]cannot be turned into seed length lower bounds, as there are constructions of large-stretch pseudorandom generators from regular one-way functions with seed length $\widetilde{O}(n)$ [HHR06]. That is, although those constructions make polynomially many queries to the one-way functions, the queries are highly correlated (and even adaptive).

Other cryptographic primitives. Flattening is also an efficiency bottleneck in the constructions of other cryptographic primitives from arbitrary one-way functions, such as universal one-way hash functions [Rom90, KK05, HHR+10] and statistically hiding commitment schemes [HNO+09, HRVW09, HRVW19]. In both cases, the state-of-the-art constructions begin by constructing a function f where there is a gap between its output entropy $\mathrm{H}_{\mathsf{Sh}}(f(U_n))$ and a computational analogue of Shannon entropy (namely, a form of “inaccessible entropy”). Then flattening is applied, after which some (possibly interactive) hashing techniques are used to obtain the final cryptographic primitive. Again, our lower bound on flattening can be viewed as a first step towards proving an efficiency lower bound on black-box constructions.

We note that there was a very fruitful interplay between this sequence of works on constructions of cryptographic primitives from one-way functions and general results about **SZK** and **prBPL**, with inspirations going in both directions (e.g., [NV06, HRVW09, OV08, HRVW09]). This reinforces the feeling that our lower bound for flattening the **prBPL**-complete problem Entropy-Approximation can help in understanding the PRG constructions.

3.2 Proof Overview

Our proof builds on the recent result of Lovett and Zhang [LZ17], who showed that there is no efficient black-box reduction (making polynomially many queries) from Entropy-Approximation to its complement, thereby giving evidence that **prBPL** is not closed under complement and hence that **prBPL** $\neq$ **SZK**. The result of [LZ17] a qualitative one, whereas here we are concerned with a quantitative question: What is the exact query complexity of flattening? Nevertheless, we use a similar construction of hard instances as [LZ17] and make use of a variation of their key lemma.

3.2.1 Simplification: the SDU problem

We find it convenient to work with a slightly simplified version of the flattening task, having one fewer parameter to worry about.

Definition 3.2.1 (statistical distance from uniform (SDU))**.** *We say an algorithm* $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ *is a* k-SDU algorithm *if for all* $f : \{0,1\}^n \to \{0,1\}^m$*, we have*

- *If* (f, τ) *is a YES input to* Entropy-Approximation*, then* $\mathsf{A}^f(U_{n'})$ *is* 2^{-k}*-close to* $U_{m'}$*.*
- *If* (f, τ) *is a NO input to* Entropy-Approximation*, then* $|\mathrm{Supp}(\mathsf{A}^f(U_{n'}))| \leq 2^{m'-k}$*.*

Note that a k-SDU algorithm is a $(2^{-k}, k/2)$-flattening algorithm (with threshold $\kappa = m' - k/2$). Conversely, we can transform any flattening algorithm to a SDU algorithm using hashing techniques similar to [GSV99a]:

(see Section 3.5.3 for the proof)

Lemma 3.2.2. *If there exists a (ε, Δ)-flattening algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q, then there exists a k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n''} \to \{0,1\}^{n''-3k}$ where $n'' = O(n' + m')$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q and $k = \Omega(\min\{\Delta, \log(1/\varepsilon)\})$. In particular, there exists such a k-SDU algorithm with query complexity $O(k \cdot \min\{n,m\}^2)$.*

Remark 3.2.3. *Note that Lemma 2.2 guarantees not only that A is a k-SDU algorithm but also that its output length is only $3k$ bits shorter than its input length. This additional property will be useful in our proof.*

By Lemma 3.2.2, for our main result (Theorem 3.1.2), it suffices to prove an $\Omega(kn^2)$ query lower bound for any k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ with $m' = n' - 3k$ and $k \leq n/25$.

Theorem 3.2.4. *Let $k \leq n$. Every k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ for function $f : \{0,1\}^n \to \{0,1\}^m$ has query complexity $\Omega(kn^2)$.*

3.2.2 Hard instances

We consider two input distributions F_{H} and F_{L} over functions $f : \{0,1\}^n \to \{0,1\}^{3n}$ such that the entropies of most functions in F_{H} and F_{L} are at least $\tau + 1$ and at most $\tau - 1$ (where $\tau = \Theta(n)$), respectively. To sample a function from F_{H}, we randomly partition the domain of f into many blocks $B_1, B_2, \ldots, B_S$, each of size $T = 2^n/S$ where $S = 2^{3n/4}$. For each block B_i,

- with probability $1/2 + \Theta(1/n)$ we insert a high-entropy block: $f|_{B_i}$ will be a uniformly random mapping from B_i to $\{0,1\}^{3n}$; and
- with the remaining probability $1/2 - \Theta(1/n)$, we insert a low-entropy block: all elements of B_i are mapped to the same random element of $\{0,1\}^{3n}$.

The distribution F_{L} is the same, except we swap the two $1/2 \pm \Theta(1/n)$ probabilities.

Note that since the range $\{0,1\}^{3n}$ is much larger than the domain $\{0,1\}^n$, with high probability f will be injective on the high-entropy blocks and have no collisions between different blocks. Under this condition, if we let $\mathrm{Blo}(x)$ denote the block containing x (which is determined by $f(x)$) and p be the fraction of high entropy blocks, we have

$$\mathrm{H}_{\mathsf{Sh}}(f(U_n)) = \mathrm{H}_{\mathsf{Sh}}(\mathrm{Blo}(U_n)) + \mathrm{H}_{\mathsf{Sh}}(f(U_n) \mid \mathrm{Blo}(U_n)) \tag{3.1}$$

$$= \log S + p \cdot \log \frac{2^n}{S} + (1-p) \cdot 0 = \frac{3n}{4} + p \cdot \frac{n}{4}. \tag{3.2}$$

Under F_{H} we have $p = \frac{1}{2} + \Theta(\frac{1}{n})$ whp, and under F_{L} we have $p = \frac{1}{2} - \Theta(\frac{1}{n})$ whp, which yields a constant gap in Shannon entropies, as desired.

3.2.3 Basic intuition—and a warning!

The first natural instinct—but too naive, we argue—is that since the bias between observing a high-entropy block versus a low-entropy block is only $\Theta(1/n)$, an anti-concentration bound should imply that distinguishing the two distributions takes $\Omega(n^2)$ queries.

This intuition indeed applies to simple bounded-error randomized decision trees (which output just a 1-bit answer). Concretely, suppose for simplicity that our input is just an n^2-bit string x (instead of an exponentially large oracle f): each bit x_i represents either a high-entropy block ($x_i = 1$) or a low-entropy block ($x_i = 0$). We are given the following *gap-majority* promise: the relative Hamming weight $|x|/n^2$ is either $1/2 + 1/n$ or $1/2 - 1/n$. It is a well-known fact that any bounded-error query algorithm needs $\Omega(n^2)$ queries to distinguish these two cases.

But surprisingly enough, there does exist[2] a flattening/SDU algorithm A^x that solves the *gap-majority* promise problem with only $O(n)$ queries! This suggests that any superlinear

[2] Consider the following algorithm A^x on input a random seed w: query a sequence of random positions i (according to w) until a position with $x_i = 1$ is found, output $\mathsf{A}^x(w) = i$. It is easy to verify that this is an $(0, \Theta(1/n))$-flattening algorithm with expected query complexity $O(1)$. Repeating the algorithm $\Theta(n)$ many times yields an $(0, \Omega(1))$-flattening algorithm with expected query complexity $O(n)$. Finally, we can make the algorithm abort if any run exceeds the expected query complexity by a large constant factor; this results in an $(\varepsilon, \Omega(1))$-flattening algorithm of worst-case query complexity $O(n)$.

lower bound must somehow hide from the algorithm the type (high vs. low) of a queried block. Our choice of distributions F_{H} and F_{L} does indeed achieve this: since there are so many blocks, a single run of the algorithm is unlikely to query more than one point in a single block, and the marginal distribution of such a single query is the same in both F_{H} and F_{L}. The more precise way in which we exploit the hidden type of a block is in invoking the main result of [LZ17]: when switching a high-entropy block in an f to a low-entropy block, the support of an SDU algorithm's output distribution, $\mathrm{Supp}(\mathsf{A}^f(U_{n'}))$, cannot increase by much.

3.2.4 Technical outline

Recall that $\mathsf{A}^f(U_{n'})$ is almost-uniform when f has high entropy. For almost all $z \in \{0,1\}^{m'}$, most of the high-entropy functions f make the algorithm A^f output z (on some random seed):

$$\Pr_{f \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}}}\left[\exists w \in \{0,1\}^{n'}, \mathsf{A}^f(w) = z\right] \geq 1 - 2^{-\Omega(k)}. \tag{3.3}$$

On the other hand, since the support of $\mathsf{A}^f(U_{n'})$ is small when f has low entropy, there should be many z such that when we sample f from F_{L}, with high probability $\mathsf{A}^f(w)$ does not output z:

$$\Pr_{f \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}}\left[\exists w \in \{0,1\}^{n'}, \mathsf{A}^f(w) = z\right] \leq 2^{-\Omega(k)}. \tag{3.4}$$

To connect the high-entropy and low-entropy cases, we essentially prove that for many $z \in \{0,1\}^{m'}$ and every algorithm A making $o(kn^2)$ queries, we have

$$\Pr_{f \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}}}[\exists w \in \{0,1\}^{n'}, \mathsf{A}^f(w) = z] \leq 2^{o(k)} \cdot \Pr_{f \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}}[\exists w \in \{0,1\}^{n'}, \mathsf{A}^f(w) = z] + O(2^{-k}). \tag{3.5}$$

As long as there exists z such that Equation (3.3), (3.4) and (3.5) hold, combining those equations contradict inequality (3.5).

Our inequality (3.5) is similar to the key lemma of Lovett and Zhang [LZ17]. However, the inequality is reversed, we have an extra multiplicative factor of $2^{o(k)}$, and our lemma (necessarily) only applies to algorithms making $o(kn^2)$ queries (where the [LZ17] lemma applies even to exponentially many queries).

One key step toward Inequality (3.5) is to reverse the direction of the inequality by the following trick. We name elements of $\{0,1\}^{n'}$ as $w_1, \ldots, w_{2^{n'}}$ in some arbitrarily fixed order. Then

$$
\begin{aligned}
&\Pr_{f \overset{\mathrm{r}}{\leftarrow} F}\left[\exists w \in \{0,1\}^{n'}, \mathsf{A}^f(w) = z\right] \\
&= \sum_{\ell=1}^{2^{n'}} \Pr_{f \overset{\mathrm{r}}{\leftarrow} F}\left[\mathsf{A}^f(w_\ell) = z \text{ and } \nexists w \in \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z\right] \\
&= \sum_{\ell=1}^{2^{n'}} \left(1 - \Pr_{f \overset{\mathrm{r}}{\leftarrow} F}\left[\exists w \in \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z \,\middle|\, \mathsf{A}^f(w_\ell) = z\right]\right) \cdot \Pr_{f \overset{\mathrm{r}}{\leftarrow} F}\left[\mathsf{A}^f(w_\ell) = z\right].
\end{aligned}
$$

Having a negative sign, now we wish to relate the probability of

$$
\Pr_{f \overset{\mathrm{r}}{\leftarrow} F}\left[\exists v \in \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z \,\middle|\, \mathsf{A}^f(w_\ell) = z\right]
$$

over F_{H} and F_{L} in the same direction as [LZ17]. It is not a direct application of their lemma due to the fact that the block size is constant in their construction and our probability is conditioned on the event $\mathsf{A}^f(w_\ell) = z$, but we prove a generalization (Lemma 3.5.3) of their lemma that suffices for our purpose. In fact, the proof we provide in Section 3.5.2 is potentially simpler than the one in [LZ17] and yields better parameters.

Like in [LZ17], instead of considering the event $\exists w, \mathsf{A}^f(w) = z$ in all the probabilities above, we further impose the restriction that $\mathsf{A}^f(w)$ queries each block B_i of the domain at most once, since this event happens with high probability. Furthermore (unlike [LZ17]), we also restrict to the case that the number of high-entropy block queries is in the range $q \cdot (1/2 \pm (O(1/n) + O(1/\sqrt{q})))$ out of a total of q queries, which also occurs with high probability.

3.3 The Hard Distribution

Let $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ be a potential k-SDU algorithm for functions $f : \{0,1\}^n \to \{0,1\}^m$. Throughout, we will consider a fixed oracle algorithm A^f with query complexity q, and will omit the dependency of A in most notations. For a vector $\vec{\mathcal{X}}$, we use $\vec{\mathcal{X}}(j)$ to denote the j-th element of $\vec{\mathcal{X}}$, and $\vec{\mathcal{X}}$ means the unordered set $\left\{\vec{\mathcal{X}}(j) : j \in [|\vec{\mathcal{X}}|]\right\}$.

It is equivalent to interpret an element $\{0,1\}^n$ as an integer in $[N]$ where $N = 2^n$, since we do not make a use of any structure in $\{0,1\}^n$. Under this notation, we are considering a fixed oracle algorithm $\mathsf{A}^f : [N'] \to [M']$ for functions $f : [N] \to [M]$ where $N' = 2^{n'}, M' = 2^{m'}, N = 2^n$ and $M = 2^m$. Actually, we will allow N, M, N' and M' to be arbitrary positive integers (not necessary a power of 2).

Partition. Given parameters $S, T \in \mathbb{N}$ where $ST = N$, and a function $f : [N] \to [M]$, we will partition the domain $[N]$ into S *blocks* $\mathcal{X}_1, \ldots, \mathcal{X}_S$ each of size T. We will also fix an order for the blocks and the elements in each block: $\boldsymbol{\vec{\mathcal{X}}} = (\vec{\mathcal{X}}_1, \ldots, \vec{\mathcal{X}}_S)$. So $\vec{\mathcal{X}}_i(j)$ denotes the j-th element of the i-th block. Given a vector $\vec{\mathcal{Y}}_i \in [M]^T$, we use the shorthand $f(\vec{\mathcal{X}}_i) = \vec{\mathcal{Y}}_i$ to denote the assignments $f(\vec{\mathcal{X}}_i(j)) = \vec{\mathcal{Y}}_i(j)$, for all $j \in [T]$. Therefore, once vectors $\vec{\mathcal{Y}}_1, \ldots, \vec{\mathcal{Y}}_S \in [M]^T$ and a partition $\boldsymbol{\vec{\mathcal{X}}}$ are determined, the function f is fully defined as $f(\vec{\mathcal{X}}_i) = \vec{\mathcal{Y}}_i$ for all $i \in [S]$.

Distributions.

- Let $\mathbf{X}_S$ be a uniform distribution over an ordered partitions $\boldsymbol{\vec{\mathcal{X}}} = (\vec{\mathcal{X}}_1, \ldots, \vec{\mathcal{X}}_S)$ of $[N]$ where $|\vec{\mathcal{X}}_i| = N/S = T$ for all $i \in [S]$.

- Let $\mathbf{Y}_0$ and $\mathbf{Y}_1$ be distributions on vectors $\vec{\mathcal{Y}} \in [M]^T$ defined as follows,

  For $\mathbf{Y}_0$, uniformly sample an element $y \leftarrow [M]$, and output $\vec{\mathcal{Y}}(1) = \cdots = \vec{\mathcal{Y}}(T) = z$.

  For $\mathbf{Y}_1$, uniformly and independently sample $\vec{\mathcal{Y}}(1), \ldots, \vec{\mathcal{Y}}(T)$ from $[M]$.

- Given a vector $\vec{b} \in \{0,1\}^S$ and a partition $\boldsymbol{\vec{\mathcal{X}}} = (\vec{\mathcal{X}}_1, \ldots, \vec{\mathcal{X}}_S)$ of $[N]$, we define the distribution $F(\boldsymbol{\vec{\mathcal{X}}}, \vec{b})$ of function $f : [N] \to [M]$ such that $f(\vec{\mathcal{X}}_i) = \vec{\mathcal{Y}}_i$ where $\vec{\mathcal{Y}}_i \leftarrow \mathbf{Y}_{\vec{b}(i)}$ for all $i \in [S]$. Essentially, $\vec{b}$ indicates whether each block is "high entropy" or "low entropy".

- For $0 \leq \alpha \leq 1$, let $\mathbf{B}_\alpha$ be a distribution over vectors $\vec{b} \in \{0,1\}^S$, so that each entry of $\vec{b}$ is sampled from the Bernoulli distribution $\mathrm{Bern}(\alpha)$ independently.

- For $0 \leq \alpha \leq 1$, F_α is a distribution on functions $f : [N] \to [M]$, a partition $\vec{\boldsymbol{\mathcal{X}}}$, and an indicator vector $\vec{b}$, where $(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_\alpha$ means that $\vec{b} \overset{\mathrm{r}}{\leftarrow} \mathbf{B}_\alpha$, $\vec{\boldsymbol{\mathcal{X}}} \overset{\mathrm{r}}{\leftarrow} \mathbf{X}_S$ and $f \overset{\mathrm{r}}{\leftarrow} F(\vec{\boldsymbol{\mathcal{X}}}, \vec{b})$.

Block-Compatibility. When an algorithm A runs with input $w \in [N']$ and oracle $f : [N] \to [M]$, let $\mathrm{Query}_f(w) \subseteq [N]$ be the set of the queries made by the algorithm $\mathsf{A}^f(w)$ to f. We say w is *block-compatible* with $(f, \boldsymbol{\mathcal{X}})$ if $|\mathrm{Query}_f(w) \cap \mathcal{X}| \leq 1$ for all blocks $\mathcal{X} \in \boldsymbol{\mathcal{X}}$. The set of block-compatible inputs with $(f, \boldsymbol{\mathcal{X}})$ is denoted

$$\mathrm{BC}(f, \boldsymbol{\mathcal{X}}) = \{w : w \text{ is block-compatible with } (f, \boldsymbol{\mathcal{X}})\}$$

Construction. Set $m = 3n$, so $M = N^3$. Also, set $S = 2^{3n/4} = N^{3/4}$ and $T = 2^{n/4} = N^{1/4}$. Let the high entropy distribution be $F_{\mathsf{H}} \overset{\mathrm{def}}{=} F_{1/2+5/n}$ and the low entropy distribution be $F_{\mathsf{L}} \overset{\mathrm{def}}{=} F_{1/2+5/n}$. We claim that with high probability, a function f from F_{H} and F_{L} has entropy at least $\tau + 1$ and at most $\tau - 1$ for $\tau = 7n/8$.

Lemma 3.3.1. *Let the parameters be as above. Then we have*

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{H}}} [\mathrm{H}_{\mathsf{Sh}}(f) \geq \tau + 1] \geq 1 - 2^{-0.9n}$$

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{L}}} [\mathrm{H}_{\mathsf{Sh}}(f) \leq \tau - 1] \geq 1 - 2^{-0.9n}$$

Proof. For any pair of independent and random mappings to M, the collision probability is $1/M$. There are no more than N^2 pairs of inputs, so with probability at least $1 - N^2/M = 1 - 2^{-n}$, there is no collision when two images are sampled independently. Under that condition, as shown by Equation (3.1), let p be the fraction of high entropy blocks, namely p is the hamming weight of $\vec{b}$ divided by S, the entropy of the function f is

$$\mathrm{H}_{\mathsf{Sh}}(f(U_n)) = \frac{3n}{4} + p \cdot \frac{n}{4}.$$

Recall that when we sample $\vec{b}$ from F_{H}, $\vec{b}(i) \overset{\mathrm{r}}{\leftarrow} \mathrm{Bern}(1/2 + 5/n)$ for all $i \in [S]$. By the

Chernoff bound,

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{\mathrm{r}}F_{\mathsf{H}}}\left[p \geq \frac{1}{2}+\frac{4}{n}\right] \geq 1-\frac{1}{4}2^{S\cdot(1/n)^2},$$

which implies

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{\mathrm{r}}F_{\mathsf{H}}}\left[\mathrm{H}_{\mathsf{Sh}}(f) \geq \frac{3n}{4}+\left(\frac{1}{2}+\frac{4}{n}\right)\cdot\frac{n}{4}=\frac{7n}{8}+1\right] \geq 1-2^{-\frac{1}{4}\cdot S\cdot(1/n)^2}-2^{-n}=1-2^{-0.9n}.$$

Similarly, when sampling from F_{L},

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{\mathrm{r}}F_{\mathsf{L}}}\left[\mathrm{H}_{\mathsf{Sh}}(f) \leq \frac{3n}{4}+\left(\frac{1}{2}-\frac{4}{n}\right)\cdot\frac{n}{4}=\frac{7n}{8}-1\right] \geq 1-2^{-\frac{1}{4}\cdot S\cdot(1/n)^2}-2^{-n}=1-2^{-0.9n}.$$

Taking $\tau=\frac{7n}{8}$ concludes the lemma. □

3.4 Query Lower Bound for SDU Algorithms

Let A^f be a k-SDU algorithm making exact q oracle queries to f and all the query positions are distinct. We may assume that since it is useless to query same positions, and if the number of queries is less than q then we simply make some dummy queries. We derive the lower bound (Theorem 3.2.4) from the following two lemmas.

Lemma 3.4.1. *Let A^f be a k-SDU algorithm making q queries. For every $n > 25k$ and $z \in [M']$ that satisfies*

$$\mathop{\mathbb{E}}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{H}}}\left[\left|\{w : \mathsf{A}^f(w)=z\}\right|\right] \leq 2^{4k}, \tag{3.6}$$

we have

$$\begin{aligned}&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{H}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w)=z\right]\\ \leq\quad & 2^{O\left(\frac{q}{n^2}+\sqrt{\frac{kq}{n^2}}\right)}\cdot\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{L}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w)=z\right]+O(2^{-k})\end{aligned} \tag{3.7}$$

Lemma 3.4.2. *There exists a universal constant $c > 0$ such that for every sufficiently large n and $k \leq n$, there is an output $z \in [M']$ that satisfies*

1. $\displaystyle\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{H}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w)=z\right] \geq 1-2^{-ck} \geq \frac{1}{2}.$

2. $\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{L}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\right] \le 2^{-ck}$.

3. $\mathbb{E}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{H}}}\left[\left|\{w : \mathsf{A}^f(w) = z\}\right|\right] \le 2^{4k}$.

Theorem 3.2.4 follows by plugging z that satisfies the inequalities in Lemma 3.4.2 into Inequality (3.7). If $q = o(kn^2)$, then the exponent in Equation (3.7) is $o(k)$, which yields a contradiction.

In the following section, we prove that most inputs are block-compatible and hence we can only consider the block-compatible inputs rather than the whole domain $[N']$. Then we prove Lemma 3.4.1 and 3.4.2 in Section 3.4.2 and 3.5.1, respectively.

3.4.1 Block-compatible inputs

As in [LZ17], we only consider block-compatible inputs, where each block is queried at most once. In that case, it is easier to compare the behavior of the SDU algorithms. Since there are exponentially many blocks but only polynomially many queries, intuitively, the probability of having block-compatible property is overwhelming if we randomly partition the domain of f. Formally,

Lemma 3.4.3. *For every $w \in [N']$ and $\alpha \in [0,1]$,*

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\alpha}}\left[w \notin \mathrm{BC}(f,\boldsymbol{\mathcal{X}})\right] \le \frac{q^2}{S} \le 2^{-0.6n}.$$

Proof. In order to handle adaptive algorithms, we consider Procedure 3.4.1 for sampling $(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}})$, which is equivalent to sampling from F_α. The essential idea is sampling the parts that are related to w first. By the principle of deferred decisions, it can be verified that the joint distribution of $(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}})$ is identical to F_α.

Notice that $w \in \mathrm{BC}(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}})$ if and only if the sequence of q values of i selected in Step 2(a) are all distinct. The probability that the $(r+1)^{\mathrm{st}}$ value of i is the same one comparing to the previous r values is at most $rT/(ST - r) \le q/S$, since $r \le q - 1$ and $qr \le ST$. So the probability that there are any repetitions is at most q^2/S. □

By Markov's inequality, almost all inputs are block-compatible:

Procedure 3.4.1

1. Initially, $\vec{\mathcal{X}}_i(j) = *$ and $\vec{b}(i) = *$ for all $i \in [S], j \in [T]$.
2. Simulate $\mathsf{A}^f(w)$ handling the r-th oracle query x_r as follows. For $r = 1, \dots, q$,
   (a) Based on previous queries and results as well as w, let the r-th query be x_r. Select (i, j) uniformly at random from $[S] \times [T]$ subject to $X_i(j) = *$ and assign $\vec{\mathcal{X}}_i(j) = x_r$.
   (b) If $\vec{b}(i) = *$, then assign $\vec{b}(i) \overset{\mathrm{r}}{\leftarrow} \mathrm{Bern}(\alpha)$ and $\vec{\mathcal{Y}}_i \overset{\mathrm{r}}{\leftarrow} \mathbf{Y}_{\vec{b}(i)}$.
   (c) Set $f(x_r) = \vec{\mathcal{Y}}_i(j)$ and return $f(x_r)$ as the answer to the query.
3. Assign the rest of the vectors $\vec{\mathcal{X}}$ and $\vec{b}$ by executing Step 2(a)–2(c) for all $x \in [N] \setminus \{x_1, \dots, x_q\}$.

Corollary 3.4.4. *For every $\alpha \in [0, 1]$,*

$$\Pr_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_\alpha} \left[|\mathrm{BC}(f, \boldsymbol{\mathcal{X}})| > N' \cdot (1 - 2^{-0.3n}) \right] \geq 1 - 2^{-0.3n}$$

3.4.2 Proof of Lemma 3.4.1

Lemma 3.4.1 (restatement)**.** *Let A^f be a k-SDU algorithm making q queries. For every $n > 25k$ and $z \in [M']$ that satisfies*

$$\mathop{\mathbb{E}}_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{H}}} \left[\left| \{ w : \mathsf{A}^f(w) = z \} \right| \right] \leq 2^{4k}, \tag{3.6}$$

we have

$$\begin{aligned} &\Pr_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{H}}} \left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z \right] \\ \leq \quad & 2^{O\left(\frac{q}{n^2} + \sqrt{\frac{kq}{n^2}} \right)} \cdot \Pr_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{L}}} \left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z \right] + O(2^{-k}) \end{aligned} \tag{3.7}$$

Proof. Define the set

$$W_z(f, \boldsymbol{\mathcal{X}}) = \left\{ w : w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}) \ , \ \mathsf{A}^f(w) = z \right\}.$$

Let $w_1, \cdots, w_{N'}$ be all possible inputs sorted in arbitrary but fixed order. The first step is to break the event $\exists w \in W_z(f, \boldsymbol{\mathcal{X}})$ to the events that w_ℓ is the "first" one in $W_z(f, \boldsymbol{\mathcal{X}})$ for all $\ell \in [N']$.

$$\begin{aligned}
&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_\alpha}\Big[\exists w \in W_z(f, \boldsymbol{\mathcal{X}})\Big] \\
&= \sum_{\ell=1}^{N'} \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_\alpha}\Big[w_\ell \in W_z(f, \boldsymbol{\mathcal{X}}) \wedge w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}})\Big] \\
&= \sum_{\ell=1}^{N'} \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_\alpha}\Big[w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\Big|\; w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big] \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_\alpha}\Big[w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big]
\end{aligned}$$

Our goal is to switch the distribution from F_{H} to F_{L} and see how the probability changes. We switch using the following two claims.

Claim 3.4.5. *For every $w_\ell \in [N']$, $\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_\alpha}[w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})]$ does not depend on $\alpha \in [0,1]$. In particular,*

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{H}}}\Big[w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big] = \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{L}}}\Big[w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big].$$

Claim 3.4.6. *For every $w_\ell \in [N']$ and $z \in [M']$,*

$$\begin{aligned}
&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{H}}}\Big[w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\Big|\; w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big] \\
&\leq 2^{O\left(\frac{q}{n^2} + \sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{L}}}\Big[w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\Big|\; w_\ell \in W_z(f, \boldsymbol{\mathcal{X}})\Big] + O\Big(\frac{q^2}{S}\Big) + 2^{-5k}
\end{aligned}$$

The intuition behind Claim 3.4.5 is that as long as w_ℓ is block-compatible, the query results are independently uniform over $[M]$ in both F_{H} or F_{L} case. Note that unlike Lemma 3.4.1, Claim 3.4.6 refers to *non*-membership in $W_z(f, \boldsymbol{\mathcal{X}})$, which allows us to use the main lemma of Lovett and Zhang [LZ17], which provides an inequality in the opposite direction of Lemma 3.4.1. See the formal proofs of those Claims after the main proof.

Once we have the above claims, we can prove the lemma:

$$\begin{aligned}
&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_{\mathsf{H}}}\Big[\exists w \in W_z(f, \boldsymbol{\mathcal{X}})\Big] \\
&\leq 2^{O\left(\frac{q}{n^2} + \sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_{\mathsf{L}}}\Big[\exists w \in W_z(f, \boldsymbol{\mathcal{X}})\Big]
\end{aligned}$$

$$
\begin{aligned}
&\qquad + \left(O\Big(\frac{q^2}{S}\Big) + 2^{-5k}\right) \cdot \sum_{\ell=1}^{2^{n'}} \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{H}}} \Big[w_\ell \in W_z(f,\boldsymbol{\mathcal{X}})\Big] \\
&\le 2^{O\left(\frac{q}{n^2} + \sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}} \Big[\exists w \in W_z(f,\boldsymbol{\mathcal{X}})\Big] \\
&\qquad + \left(O\Big(2^{-n/5}\Big) + 2^{-5k}\right) \cdot \mathop{\mathbb{E}}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}}} \Big[\big|\{w : \mathsf{A}^f(w) = z\}\big|\Big] \\
&\le 2^{O\left(\frac{q}{n^2} + \sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}} \Big[\exists w \in W_z(f,\boldsymbol{\mathcal{X}})\Big] + O(2^{-k}).
\end{aligned}
$$

The second inequality is by the assumption of $n > 25k$, and the last inequality is by Inequality (3.6). □

Proof of Claim 3.4.5

Claim 3.4.5 (restatement)**.** *For every $w_\ell \in [N']$, $\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_\alpha}[w_\ell \in W_z(f,\boldsymbol{\mathcal{X}})]$ does not depend on $\alpha \in [0,1]$. In particular,*

$$
\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{H}}} \Big[w_\ell \in W_z(f,\boldsymbol{\mathcal{X}})\Big] = \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{r}{\leftarrow} F_{\mathsf{L}}} \Big[w_\ell \in W_z(f,\boldsymbol{\mathcal{X}})\Big].
$$

Proof. We factorize the probability into two parts and prove that both of them are independent of α.

$$
\begin{aligned}
&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_\alpha} \Big[w_\ell \in W_z(f,\boldsymbol{\mathcal{X}})\Big] \\
&= \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_\alpha} \Big[\mathsf{A}^f(w_\ell) = z \,\Big|\, w_\ell \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}})\Big] \cdot \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_\alpha} \Big[w_\ell \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}})\Big]
\end{aligned}
$$

We use Procedure 3.4.1 to sample $(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}})$. We will prove the second factor is independent of α by induction over r. Conditioning on the first $(r-1)$ values of i selected in Step 2(a) being all distinct, that is, the block-compatible property has not been violated in the first r rounds, we have $\vec{b}(i) = *$ at the beginning of Step 2(b) in the r-th round. Thus no matter what α is and what $\vec{b}(i)$ is assigned, $\vec{\mathcal{Y}}_i(j)$ is uniform over $[M]$ in the r-th round. Therefore, under the assumed condition, the distribution of x_r and $f(x_r)$ are independent of α and the probability of maintaining the block-compatible property in the r-th round is independent of α. By induction, we know that the probability of maintaining the block-compatible property in all q rounds is independent of α.

For the first factor, as discussed above, conditioning on the block-compatible property, the distributions of x_r and $f(x_r)$ are independent of α, so the probability of getting z as the output of $\mathsf{A}^f(w_\ell)$ is also independent of α. □

Proof of Claim 3.4.6

Claim 3.4.6 (restatement)**.** *For every $w_\ell \in [N']$ and $z \in [M']$,*

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_{\mathsf{H}}}\left[w_1,\dots,w_{\ell-1}\notin W_z(f,\boldsymbol{\mathcal{X}})\;\middle|\;w_\ell\in W_z(f,\boldsymbol{\mathcal{X}})\right]$$
$$\le 2^{O\left(\frac{q}{n^2}+\sqrt{\frac{kq}{n^2}}\right)}\cdot\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_{\mathsf{L}}}\left[w_1,\dots,w_{\ell-1}\notin W_z(f,\boldsymbol{\mathcal{X}})\;\middle|\;w_\ell\in W_z(f,\boldsymbol{\mathcal{X}})\right]+O\left(\frac{q^2}{S}\right)+2^{-5k}$$

Proof. We consider the Procedure 3.4.2 for sampling $(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})$, which is equivalent to sampling from F_α conditioned on $w_\ell\in W_z(f,\boldsymbol{\mathcal{X}})$ (Namely, $\mathsf{A}^f(w_\ell)=z$ and $w_\ell\in \mathrm{BC}(f,\boldsymbol{\mathcal{X}})$). We denote such a distribution as $(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_\alpha(w_\ell,z)$. It follows the same idea as in Procedure 3.4.1 — sampling the blocks that are queried by $\mathsf{A}^f(w_\ell)$ first, and using the rejection sampling to handle the condition $w_\ell\in W_z(f,\boldsymbol{\mathcal{X}})$. Notice that until Step 5, information (including the partition $\vec{\mathcal{X}}^*$, function mapping f^* and the indicator $\vec{b}^*$) on exactly q blocks is decided.

The probability we consider then can be written as

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_\alpha}\left[w_1,\dots,w_{\ell-1}\notin W_z(f,\boldsymbol{\mathcal{X}})\;\middle|\;w_\ell\in W_z(f,\boldsymbol{\mathcal{X}})\right]$$
$$=\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_\alpha(w_\ell,z)}\left[w_1,\dots,w_{\ell-1}\notin W_z(f,\boldsymbol{\mathcal{X}})\right]$$
$$=\sum_{(f^*,\vec{b}^*,\vec{\mathcal{X}}^*)}\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\overset{r}{\leftarrow}F_\alpha(w_\ell,z)}\left[w_1,\dots,w_{\ell-1}\notin W_z(f,\boldsymbol{\mathcal{X}})\;\middle|\;(f^*,\vec{b}^*,\vec{\boldsymbol{\mathcal{X}}}^*)\right]\times\Pr_{F^*_\alpha(w_\ell,z)}\left[(f^*,\vec{b}^*,\vec{\boldsymbol{\mathcal{X}}}^*)\right]$$

Now we introduce a property of a partial indicator. We say a partial indicator is balanced if the number of zeros (low entropy block) and ones (high entropy block) are about the same.

Definition 3.4.7 (balanced)**.** *Let $\vec{b}^*\in\{0,1,*\}^S$ be a "partial" indicator vector where there are q non-star entries. We say it is* balanced *if the number of 1's is in $[q\cdot(1/2-5/n-\sqrt{25k/q}),q\cdot(1/2+5/n+\sqrt{25k/q})]$.*

Procedure 3.4.2: PROC:W-FIRST-BC

1. Initially, $\vec{\mathcal{X}}_i(j) = *$ and $\vec{b}(i) = *$ for all $i \in [S], j \in [T]$ and $f(x) = *$ for all $x \in [N]$.

2. Simulate $\mathsf{A}^f(w_\ell)$ handling the r-th oracle query x_r as follows. For $r = 1 \dots, q$,

   (a) Based on previous queries and results as well as w, let the r-th query be x_r. Select (i, j) uniformly at random from $[S] \times [T]$ subject to $\vec{\mathcal{X}}_i(j) = *$ and assign $\vec{\mathcal{X}}_i(j) = x_r$.

   (b) If $\vec{b}(i) = *$, then assign $\vec{b}(i) \overset{\text{r}}{\leftarrow} \mathrm{Bern}(\alpha)$ and $\vec{\mathcal{Y}}_i \overset{\text{r}}{\leftarrow} \mathbf{Y}_{\vec{b}(i)}$.

   (c) Set $f(x_r) = \vec{\mathcal{Y}}_i(j)$ and return $f(x_r)$ as the answer to the query.

3. If q values of i in Step 2(a) are not all distinct, or $\mathsf{A}^f(w_\ell) \neq z$, **restart**.

4. For all (i, j) such that $\vec{b}(i) \neq *$ and $\vec{\mathcal{X}}_i(j) = *$, randomly sample $x \in [N]$ that has not been assigned to any partition. Set $\vec{\mathcal{X}}_i(j) = x$ and $f(x) = \vec{\mathcal{Y}}_i(j)$.

5. Denote the partially assigned (some of them are mapped to $*$) function and vectors sampled so far as $(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*) \overset{\text{r}}{\leftarrow} F^*_\alpha(w_\ell, z)$.

6. Assign the rest of the vectors $\vec{\boldsymbol{\mathcal{X}}}$, $\vec{b}$ and the mapping f by executing Step 2(a)–(c) for all $x \in [N] \setminus \{x_1, \dots, x_q\}$ (instead of x_r).

According to Procedure 3.4.2, each non-star entry of $\vec{b}^*$ is sampled uniformly and independently from $\mathrm{Bern}(\alpha)$. When $\alpha \in [1/2 - 5/n, 1/2 + 5/n]$, by Chernoff bound, we have

$$\Pr_{(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*) \overset{\text{r}}{\leftarrow} F^*_\alpha(w_\ell, z)} \left[\vec{b}^* \text{ is balanced } \right] \geq 1 - 2^{-5k}.$$

And thus we can sum over only balanced $\vec{b}^*$ by paying an additive term.

$$\begin{aligned}
&\Pr_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{\text{r}}{\leftarrow} F_\alpha} \left[w_1, \dots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\middle|\; w_\ell \in W_z(f, \boldsymbol{\mathcal{X}}) \right] \\
&\leq 2^{-5k} \quad + \sum_{\substack{(f^*, \vec{b}^*, \vec{\mathcal{X}}^*) \\ \text{where } \vec{b}^* \text{ is balanced}}} \Pr_{(f, \vec{b}, \vec{\boldsymbol{\mathcal{X}}}) \overset{\text{r}}{\leftarrow} F_\alpha(w_\ell, z)} \left[w_1, \dots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*) \right] \\
&\qquad\qquad \times \Pr_{F_\alpha(w_\ell, z)^*} \left[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*) \right]
\end{aligned} \tag{3.8}$$

Now we use the following two claims (proved in the later paragraphs) to connect the high entropy case (F_{H}) and the low entropy case (F_{L}) on those two factors.

Subclaim 3.4.8. *For every $w_\ell \in [N']$, $z \in [M']$ and $(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*) \in \operatorname{Supp}(F_H^*(w_\ell, z))$, we have*

$$\begin{aligned}\Pr_{F_{\mathsf{H}}(w_\ell,z)}&\left[w_1,\ldots,w_{\ell-1} \notin W_z(f,\boldsymbol{\mathcal{X}}) \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \\ &\leq \Pr_{F_{\mathsf{L}}(w_\ell,z)}\left[w_1,\ldots,w_{\ell-1} \notin W_z(f,\boldsymbol{\mathcal{X}}) \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] + O\left(\frac{q^2}{S}\right)\end{aligned} \tag{3.9}$$

Subclaim 3.4.9. *For every $w_\ell \in [N']$, $z \in [M']$ and every $(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)$ where $\vec{b}^*$ is balanced,*

$$\Pr_{F_{\mathsf{H}}^*(w_\ell,z)}\left[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \leq 2^{O\left(\frac{q}{n^2}+\sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{F_{\mathsf{L}}^*(w_\ell,z)}\left[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \tag{3.10}$$

Inserting Inequalities (3.9) and (3.10) to Equation (3.8) with $\alpha = 1/2 + 5/n$, we conclude the claim. □

Proof of Subclaim 3.4.8. This claim is heavily relied on a variant of the main lemma (Lemma 3) in [LZ17] (see the proof in Section 3.5.2):

Lemma 3.4.10. *Let $\hat{\mathsf{A}}^{\hat{f}} : [\hat{N}'] \to [\hat{M}']$ be an algorithm making at most q oracle queries to $\hat{f} : [\hat{N}] \to [\hat{M}]$. Let $\hat{F}_H = \hat{F}_{1/2+5/n}$ and $\hat{F}_L = \hat{F}_{1/2-5/n}$ be the distribution over a function $\hat{f} : [\hat{N}] \to [\hat{M}]$, a partition $\vec{\hat{\boldsymbol{\mathcal{X}}}} \in \left([\hat{N}]^{\hat{T}}\right)^{\hat{S}}$ where $\hat{T}\hat{S} = \hat{N}$, and an indication vector $\vec{\hat{b}} \in \{0,1\}^{\hat{S}}$ defined in Section 3.3. Then for all $z \in [\hat{N}']$,*

$$\Pr_{(\hat{f},\vec{\hat{b}},\vec{\hat{\boldsymbol{\mathcal{X}}}})\overset{r}{\leftarrow}\hat{F}_L}\left[\exists w \in \mathrm{BC}(\hat{f},\hat{\boldsymbol{\mathcal{X}}}), \hat{\mathsf{A}}^{\hat{f}}(w) = z\right] - \Pr_{(\hat{f},\vec{\hat{b}},\vec{\hat{\boldsymbol{\mathcal{X}}}})\overset{r}{\leftarrow}\hat{F}_H}\left[\exists w \in \mathrm{BC}(\hat{f},\hat{\boldsymbol{\mathcal{X}}}), \hat{\mathsf{A}}^{\hat{f}}(w) = z\right] \leq \frac{O(q^2)}{\hat{S}}.$$

For a fixed $(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)$, apply the above lemma in the following way:

- Let $\hat{S} = S - q$, $\hat{T} = T$, and so $\hat{N} = \hat{S} \cdot \hat{T} = N - qT$.

- Let $\mathcal{Z} = \{x \mid f^*(x) = *\} \subseteq [N]$, $\mathcal{I} = \{i \mid \vec{b}^*(i) = *\} \subseteq [S]$ and $\pi_{\mathsf{x}} : \mathcal{Z} \to [\hat{N}]$,

$\pi_{\mathsf{i}} : \mathcal{I} \to [\hat{S}]$ be arbitrary bijection mappings. Then we define $\hat{f}, \vec{\hat{X}}$ and $\vec{\hat{b}}$ as follows.

$$\begin{cases} \forall\, \hat{x} \in [\hat{N}] & , \quad \hat{f}(\hat{x}) \overset{\text{def}}{=} f(\pi_{\mathsf{x}}^{-1}(\hat{x})) \\ \forall\, (\hat{i}, \hat{j}) \in [\hat{S}] \times [\hat{t}] & , \quad \vec{\hat{\boldsymbol{\mathcal{X}}}}_{\hat{i}}(\hat{j}) \overset{\text{def}}{=} \pi_{\mathsf{x}}(\vec{\boldsymbol{\mathcal{X}}}_{\pi_{\mathsf{i}}^{-1}(\hat{i})}(\hat{j})) \\ \forall\, \hat{i} \in [\hat{S}] & , \quad \vec{\hat{b}}(\hat{i}) \overset{\text{def}}{=} \vec{b}(\pi_{\mathsf{i}}^{-1}(\hat{i})) \end{cases} .$$

- For $\hat{w} \in [\hat{N}]$, define $\hat{\mathsf{A}}^{\hat{f}}(\hat{w})$ to simulate $\mathsf{A}^f(w)$ and $w \in \{w_1, \ldots, w_{\ell-1}\}$ in the following way. It first check that if $w \notin \{w_1, \ldots, w_{\ell-1}\}$, output something not equal to z. Otherwise simulate $\mathsf{A}^f(w)$ and when A makes a query $x \in \boldsymbol{\mathcal{X}}^*$, $\hat{\mathsf{A}}$ hardwire the result $f(x)$ as the answer. When $x \in \mathcal{Z}$, return $\hat{f}(\pi_{\mathsf{x}}(x))$ as the answer.

By the above mapping, we have

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{\text{r}} F_\alpha(w_\ell,z)} \left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}) \cap \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right]$$
$$= \Pr_{(\hat{f},\vec{\hat{b}},\vec{\hat{\boldsymbol{\mathcal{X}}}}) \xleftarrow{\text{r}} \hat{F}_\alpha} \left[\exists w \in \mathrm{BC}(\hat{f}, \hat{\boldsymbol{\mathcal{X}}}), \hat{\mathsf{A}}^{\hat{f}}(w) = z\right].$$

By Lemma 3.4.10,

$$\begin{aligned} &\Pr_{F_{\mathsf{H}}(w_\ell,z)} \left[w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \\ &= 1 - \Pr_{F_{\mathsf{H}}(w_\ell,z)} \left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}) \cap \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \\ &= 1 - \Pr_{(\hat{f},\vec{\hat{b}},\vec{\hat{\mathbf{X}}}) \xleftarrow{\text{r}} \hat{F}_H} \left[\exists w \in \mathrm{BC}(\hat{f}, \hat{\boldsymbol{\mathcal{X}}}), \hat{\mathsf{A}}^{\hat{f}}(w) = z\right] \\ &\le 1 - \Pr_{(\hat{f},\vec{\hat{b}},\vec{\hat{\mathbf{X}}}) \xleftarrow{\text{r}} \hat{F}_L} \left[\exists w \in \mathrm{BC}(\hat{f}, \hat{\boldsymbol{\mathcal{X}}}), \hat{\mathsf{A}}^{\hat{f}}(w) = z\right] + O\left(\frac{q^2}{\hat{S}}\right) \\ &= 1 - \Pr_{F_{\mathsf{L}}(w_\ell,z)} \left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}) \cap \{w_1, \ldots, w_{\ell-1}\}, \mathsf{A}^f(w) = z \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] + O\left(\frac{q^2}{S}\right) \\ &= \Pr_{F_{\mathsf{L}}(w_\ell,z)} \left[w_1, \ldots, w_{\ell-1} \notin W_z(f, \boldsymbol{\mathcal{X}}) \;\middle|\; (f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] + O\left(\frac{q^2}{S}\right). \end{aligned}$$

□

Proof of Subclaim 3.4.9. We restate the subclaim:

Subclaim 3.4.9 (restatement)**.** *For every $w_\ell \in [N']$, $z \in [M']$ and every $(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)$ where $\vec{b}^*$ is balanced,*

$$\Pr_{F^*_{\mathsf{H}}(w_\ell,z)}\left[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \leq\ 2^{O\left(\frac{q}{n^2}+\sqrt{\frac{kq}{n^2}}\right)} \cdot \Pr_{F^*_{\mathsf{L}}(w_\ell,z)}\left[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)\right] \tag{3.10}$$

Proof. The only difference between distributions $F_{\mathsf{L}}(w_\ell, z)$ and $F_{\mathsf{H}}(w_\ell, z)$ is when sampling $\vec{b}^*$. Recall that a balanced partial indicator means the hamming weight is within the range $q \cdot \left(1/2 \pm (1/n + \sqrt{25k/q})\right)$. Since we only consider the cases where $\vec{b}^*$ is balanced, the ratio can be bounded as follows.

$$\begin{aligned}
\frac{\Pr_{F^*_{\mathsf{H}}(w_\ell,z)}[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)]}{\Pr_{F^*_{\mathsf{L}}(w_\ell,z)}[(f^*, \vec{b}^*, \vec{\boldsymbol{\mathcal{X}}}^*)]} &\leq \left(\frac{\frac{1}{2}+\frac{5}{n}}{\frac{1}{2}-\frac{5}{n}}\right)^{q\left(\frac{1}{2}+\left(\frac{1}{n}+\sqrt{\frac{25k}{q}}\right)\right)} \left(\frac{\frac{1}{2}-\frac{5}{n}}{\frac{1}{2}+\frac{5}{n}}\right)^{q\left(\frac{1}{2}-\left(\frac{1}{n}+\sqrt{\frac{25k}{q}}\right)\right)} \\
&\leq \left(1+\frac{10}{n}\right)^{2q\left(\frac{1}{n}+\sqrt{\frac{25k}{q}}\right)} \left(1-\frac{10}{n}\right)^{-2q\left(\frac{1}{n}+\sqrt{\frac{25k}{q}}\right)} \qquad (3.11) \\
&\leq 2^{O\left(\frac{q}{n^2}+\sqrt{\frac{kq}{n^2}}\right)}
\end{aligned}$$

□

3.5 Appendix

3.5.1 Proof of Lemma 3.4.2

Lemma 3.4.2 (restatement)**.** *There exists a universal constant $c > 0$ such that for every sufficiently large n and $k \leq n$, there is an output $z \in [M']$ that satisfies*

1. $$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{H}}}\left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\right] \geq 1 - 2^{-ck} \geq \frac{1}{2}.$$

2. $$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{L}}}\left[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\right] \leq 2^{-ck}.$$

3. $$\mathop{\mathbb{E}}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} F_{\mathsf{H}}}\left[\left|\{w : \mathsf{A}^f(w) = z\}\right|\right] \leq 2^{4k}.$$

Proof. In this proof, we abuse notation by denoting $\mathrm{BC}(f, \boldsymbol{\mathcal{X}})$ also to be the uniform distribution over the set $\mathrm{BC}(f, \boldsymbol{\mathcal{X}})$. We will show that that for a random z sampled from $[M']$, it satisfies each property with probability at least $1 - 2^{-\Omega(k)}$, and hence by the union bound, it satisfies all three properties with probability at least $1 - 2^{-\Omega(k)}$. In particular, there exists $z \in [M']$ satisfying all three conditions simultaneously.

1.
$$\begin{aligned}
\Pr_{z \overset{\mathrm{r}}{\leftarrow} \{0,1\}^{m'}}\Big[z \notin \mathsf{A}^f(\mathrm{BC}(f, \boldsymbol{\mathcal{X}})\Big] &= 1 - \frac{\left|\mathrm{Supp}(\mathsf{A}^f(\mathrm{BC}(f, \boldsymbol{\mathcal{X}})))\right|}{[M']} \\
&\leq \mathrm{d}_{\mathsf{TV}}\Big(\mathsf{A}^f(\mathrm{BC}(f, \boldsymbol{\mathcal{X}}))\,,\, U_{m'}\Big) \\
&\leq \mathrm{d}_{\mathsf{TV}}\Big(\mathsf{A}^f(U_{n'})\,,\, U_{m'}\Big) + \mathrm{d}_{\mathsf{TV}}\Big(\mathrm{BC}(f, \boldsymbol{\mathcal{X}})\,,\, U_{m'}\Big) \\
&= \mathrm{d}_{\mathsf{TV}}\Big(\mathsf{A}^f(U_{n'})\,,\, U_{m'}\Big) + 1 - \frac{|\mathrm{BC}(f, \boldsymbol{\mathcal{X}})|}{[N']}
\end{aligned} \tag{3.12}$$

   Take the expectation over $(f, \vec{b}, \vec{\mathcal{X}})$ from F_{H} for Equation (3.12). By Lemma 3.3.1, Definition 3.2.1 and Corollary 3.4.4 we have

$$\Pr_{\substack{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}} \\ z \overset{\mathrm{r}}{\leftarrow} [M']}}\Big[z \notin \mathsf{A}^f(\mathrm{BC}(f, \boldsymbol{\mathcal{X}}))\Big] \leq \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}}}\Big[\mathrm{H}_{\mathsf{Sh}}(f) < \tau + 1\Big] + 2^{-k} + 2^{-0.3n} \tag{3.13}$$

$$\leq 2^{-0.9n} + 2^{-k} + 2^{-0.3n} \leq 2^{-0.2k} \tag{3.14}$$

   By the Markov inequality,

$$\Pr_{z \in [M']}\left[\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{H}}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \geq 1 - 2^{-0.1k}\right] \geq 1 - 2^{-0.1k}.$$

2. By Lemma 3.3.1 and Definition 3.2.1, we have

$$\begin{aligned}
&\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}, z \overset{\mathrm{r}}{\leftarrow} [M']}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \\
\leq\; &\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}, z \overset{\mathrm{r}}{\leftarrow} [M']}\Big[\exists w \in [N'], \mathsf{A}^f(w) = z\Big] \\
\leq\; &\Pr_{z \overset{\mathrm{r}}{\leftarrow} [M']}\Big[\exists w \in [N'], \mathsf{A}^f(w) = z \;\Big|\; \mathrm{H}_{\mathsf{Sh}}(f) \leq \tau - 1\Big] + \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}}\Big[\mathrm{H}_{\mathsf{Sh}}(f) > \tau - 1\Big] \\
\leq\; &2^{-k} + 2^{-0.9n} \leq 2^{-0.8k}.
\end{aligned}$$

   By the Markov inequality,

$$\Pr_{z \in [M']}\left[\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} F_{\mathsf{L}}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \leq 2^{-0.1k}\right] \geq 1 - 2^{-0.7k}.$$

3. Since $m' = n' + 3k$,

$$\mathop{\mathbb{E}}_{z\in[M']}\left[\left|\{w : \mathsf{A}^f(w) = z\}\right|\right] = 2^{n'} \cdot 2^{-m'} = 2^{3k}\,.$$

In particular,

$$\mathop{\mathbb{E}}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{\text{r}}F_{\mathsf{H}},z\in[M']}\left[\left|\{w : \mathsf{A}^f(w) = z\}\right|\right] = 2^{3k}.$$

By the Markov inequality,

$$\Pr_{z\xleftarrow{\text{r}}[M']}\left[\mathop{\mathbb{E}}_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{\text{r}}F_{\mathsf{H}}}\left[\left|\{w : \mathsf{A}^f(w) = z\}\right|\right] \le 2^{4k}\right] \ge 1 - 2^{-k}\,.$$

□

3.5.2 Entropy Reversal Lemma

We restate the Lemma 3.4.10 with simplified notation for clarity. Note that it is unnecessarily that $N = 2^n$ or being a power of two (and similarly for M, N' and M').

Lemma 3.5.1. *Let $\mathsf{A}^f : [N'] \to [M']$ be an algorithm making at most q oracle queries to $f : [N] \to [M]$. Let $F_{\mathsf{H}} = F_{1/2+5/n}$ and $F_{\mathsf{L}} = F_{1/2-5/n}$ be the distribution over a function $f : [N] \to [M]$, a partition $\vec{\mathcal{X}} \in ([N]^T)^S$ where $TS = N$, and the indication vector $\vec{b} \in \{0,1\}^S$ as defined in Section 3.3. Then for all $z \in [N]$,*

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{L}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\right] - \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}})\xleftarrow{r}F_{\mathsf{H}}}\left[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\right] \le \frac{O(q^2)}{S}.$$

Besides the parameters difference, a key difference between Lemma 3.5.1 and the key lemma in [LZ17] is that in our construction, the indicator vectors $\vec{b}$ consist of S independent Bernoulli random variables, while in their case, the number of ones, namely the Hamming weight is fixed. Formally, they consider the following distribution.

Definition 3.5.2. *For $i \in [S]$, $\widetilde{F}_i$ is the distribution over functions $f : [N] \to [M]$ and partitions $\vec{\mathcal{X}}$ defined as follows. Let $\vec{b_i} = (\underbrace{1,\ldots,1}_{i},\underbrace{0,\ldots,0}_{S-i})$. Then $(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_i$ denotes that $\vec{\mathcal{X}} \xleftarrow{r} \mathcal{X}_S$ and $f \xleftarrow{r} F(\vec{\mathcal{X}},\vec{b_i})$.*

A more direct analogue of the main lemma in [LZ17] with improved parameters stated using our notation:

Lemma 3.5.3. *Let $\mathsf{A}^f : [N'] \to [M']$ be an algorithm, which makes at most q queries to its oracle $f : [N] \to [M]$. If $2q^2 < i$, then for all $z \in \{0,1\}^{m'}$,*

$$\Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{i-1}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] - \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{i}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \le \frac{O(q^2)}{i^2}.$$

The improved parameters is also proved implicitly in [LZ17]. For completeness we also provide a (arguably simpler) proof of Lemma 3.5.3 below. Now we prove Lemma 3.5.1 using Lemma 3.5.3.

Proof of Lemma 3.5.1. By telescoping over i in Lemma 3.5.3, we get that for $\frac{1}{4} \le \alpha < \beta \le 1$ where αS and βS are integers, we have

$$\Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{\alpha S}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] - \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{\beta S}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big]$$
$$\le \frac{O\big(q^2(\beta - \alpha)\big)}{S}.$$

Conditioning on the Hamming weight of $\vec{b}$ being αS when we sample $\mathcal{D}_{1/2-5/n}$ or $\mathcal{D}_{1/2+5/n}$, the probability of the event $\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z$ is same as sampling from $\widetilde{F}_{\alpha S}$, because this event is invariant to permuting the indices of the S blocks, so the vector $\vec{b} = (\underbrace{1, \dots, 1}_{\alpha S}, \underbrace{0, \dots, 0}_{S-\alpha S})$ is equivalent to any other vector of the same Hamming weight. Hence, we have

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \mathcal{D}_{1/2\pm 5/n}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big]$$
$$= \sum_{h=0}^{S} \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{h}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \cdot \Pr\Big[\mathrm{Bin}(S, 1/2 \pm 5/n) = h\Big],$$

where Bin denotes the binomial distribution. By the Chernoff bound,

$$\Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \mathcal{D}_{\frac{1}{2}-\frac{5}{n}}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] - \Pr_{(f,\vec{b},\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \mathcal{D}_{\frac{1}{2}+\frac{5}{n}}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big]$$
$$\le 2^{-\Omega(s)} + \sum_{S/4<h<3S/4} \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \xleftarrow{r} \widetilde{F}_{h}}\Big[\exists w \in \mathrm{BC}(f, \boldsymbol{\mathcal{X}}), \mathsf{A}^f(w) = z\Big] \cdot \Pr\Big[\mathrm{Bin}(S, 1/2 + 5/n) = h\Big]$$

$$-\sum_{S/4<h<3S/4}\ \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}})\overset{\mathrm{r}}{\leftarrow}\widetilde{F}_h}\Big[\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z\Big]\cdot\Pr\Big[\mathrm{Bin}(S,1/2-5/n)=h\Big]\,.$$

Then by symmetry ($\Pr[\mathrm{Bin}(S,p)=h]=\Pr[\mathrm{Bin}(S,1-p)=s-h]$) and the bound we got at the beginning by telescoping, the difference is bounded by

$$\sum_{S/4<h<3S/4}\left(\Pr_{(f,\vec{\boldsymbol{\mathcal{X}}})\overset{\mathrm{r}}{\leftarrow}\widetilde{F}_h}[\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z]-\Pr_{(f,\vec{\boldsymbol{\mathcal{X}}})\overset{\mathrm{r}}{\leftarrow}\widetilde{F}_{S-h}}[\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z]\right)$$

$$\times\Pr[\mathrm{Bin}(S,1/2-5/n)=h]+2^{-\Omega(S)}[2]$$

$$\leq\sum_{S/4<h<S/2}\frac{O\left(q^2(S-2h)/S\right)}{S}\cdot\Pr[\mathrm{Bin}(S,1/2-5/n)=h]+2^{-\Omega(S)}\ \leq\ \frac{O(q^2)}{S}\,.$$

□

Proof of Lemma 3.5.3. Distributions $\widetilde{F}_{i-1}$ and $\widetilde{F}_i$ differ only on the block $\vec{\mathcal{X}}_i$. So an equivalent way to sample both distributions is that we can first sample the partition $\vec{\mathcal{X}}$, and the mapping except on the set X_i. In particular, we sample $\vec{\mathcal{Y}}_1,\ldots,\vec{\mathcal{Y}}_{i-1}\overset{\mathrm{r}}{\leftarrow}\mathbf{Y}_0$ and $\vec{\mathcal{Y}}_{i+1},\ldots,\vec{\mathcal{Y}}_S\overset{\mathrm{r}}{\leftarrow}\mathbf{Y}_1$. After that, for fixed $\vec{\mathcal{X}}$ and $\vec{\mathcal{Y}}_1,\ldots,\vec{\mathcal{Y}}_{i-1},\vec{\mathcal{Y}}_{i+1},\ldots,\vec{\mathcal{Y}}_S$, we sample $\vec{\mathcal{Y}}_i$ from $\mathbf{Y}_1$ or $\mathbf{Y}_0$ for distribution $\widetilde{F}_i$ or $\widetilde{F}_{i-1}$, respectively.

For notational convenience, we define

$$\begin{aligned}
\vec{\boldsymbol{\mathcal{X}}}_{-i}&\overset{\mathrm{def}}{=}(\vec{\mathcal{X}}_1,\ldots,\vec{X}_{i-1},\vec{X}_{i+1},\ldots,\vec{\mathcal{X}}_s)\\
\vec{\boldsymbol{\mathcal{X}}}_{\leq i}&\overset{\mathrm{def}}{=}(\vec{\mathcal{X}}_1,\ldots,\vec{\mathcal{X}}_i)\\
\vec{\boldsymbol{\mathcal{X}}}_{>i}&\overset{\mathrm{def}}{=}(\vec{\mathcal{X}}_{i+1},\ldots,\vec{X}_n)
\end{aligned}$$

We denote the difference of the probabilities by Δ_i to be

$$\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\left[\Pr_{\vec{\mathcal{Y}}_i\overset{\mathrm{r}}{\leftarrow}\mathbf{Y}_0}\Big[\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z\Big]\right]-\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\left[\Pr_{\vec{\mathcal{Y}}_i\overset{\mathrm{r}}{\leftarrow}\mathbf{Y}_1}\Big[\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z\Big]\right].\tag{3.15}$$

Conditioning on $\mathcal{X}_i$ being not queried, $\mathsf{A}^f(w)$ behaves identically under the two distributions. Thus, to compare two probabilities better, we refine the event $\exists w\in\mathrm{BC}(f,\boldsymbol{\mathcal{X}}),\mathsf{A}^f(w)=z$

based on the block $\mathcal{X}_i$. For given $f, \vec{\mathcal{X}}$ and z, we define the following events.

$$\begin{aligned}\forall j \in [T]\ ,\ E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j) &\overset{\text{def}}{=} \Big[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}) \text{ s.t. } \mathsf{A}^f(w) = z \wedge \vec{\mathcal{X}}_i(j) \in \mathrm{Query}_f(w)\Big] \\ E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) &\overset{\text{def}}{=} \Big[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}}) \text{ s.t. } \mathsf{A}^f(w) = z \wedge \mathrm{Query}_f(w) \cap X_i = \emptyset\Big]\,,\end{aligned}$$

where $\mathrm{Query}_f(w)$ denotes the set of the queries made by $\mathsf{A}^f(w)$ to the f with input w.

The main events that we care about is the union of the above events we defined, so for $\mathbf{Y} \in \{\mathbf{Y}_0, \mathbf{Y}_1\}$

$$\begin{aligned}\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}}\Big[\exists w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}})\Big] &= \Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}}\Big[E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \vee \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big] \\ &= \Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}}\Big[E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp)\Big] + \Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big].\end{aligned}$$

An important observation is that the event $E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp)$ does not depend on $f(X_i)$, so sampling $\vec{\mathcal{Y}}_i$ from $\mathbf{Y}_0$ or $\mathbf{Y}_1$ does not affect the probability of the event. Hence, Equation (3.15) can be written as

$$\begin{aligned}\Delta_i = &\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Big[\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_0}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big]\Big] \\ &- \mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Big[\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_1}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big]\Big].\end{aligned}$$

Now, for the probability over $\mathbf{Y}_0$ part, we apply the union bound.

$$\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Big[\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_0}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big]\Big] \le \mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Bigg[\sum_{j=1}^{t}\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_0}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big]\Bigg]$$

For the $\mathbf{Y}_1$ part, we bound the probability via the inclusion-exclusion principle.

$$\begin{aligned}&\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Big[\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_1}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge \Big(\textstyle\bigvee_{j=1}^{t} E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big)\Big]\Big] \\ \ge &\mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Bigg[\sum_{j=1}^{t}\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_1}\Big[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\Big]\Bigg] \\ &- \mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\Big[\Pr_{\vec{\mathcal{Y}}_i \xleftarrow{r} \mathbf{Y}_1}\Big[\exists j \ne j', \neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j')\Big]\Big]\end{aligned}$$

Observe that $\mathsf{A}^f(w)$ only queries $\mathcal{X}_i$ at most once for all $w \in \mathrm{BC}(f,\boldsymbol{\mathcal{X}})$, and the marginal distributions of the mapping on $\vec{\mathcal{X}}_i(j)$ for every $j \in [T]$ are the same in both $\mathbf{Y}_1$ and $\mathbf{Y}_0$ cases, so for every $j \in [T]$

$$\Pr_{\vec{\mathcal{Y}}_i \overset{\mathrm{r}}{\leftarrow} \mathbf{Y}_0}\left[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\right] = \Pr_{\vec{\mathcal{Y}}_i \overset{\mathrm{r}}{\leftarrow} \mathbf{Y}_1}\left[\neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j)\right]$$

Therefore, the difference between two cases is bounded as

$$\begin{aligned}\Delta_i &\leq \mathop{\mathbb{E}}_{\vec{\boldsymbol{\mathcal{Y}}}_{-i},\vec{\boldsymbol{\mathcal{X}}}}\left[\Pr_{\vec{\mathcal{Y}}_i \overset{\mathrm{r}}{\leftarrow} \mathbf{Y}_1}\left[\exists j \neq j', \neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j')\right]\right] \\ &= \Pr_{(f,\vec{\boldsymbol{\mathcal{X}}}) \overset{\mathrm{r}}{\leftarrow} \widetilde{F}_i}\left[\exists j \neq j', \neg E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j) \wedge E_{f,\vec{\boldsymbol{\mathcal{X}}},z}(j')\right]. \qquad (3.16)\end{aligned}$$

To bound the term, we consider another way to sample $(f, \vec{\boldsymbol{\mathcal{X}}})$ from $\widetilde{F}_i$. Given $(f, \vec{\boldsymbol{\mathcal{X}}})$, we define the function Blo : $\boldsymbol{\mathcal{X}}_{\leq i} \to [i]$ by

$$\mathrm{Blo}(x) = \text{the block that } x \text{ is in} = \text{the unique } i' \leq i \text{ s.t. } \exists j, \vec{\mathcal{X}}_{i'}(j) = x.$$

We will re-sample the "block structure for $\boldsymbol{\mathcal{X}}_{\leq i}$" after getting $(f, \vec{\boldsymbol{\mathcal{X}}})$. Namely, we will sample Blo given fixed f and $\vec{\boldsymbol{\mathcal{X}}}_{>i}$) using principle of deferred decisions. Note that conditioned on f and $\vec{\boldsymbol{\mathcal{X}}}_{>i}$, Blo is a uniformly random regular mapping from $\boldsymbol{\mathcal{X}}_{\leq i}$ to $[i]$ where regular means that all preimage sets $B^{-1}(i')$ are of size t.

Along the way of sampling Blo, we abuse the notation and consider a more general block assignment Blo : $\vec{\boldsymbol{\mathcal{X}}}_{\leq i} \to [i] \cup \{*\}$ where "$*$" represent values not yet determined as before. Initially, $\mathrm{Blo}(x) = *$ for all $x \in \vec{\mathcal{X}}_{\leq i}$. For an input $w \in [N']$, we say w is *partially block compatible*, written as $w \in \mathrm{PBC}(f, \vec{\mathcal{X}}_{>i}, \mathrm{Blo})$ if $\mathsf{A}^f(w)$ queries each block (defined by $\vec{\mathcal{X}}_{>i}$ or Blo) at most once (among the queries whose block is determined).

The procedure for sampling Blo given fixed f and $\vec{\mathcal{X}}_{>i}$ is as follows.

Procedure 3.5.1

1. Set $\mathrm{Blo}(x) = *$ for all $x \in [N] \setminus \mathbf{X}_{>i}$.
2. While $\exists w$ s.t. $w \in \mathrm{PBC}(f, \vec{\mathcal{X}}_{>i}, \mathrm{Blo})$ and $\mathsf{A}^f(w) = z$,
    (a) Randomly assign Blo on undetermined element of $\mathrm{Query}_f(w)$ conditional on assignment to Blo so far. That is, for each $x \in \mathrm{Query}_f(w)$ s.t. $\mathrm{Blo}(x) = *$ set $\mathrm{Blo}(x) = i'$ with probability $\frac{T - |\{x' : \mathrm{Blo}(x') = i'\}|}{iT - |\{x' : \mathrm{Blo}(x') \neq *\}|}$.
3. Randomly assign Blo on all undetermined elements conditioned on assignment to Blo so far.

By considering the above sampling procedure, let w_ℓ be the value of w chosen in the ℓ-th iteration of the while loop (Step 2). Then we define the following events for $\ell \in \mathbb{N}$.

$$\begin{aligned}
E_\ell^{(0)} =& [\text{None of the new assignments to Blo in } \ell\text{-th iteration equal } i \\
& \qquad \wedge \text{ after } \ell\text{-th iteration, } w_\ell \neq \mathrm{PBC}(f, \vec{\mathcal{X}}_{>i}, \mathrm{Blo}).] \\
E_\ell^{(1)} =& [\text{Exactly one of the new assignments to Blo in } \ell\text{-th iteration equals } i.] \\
E_\ell^{(\geq 2)} =& [\text{At least two of the new assignments to Blo in } \ell\text{-th iteration equals } i.]
\end{aligned}$$

Denote the assignments of Blo after the first $\ell - 1$ rounds in the while loop as $\mathrm{Blo}_{\ell-1}$. Suppose $q\ell \leq iT/2$ and $q^2 \leq i/2$, then we have

$$\begin{aligned}
\Pr\Big[E_\ell^{(0)} \mid \mathrm{Blo}_{\ell-1}\Big] &\leq \binom{q}{2} \cdot \frac{T}{iT - q \cdot (\ell - 1)} \leq \frac{1}{2} \\
\Pr\Big[E_\ell^{(1)} \mid \mathrm{Blo}_{\ell-1}\Big] &\leq q \cdot \frac{T}{iT - q \cdot (\ell - 1)} \leq \frac{2q}{i} \\
\Pr\Big[E_\ell^{(\geq 2)} \mid \mathrm{Blo}_{\ell-1}\Big] &\leq \sum_{j=2}^{q} \binom{q}{j} \cdot \left(\frac{T}{iT - q \cdot (\ell - 1)}\right)^j \leq \frac{2q}{i^3} \\
&\leq \sum_{j=2}^{q} \left(\frac{q}{2}\right)^j \left(\frac{2}{i}\right)^j \leq \sum_{j=2}^{q} \left(\frac{q}{i}\right)^j \leq \frac{2q^2}{i^2}
\end{aligned} \tag{3.17}$$

Let L be a parameter to be chosen later. The event in Equation (3.16) happens only if the event $E^{(1)}$ happens at least twice in the while loop and for the rest of the while loops, $E^{(0)}$ or $E^{(\geq 2)}$ happens. We focus on the sampling procedure for the first L rounds. Then Equation (3.16) can be bounded as

$$\begin{aligned}
&\Pr_{(f,\vec{\boldsymbol{x}})\xleftarrow{r}\widetilde{F}_i}\left[\exists j \neq j', \neg E_{f,\vec{\boldsymbol{x}},z}(\perp) \wedge E_{f,\vec{\boldsymbol{x}},z}(j) \wedge E_{f,\vec{\boldsymbol{x}},z}(j')\right] \\
\leq &\sum_{0<\ell_1<\ell_2\leq L} \Pr\left[\left(E_1^{(0)} \wedge \cdots \wedge E_{\ell_1-1}^{(0)}\right) \wedge E_{\ell_1}^{(1)} \wedge \left(E_{\ell_1+1}^{(0)} \wedge \cdots \wedge E_{\ell_2-1}^{(0)}\right) \wedge E_{\ell_2}^{(1)}\right] \qquad (3.18)\\
&+ \sum_{0<\ell\leq L} \Pr\left[\left(F_1 \wedge \cdots \wedge E_{\ell-1}^{(0)}\right) \wedge E_{\ell}^{(1)} \wedge \left(E_{\ell+1}^{(0)} \wedge \cdots \wedge E_{L}^{(0)}\right)\right] \qquad (3.19)\\
&+ \sum_{0<\ell\leq L} \Pr\left[\left(E_1^{(0)} \wedge \cdots \wedge E_{\ell-1}^{(0)}\right) \wedge E_{\ell}^{(\geq 2)}\right] \qquad (3.20)\\
&+ \Pr\left[E_1^{(0)} \wedge \cdots , \wedge E_L^{(0)}\right] \qquad (3.21)
\end{aligned}$$

As long as $qL \leq iT/2$, we can bound Equation (3.18), (3.19), (3.20) and (3.21) using Equation 3.17:

$$\begin{aligned}
\text{Equation (3.18)} &\leq \sum_{0<\ell_1<\ell_2\leq L} \frac{1}{2^{\ell_2-2}} \cdot \frac{4q^2}{i^2} \leq 16 \sum_{0<\ell_2\leq L} \frac{\ell_2}{2^{\ell_2}} \cdot \frac{q^2}{i^2} = O\left(\frac{q^2}{i^2}\right) \\
\text{Equation (3.19)} &\leq \sum_{0<\ell\leq L} \frac{1}{2^{L-1}} \cdot \frac{2q}{i} = O\left(2^{-L}\right) \\
\text{Equation (3.20)} &\leq \sum_{0<\ell\leq L} \frac{1}{2^{\ell-1}} \cdot \frac{2q^2}{i^2} = O\left(\frac{q^2}{i^2}\right) \\
\text{Equation (3.19)} &\leq O\left(2^{-L}\right)
\end{aligned}$$

If we choose $L = 2\log(i/q)$ (which satisfies $qL \leq iT/2q$), then all Equation (3.18), (3.19), (3.20) and (3.21) is at most $O\left(q^2/i^2\right)$, and so is Δ_i. □

3.5.3 From flattening to SDU algorithm (Lemma 3.2.2)

Lemma 3.2.2 (restatement)**.** *If there exists a (ε, Δ)-flattening algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q, then there exists a k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n''} \to \{0,1\}^{n''-3k}$ where $n'' = O(n'+m')$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q and $k = \Omega\big(\min\{\Delta, \log(1/\varepsilon)\}\big)$. In particular, there exists such a k-SDU* *algorithm with query complexity* $O(k \cdot \min\{n, m\}^2)$.

Proof. The Lemma follows directly by chaining Claim 3.5.4 and 3.5.5. □

Claim 3.5.4. *If there exists a* (ε, Δ)*-flattening algorithm* $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ *for function* $f : \{0,1\}^n \to \{0,1\}^m$ *with query complexity* q*, then there exists an* k*-SDU algorithm* $\mathsf{B}^f : \{0,1\}^{n''} \to \{0,1\}^{m''}$ *where* $n'' = O(n' + m')$ *and* $m'' = O(n' + m')$ *for function* $f : \{0,1\}^n \to \{0,1\}^m$ *with query complexity* q *and* $k = \Omega(\min\{\Delta, \log(1/\varepsilon)\})$.

Proof. This proof mostly follows the idea in [GSV99b]. It suffices to prove the existence of $\Omega(k)$-SDU algorithm for $k = \min\{\Delta, \log(1/\varepsilon)\}$. Let $\mathcal{H}_{a,b}$ be a family of 2-universal hash function from a bits to b bits. We sample hash functions h_1 and h_2 from $\mathcal{H}_{m',\kappa}$ and $\overset{r}{\leftarrow} \mathcal{H}_{n',n'-\kappa-k/3}$, respectively, where κ is the parameter chosen by the flattening algorithm A^f. We will show that

$$\mathsf{B}^f(w, h_1, h_2) = (h_1, h_1(\mathsf{A}^f(w)), h_2, h_2(w))$$

is a $\Omega(k)$-SDU algorithm. We denote the output of $\mathsf{B}^f(w, h_1, h_2)$ as a jointly distributed random variables (H_1, Z_1, H_2, Z_2) when $w \overset{r}{\leftarrow} U_{n'}, h_1 \overset{r}{\leftarrow} \mathcal{H}_{m',\kappa}$ and $h_2 \overset{r}{\leftarrow} \mathcal{H}_{n',n'-\kappa-k/3}$.

1. When $(f, \tau) \in$ Entropy-Approximation$_Y$, there exists a distribution Z_H with $\mathrm{H}_{\mathsf{Sh}}(Z_H) \geq \kappa + \Delta$ such that $\mathrm{d}_{\mathsf{TV}}(\mathsf{A}^f(U_{n'}), Z_H) \leq \varepsilon$. First, we show that (H_1, Z_1) is close to uniform. By the Leftover Hash Lemma, $\mathrm{d}_{\mathsf{TV}}((H_1, H_1(Z_H), (H_1, U_\kappa)) \leq 2^{-\Delta/3}$, and so

$$\begin{aligned}\mathrm{d}_{\mathsf{TV}}((H_1, Z_1), (H_1, U_\kappa)) \leq& \mathrm{d}_{\mathsf{TV}}(\mathsf{A}^f(U_{n'}), Z_H) + \mathrm{d}_{\mathsf{TV}}((H_1, H_1(Z_H)), (H_1, U_\kappa)) \\ \leq& 2^{-\Delta/3} + \varepsilon \leq 2^{-\Omega(k)}.\end{aligned}$$

For the (H_2, Z_2) of part, we will show that with high probability over sampling (h_1, z_1) from (H_1, Z_1), the distribution (H_2, Z_2) conditioned on (h_1, z_1) is close to uniform. Since (H_1, Z_1) is $2^{-\Omega(k)}$-close to uniform, by the Markov inequality, with probability at least $1 - 2^{-\Omega(k)}$ over choosing (h_1, z_1) from (H_1, Z_1), we have

$$\Pr\Big[h_1(\mathsf{A}^f(U_{n'})) = z_1\Big] = \Pr\Big[Z_1 = z_1 \Big| H_1 = h_1\Big] \geq \frac{1}{2} \cdot 2^{-\kappa}.$$

Thus, except for $2^{-\Omega(k)}$ probability over (h_1, z_1), the number of w such that $h_1(\mathsf{A}^f(w)) = z_1$ is at least $2^{n'-\kappa-1}$. Again, by the Leftover Hash Lemma, (H_2, Z_2) is $2^{-\Omega(k)}$-close to uniform conditioned on any such (h_1, z_1). We then can conclude that (H_1, Z_1, H_2, Z_2) is $2^{-\Omega(k)}$-close to uniform.

2. When $(f, \tau) \in$ Entropy-Approximation, there exists a distribution Z_L with $\mathrm{H}_{\mathsf{max}}(Z_L) \leq \kappa - \Delta$ such that $\mathrm{d}_{\mathsf{TV}}(\mathsf{A}^f(U_{n'})\,, Z_L) \leq \varepsilon$. For every fixed h_1 and h_2, we will bound the support size of (Z_1, H_2, Z_2) conditioned on $H_1 = h_1$ and $H_2 = h_2$. We divide $\mathrm{Supp}(Z_1, Z_2)$ into three subset according to $z_1 \in \mathrm{Supp}(Z_1)$.

$$\begin{cases} \mathcal{S}_1 = \{(z_1, z_2) : z_1 \in \mathrm{Supp}(Z_L)\} \\ \mathcal{S}_2 = \{(z_1, z_2) : \Pr[Z_1 = z_1] \geq 2^{-\kappa-2k/3} \text{ and } z_1 \notin \mathrm{Supp}(Z_L)\} \\ \mathcal{S}_3 = \{(z_1, z_2) : \Pr[Z_1 = z_1] < 2^{-\kappa-2k/3} \text{ and } z_1 \notin \mathrm{Supp}(Z_L)\} \end{cases}$$

Since, $\mathrm{Supp}(Z_1, Z_2) = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3$, it suffices to show that $|\mathcal{S}_i| \leq 2^{-\Omega(k)} \cdot |\{0,1\}^\kappa \times \{0,1\}^{n'-\kappa-k/3}|$ for all $i = 1, 2, 3$.

(a) For $\mathcal{S}_1$, by definition, $\mathrm{H}_{\mathsf{max}}(Z_L) \leq \kappa - \Delta$ implies that $|\mathrm{Supp}(Z_L)|/|\{0,1\}^\kappa| \leq 2^{-\Delta}$, and so

$$|\mathcal{S}_1| \leq 2^{-\Delta} \cdot \left|\{0,1\}^\kappa \times \{0,1\}^{n'-\kappa-k/3}\right| \leq 2^{-\Omega(k)} \cdot \left|\{0,1\}^\kappa \times \{0,1\}^{n'-\kappa-k/3}\right|.$$

(b) For $\mathcal{S}_2$, since $\mathrm{d}_{\mathsf{TV}}(\mathsf{A}^f(U_{n'})\,, Z_L) \leq \varepsilon$, $\sum_{z_1 \notin \mathrm{Supp}(Z_L)} \Pr[Z_1 = z_1] \leq \varepsilon$. Each z_1 such that $\Pr[Z_1 = z_1] \geq 2^{-\kappa-2k/3}$ contributes at least $2^{-\kappa-2k/3}$ towards ε, so

$$\left|\{z_1 : \Pr[Z_1 = z_1] \geq 2^{-\kappa-2k/3} \text{ and } z_1 \notin \mathrm{Supp}(Z_L)\}\right| \leq \varepsilon \cdot 2^{\kappa+2k/3}.$$

Then we have $|\mathcal{S}_2| \leq 2^{-\Omega(k)} \cdot |\{0,1\}^\kappa \times \{0,1\}^{n'-\kappa-k/3}|$, since $k \leq \log(1/\varepsilon)$.

(c) For $\mathcal{S}_3$, if $\Pr[Z_1 = z_1] < 2^{-\kappa-2k/3}$, then the number of $w \in \{0,1\}^{n'}$ such that $h_1(\mathsf{A}^f(w)) = z_1$ is at most $2^{n'-\kappa-2k/3}$, which is at most a $2^{-k/3}$ fraction of $\{0,1\}^{n'-\kappa-k/3}$. Therefore, $|\mathcal{S}_3| \leq 2^{-\Omega(k)} \cdot |\{0,1\}^\kappa \times \{0,1\}^{n'-\kappa-k/3}|$.

Thus, we conclude that B^f is a $\Omega(k)$-SDU algorithm. □

Claim 3.5.5. *If there exists a k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q, then there exists an $(k-1)$-SDU algorithm $\mathsf{B}^f : \{0,1\}^{n''} \to \{0,1\}^{m''}$ where $n'' = O(n')$ and $m'' = n'' - 3k$ for function $f : \{0,1\}^n \to \{0,1\}^m$ with query complexity q.*

Proof.

Lemma 3.5.6 (Generalized Leftover Hash Lemma [DORS08, Lemma 2.1])**.** *Let (X, Y) be a jointed distributed random variables such that $\mathrm{H}_{\mathsf{Sh}}(X|Y) \geq k$. Let $\mathcal{H}_{n,m} = \{h : \{0,1\}^n \to \{0,1\}^m\}$ be a family of universal hash function where h can be described in $(n+m)$ bits and $m = k - 2\log(1/\varepsilon) + 2$. Then*

$$\mathrm{d}_{\mathsf{TV}}((h(X), Y, h)\,,\,(U_m, Y, h)) \leq \varepsilon$$

where U_m is a uniform m bits string.

Let $\mathcal{H}_{n',n'-m'-3k} = \{h : \{0,1\}^n \to \{0,1\}^m\}$ be a family of universal hash function where h can be described in $d = 2n' - m' - 3k$ bits. Based on the given k-SDU algorithm $\mathsf{A}^f : \{0,1\}^{n'} \to \{0,1\}^{m'}$, we define the algorithm $\mathsf{B}^f : \{0,1\}^{n'+d} \to \{0,1\}^{n'+d-3k}$ as

$$\mathsf{B}^f(w, h) \overset{\text{def}}{=} (\mathsf{A}^f(w), h(w), h)\,.$$

By the chain rule of average min-entropy ([DORS08, Lemma 2.2b])

$$\mathrm{H}_{\mathsf{Sh}}(w|\,\mathsf{A}(w)) \geq \mathrm{H}_{\mathsf{Sh}}(w) - \mathrm{len}(\mathsf{A}(w)) = n' - m'\,,$$

and hence

$$\mathrm{d}_{\mathsf{TV}}\Big((\mathsf{A}(w), \mathsf{Ext}(w, v))\,,\,(\mathsf{A}(w), U_{n'-m'+d-2k-O(1)})\Big) \leq 2^{-k}\,.$$

Therefore, when $\mathrm{H}_{\mathsf{Sh}}(f) \geq \tau + 1$

$$\begin{aligned}
&\mathrm{d}_{\mathsf{TV}}(\mathsf{B}^f(U_{n'+d})\,,\,U_{n'+d-3k}) \\
&= \mathrm{d}_{\mathsf{TV}}((\mathsf{A}^f(w), h(w), h)\,,\,(U_{m'}, U_{n'-m'+d-3k})) \\
&= \mathrm{d}_{\mathsf{TV}}(\mathsf{A}^f(w)\,,\,U_{m'}) + \mathrm{d}_{\mathsf{TV}}((\mathsf{A}^f(w), h(w), h)\,,\,(\mathsf{A}^f(w), U_{n'-m'+d-3k})) \\
&\leq 2^{-k} + 2^{-k} = 2^{-(k-1)}\,.
\end{aligned}$$

The last inequality is by the property of k-SDU algorithm and Lemma 3.5.6.

On the other hand, if $\mathrm{H}_{\mathsf{Sh}}(f) \leq \tau - 1$,

$$\left|\mathrm{Supp}(\mathsf{B}^f(U_{n'+d}))\right| \leq 2^{m'-k} \cdot 2^{n'-m'+d-3k} \leq 2^{(n'+d-3k)-k} .$$

Therefore, B^f is an $(k-1)$-SDU algorithm with desired parameter. □

Chapter 4

Simulating Auxiliary Input

In this Chapter, we study the complexity of the problem "simulating auxiliary input" defined in [JP14]. We construct a simulator with complexity better than all previous results and prove the optimality up to logarithmic factors by establishing a black-box lower bound. Specifically, let ℓ be the length of the auxiliary input and ε be the indistinguishability parameter. Our simulator is $\widetilde{O}(2^{\ell}\varepsilon^{2})$ more complicated than the distinguisher family. For the lower bound, we show the relative complexity to the distinguisher of a simulator is at least $\widetilde{\Omega}(2^{\ell}\varepsilon^{-2})$ assuming the simulator is restricted to use the distinguishers in a black-box way and satisfy a mild restriction.

4.1 Introduction

In the leakage simulation lemma, we are given a joint distribution (X, Z) where Z is "short". The goal is to find an "low complexity" simulator h such that (X, Z) and $(X, \mathsf{h}(X))$ are indistinguishable by a family of distinguishers. The non-triviality comes from the efficiency requirement. Otherwise, one can simply hardcode the conditional distribution Z given x for all x.

Theorem 4.1.1 (Leakage Simulation Lemma, informal)**.** *Let $\mathcal{F}$ be a family of deterministic distinguisher from $\mathcal{X} \times \{0,1\}^{\ell}$. There exists a simulator function $\mathsf{h} : \{0,1\}^{n} \to \{0,1\}^{\ell}$ with* *relative complexity* $\mathrm{poly}(2^\ell, \varepsilon^{-1})$ *relative to* $\mathcal{F}$[1] *such that for all* $f \in \mathcal{F}$*,*

$$\Big|\Pr[f(X,Z)=1] - \Pr[f(X,\mathsf{h}(X))]\Big| \le \varepsilon\,.$$

The Leakage Simulation Lemma implies many theorems in computational complexity and cryptography. For instance, Jetchev and Pietrzak [JP14] used the lemma to give a simpler and quantitatively better proof for the leakage-resilient stream-cipher[DP08]. Also, Chung, Lui, and Pass [CLP15] applied the lemma (also an interactive version) to study connections between various notions of Zero-Knowledge. Moreover, the leakage simulation lemma can be used to deduce the technical lemma of Gentry and Wichs [GW11] (for establishing lower bounds for succinct arguments) and the Leakage Chain Rule[JP14] for relaxed HILL min-entropy [HILL99].

Before Jetchev and Pietrzak formally described the Leakage Simulation Lemma as in Theorem 4.1.1, Trevisan, Tulsiani and Vadhan proved a similar lemma called Regularity Lemma [TTV09], which can be viewed as a special case of the Leakage Simulation Lemma by restricting the family of distinguishers in certain forms. In [TTV09], they also showed that all Dense Model Theorem [RTTV08], Impagliazzo's Hardcore Lemma [Imp95], and Weak Szemérdi Regularity Lemma [FK99] can be derived from the Regularity Lemma. That means the Leakage Simulation Lemma also implies all those theorems.

As the Leakage Simulation Lemma has many implications, it is essential to understand what is the best relative complexity of h to $\mathcal{F}$ in terms of 2^ℓ and ε^{-1} that we can achieve and how to achieve. In particular, the provable security level of a leakage-resilient stream-cipher can be improved significantly using better complexity bound for Leakage Simulation Lemma. We provide an improved complexity bound $\widetilde{O}(2^\ell \varepsilon^{-2})$ and also show that it is almost optimal.

[1] The "relative complexity" means the circuit complexity of h when it has an access to oracle gates that compute functions in $\mathcal{F}$.

4.1.1 Upper bound

In [TTV09], they provided two different approaches for proving the Regularity Lemma. One is by the Non-uniform Min-max Theorem, and another one is via boosting. Although it is not known whether the Regularity Lemma implies the Leakage Simulation Lemma directly, [JP14] adopted both techniques and used them to show the Leakage Chain Rule with relative complexity bound $\tilde{O}(2^{4\ell}\varepsilon^{-4})$[2]. Later on, Vadhan and Zheng derived the Leakage Simulation Lemma [VZ13a, Lemma 6.8] using so-called "Uniform Min-max Theorem", which is proved via multiplicative weight update (MWU) method incorporating with KL-projections. The circuit complexity of the simulator they got is $\widetilde{O}(t \cdot 2^{\ell}\varepsilon^{-2} + 2^{\ell}\varepsilon^{-4})$ where t is the size of the distinguisher circuits. After that, Skórski also used a boosting-type method to achieve the bound $\widetilde{O}(2^{5\ell}\varepsilon^{-2})$ [Sko16a]. It was further improved to $\widetilde{O}(2^{3\ell}\varepsilon^{-2})$ later by another boosting-type algorithm called subgradient method [Sko16b]. Note that the complexity bound in [VZ13a] has an additive term, so their result is incomparable to the others.

Our Results. In this paper, we achieve $\widetilde{O}(2^{\ell}\varepsilon^{-2})$ for the relative complexity, which contains the best components out of two worlds. The core algorithm in our proof is also the multiplicative weight update (MWU) method as in [VZ13b]. The additive term $2^{\ell}\varepsilon^{-4}$ in [VZ13a] is due to the precision issue when performing multiplication of real numbers. The saving of the additive term is based on the observation mentioned in [VZ13a] — the KL-projection step in their MWU algorithm is not necessary when proving the Leakage Simulation Lemma. We make use of that and prove that proper truncations on weights help in reducing the circuit complexity, yet accurate enough for h to simulate the auxiliary input.

Implication in leakage-resilient cryptography Leakage Simulation Lemma can be used to prove the security of Pietrzak's leakage-resilient stream-cipher [Pie09]. However, the previous security proofs suffer from the term ε^{-4} or $2^{3\ell}$ (additively or multiplicative) in the complexity bound for the Leakage Simulation Lemma mentioned above. In particular, in

[2] In the original paper, they claimed to achieve the bound $\widetilde{O}(2^{3\ell}\varepsilon^{-2})$. However, Skórski pointed out some analysis flaws [Sko16a].

Literature	Techinque	Complexity of simulator
[JP14]	Min-max / Boosting	$t \cdot \widetilde{O}(2^{4\ell}\varepsilon^{-4})$
[VZ13a]	Boosting with KL-projection	$t \cdot \widetilde{O}(2^{\ell}\varepsilon^{-2}) + \widetilde{O}(2^{\ell}\varepsilon^{-4})$
[Sko16a]	Boosting with self-defined projection	$t \cdot \widetilde{O}(2^{5\ell}\varepsilon^{-2})$
[Sko16b]	Boosting with Subgradient Method	$t \cdot \widetilde{O}(2^{3\ell}\varepsilon^{-2})$
This work	Boosting	$t \cdot \widetilde{O}(2^{\ell}\varepsilon^{-2})$
	Black-box lower bound	$t \cdot \Omega(2^{\ell}\varepsilon^{-2})$

Table 4.1: *Summary of exisiting upper bound results and our results.*

some legitimate examples mentioned by [Sko16a], the security parameters obtained from the complexity bounds provided in [JP14] and [VZ13a] only guarantee trivial security when ε is set to be 2^{-40}. On the other hand, the factor $2^{3\ell}$ (or even $2^{5\ell}$) is significant and makes the guaranteed security bound trivial when the leakage is longer. Consider the following concrete settings. We use the block cipher AES-256 as the underlying block cipher for Dziembowski and Pietrzak's construction. Suppose the target security to be 2^{-40} and the stream cipher runs for 16 rounds. Let the complexity bound for the Leakage Simulation Lemma is $2^{a\ell}/\varepsilon^2$, then it can prove to against adversarial circuit of size $\approx 2^{84}/2^{(a+1)\ell}$ [JP14, Lemma 2]. One can see that only our complexity bound ($a = 1$) can provide a non-trivial security guarantee.

4.1.2 Lower bound

Our results. We show that the simulator must have the relative complexity $\Omega(2^{\ell}\varepsilon^{-2})$ to the distinguisher family by establishing a black-box lower bound, where a simulator can only use the distinguishers in a black-box way. Our lower bound requires an additional assumption that the simulator on a given input x, does not make a query an $x' \neq x$ to distinguishers.[3] Querying at points different from the input does not seem helpful. Indeed, all the known upper bound algorithms (including the one in this work) satisfy the above assumption. Still,

[3] Many black-box lower bounds in related contexts [LTW11, Zha11, PS16] make the same mild assumption (implicitly). See Section 4.3.2 for more details.

we leave it as an open problem to formalize this intuition and close this gap completely.

Comparison to related results. In [JP14], they proved a $\Omega(2^\ell)$ lower bound for relative complexity under a hardness assumption for one-way functions. Besides, there are also lower bound results on the theorems that implied by the Leakage Simulation Lemma, including Regularity Lemma [TTV09], Hardcore Lemma [LTW11], Dense Model Theorem [Zha11], Leakage Chain Rule [PS16] and Hardness Amplification [SV10, AS14]. The best lower bound one can obtain before this work is $\Omega(\varepsilon^{-2})$ [LTW11, SV10, Zha11] and $\Omega(2^\ell\varepsilon^{-1})$ [PS16]. Thus our lower bound is the first tight lower bound $\Omega(2^\ell\varepsilon^{-2})$ for Leakage Simulation Lemma. See Section 4.3.2 for more detailed comparisons.

4.2 Efficient Simulating Auxiliary Inputs

The formal description of Leakage Simulation Lemma with our improved parameters is as follows.

Theorem 4.2.1 (Leakage Simulation Lemma)**.** *Let $n, \ell \in \mathbb{N}$, $\varepsilon > 0$ and $\mathcal{F}$ be a collection of deterministic distinguishers $f : \{0,1\}^n \times \{0,1\}^\ell \to \{0,1\}$. For every joint distribution (X, Z) on $\{0,1\}^n \times \{0,1\}^\ell$, there exists a (randomized) simulation circuit $\mathsf{h} : \{0,1\}^n \to \{0,1\}^\ell$ such that*

1. *h has complexity $\widetilde{O}(2^\ell\varepsilon^{-2})$ relative to $\mathcal{F}$. That is, h can be computed by an oracle-aided circuit of size $\widetilde{O}(2^\ell\varepsilon^{-2})$ with oracle gates computing functions in $\mathcal{F}$.*

2. *(X, Z) and $(X, \mathsf{h}(X))$ are ε-indistinguishable by $\mathcal{F}$. That is, for every $f \in \mathcal{F}$,*

$$\left| \mathop{\mathbb{E}}_{(x,z)\xleftarrow{r}(X,Z)} [f(x,z)] - \mathop{\mathbb{E}}_{\mathsf{h},x\xleftarrow{r}X} [\mathcal{F}(x,\mathsf{h}(x))] \right| \le \varepsilon\,.$$

In particular, let $\mathcal{F}$ be a set of boolean circuits of size at most t, then we have the following corollary.

Corollary 4.2.2. *Let $s, n, \ell \in \mathbb{N}$ and $\varepsilon > 0$. For every distribution (X, Z) over $\{0,1\}^n \times \{0,1\}^\ell$, there exists a simulator circuit h of size $t' = t \cdot \widetilde{O}(2^\ell \varepsilon^{-2})$ such that (X, Z) and $(X, \mathsf{h}(X))$ are (t, ε)-indistinguishable.*

The proofs of Leakage Simulation Lemma [JP14, Sko16a, Sko16b] are purely based on the Nonuniform Min-max Theorem or some "boosting-type" argument. In order to apply the Nonuniform Min-max Theorem, we have to Approximate a distribution over circuits by a small circuit (see Section 5.5.2 for an example), which incurs the ε^{-2} factor in the complexity additional to the ε^{-2} from the Nonuniform Min-Max Therefore, boosting-type of proofs are more favorable, which we also adopt in this work.

Proofs via boosting. The structure of a boosting-type proof for constructing h is as follows:

```
Meta Algorithm 4.2.1: Boosting-type Simulator h
Input: $x \in \{0,1\}^n$
  1. Maintain the weights $\{w_z\}_{z \in [2^\ell]}$ and let h be a randomized function such that
     $\Pr[\mathsf{h}(x) = z] \propto w_z$
  2. For $i = 1 \to T$
      (a) Let $f \in \mathcal{F}$ be some function that violates the indistinguishability require-
          ment. (or the one distinguishes $(X, Z)$ and $(X, \mathsf{h}(X))$ the most)
      (b) Use $f$ to update the weights $\{w_z\}$ (and h).
```

Basically, it keep updating h until the first requirement (indistinguishability) is satisfied. Th keys to have a small complexity h are how to bound the number of round and how to update h efficiently.

Starting from [TTV09], then followed by [JP14] and [Sko16a], they use *additive* update on the probability mass function of each $\mathsf{h}(x)$. However, additive update may cause negative weights, so it takes extra efforts (both algorithm-wise and complexity-wise) to fix it. Vadhan and Zheng use multiplicative weight update instead [VZ13a], which not only avoids the issue above but also converges faster. However, the number of bits to represent weights increases drastically after multiplications, and that causes the $O(2^\ell \varepsilon^{-4})$ additive term in the complexity. Since the backbone of our algorithm is same as in [VZ13a], we review their idea first in the next section, and then show how the additive term can be eliminated in Section 4.2.2.

4.2.1 Simulate leakage with multiplicative weight update

In this section, we review that by using the MWU algorithm as the update rule, we obtain an algorithm with low round complexity. It is convenient to think Z as a randomized function of X. That is, we can define $g : \{0,1\}^n \to \{0,1\}^\ell$ such that $\Pr[g(x) = z] = \Pr[Z = z | X = x]$, then $(X, Z) = (X, g(X))$. Essentially, the goal is to find an "small circuit" h to simulate g.

First, we recall the multiplicative weight algorithm and the main theorem of it.

Multiplicative weight update

One canonical usage of the multiplicative weight update is in the following prediction game. The game consist of T rounds of predicting. There are L experts who also make there prediction that we can refer to in each round. Our goal is to minimize the difference between the error to the best expert and ours, which usually called *regret* in literatures. That is, we want to minimize

$$\sum_{i=1}^{T} \mathop{\mathbb{E}}_{j \xleftarrow{r} D^{(i)}} \left[f^{(i)}(j) \right] - \max_{j \in [L]} \sum_{i=1}^{T} f^{(i)}(j)$$

where $f^{(i)}(j)$ is the error of the j-th expert's strategy in the i-th round, and $D^{(i)}$, a distribution on $[L]$ is our strategy in the i-th round.

The multiplicative weight update algorithm stated in Algorithm 4.2.2 provides a good way to minimize the regret. The regret of the multiplicative weight update algorithm is guaranteed by following theorem.

Theorem 4.2.3 (e.g., [AHK12])**.** *For given error functions $f^{(i)} : [L] \to [0,1]$ for all $i \in [T]$, let $D^{(i)}$ be distributions defined in Algorithm 4.2.2 with the parameter $\eta \in (0, 1/2)$. Then for every $j^* \in [L]$, we have*

$$\sum_{i=1}^{T} \mathop{\mathbb{E}}_{j \xleftarrow{r} D^{(i)}} \left[f^{(i)}(j) \right] - \sum_{i=1}^{T} f^{(i)}(j^*) \ \le \frac{\log L}{\eta} + T\eta \, .$$

```
Algorithm 4.2.2: Multiplicative weight update
Parameter: $\eta \in (0, 1/2)$
Set $w_j^{(0)} = 1$ for all $j \in [L]$.
For $i = 1 \to T$
    1. $\forall j \in [L]$, $w_j^{(i)} = w_j^{(i-1)} \cdot (1-\eta)^{f^{(i)}(j)}$.
    2. Let $D^{(i)}$ be such that $\Pr[D^{(i)} = j] \propto w_j^{(i)}$.
```

In particular, if we set $\eta = \sqrt{\log L/T}$, we have

$$\sum_{i=1}^{T} \mathop{\mathbb{E}}_{j \overset{r}{\leftarrow} D^{(i)}} \left[f^{(i)}(j) \right] - \sum_{i=1}^{T} f^{(i)}(j^*) \ \le O\big(\sqrt{T \log L}\big)\,.$$

An important phenomena of Theorem 4.2.3 is that the regret grows sub-linear to T. Therefore, the predictor can achieve arbitrarily small "average" regret for large enough T.

Corollary 4.2.4. *For the same setting as in Theorem 4.2.3, there exists $T = O\big(\log L/\varepsilon^2\big)$ such that for all $j^* \in [L]$,*

$$\sum_{i=1}^{T} \mathop{\mathbb{E}}_{j \overset{r}{\leftarrow} D^{(i)}} [f^{(i)}(j)] - \sum_{i=1}^{T} f^{(i)}(j^*) \ \le \varepsilon\,.$$

Now we show how to construct a simulator for the Leakage Simulation Lemma via MWU algorithm. The first step is to remove the one-sided error constraint. Let $\mathcal{F}'$ denote the closure of $\mathcal{F}$ under complement, namely, $f' = \{f, 1-f : f \in \mathcal{F}\}$. Then the indistinguishability constraint is equivalent to

$$\forall f \in \mathcal{F}'\ , \quad \mathop{\mathbb{E}}_{\mathsf{h}, x \overset{r}{\leftarrow} X} [f(x, \mathsf{h}(x))] - \mathop{\mathbb{E}}_{g, x \overset{r}{\leftarrow} X} [f(x, g(x))] \le \varepsilon.$$

One can see that the simulating auxiliary input has a similar structure as the task that MWU tries to solve. The output of the simulator h is the strategy for prediction, and g is the error criteria. The challenging part is there exists an input x, and our strategy should handle different x's. Tackling them separately would be trivial but the circuit complexity will be too high.

While the framework Vadhan and Zheng's considered is more general, the proof is also more complicated. Below we give a simpler proof which only uses the no-regret property of

MWU.[4] Note that any no-regret algorithms for making prediction based on experts works for this proof. Indeed, by applying online gradient descent instead of MWU we will get an additive boosting simulator. Nevertheless, multiplicative weight update is optimal in expert learning, which explains why MWU converges faster than additive boosting proofs.

```
Algorithm 4.2.3: Simulator $\mathsf{h}$
Input: $x \in \{0,1\}^n$
Parameter: $\varepsilon > 0$
  1. Let $T = O(\ell/\varepsilon^2)$ and $\eta = \sqrt{\log L/T}$.
  2. $\forall z \in \{0,1\}^\ell, w_z^{(0)}(x) = 1$.
  3. Let $\mathsf{h}^{(0)}$ be a randomized function such that $\Pr[\mathsf{h}^{(0)}(x) = z] \propto w_z^{(0)}(x)$.
  4. For $i = 1 \to T$
     (a) Define $\mathsf{adv}^{(i)}(f) \overset{\text{def}}{=} \mathop{\mathbb{E}}_{\mathsf{h}^{(i-1)}}[f(X, \mathsf{h}^{(i-1)}(X))] - \mathop{\mathbb{E}}_{g}[f(X, g(X))]$.
     (b) Let $f_{\mathsf{max}}^{(i)} = \arg\max_{f \in \mathcal{F}'} \mathsf{adv}^{(i)}(f)$.
     (c) If $\mathsf{adv}^{(i)}(f_{\mathsf{max}}^{(i)}) \le \varepsilon$        Return $\mathsf{h}^{(i-1)}(x)$
     (d) $\forall z \in \{0,1\}^\ell$, set $w_z^{(i)}(x) = w_z^{(i-1)}(x) \cdot (1-\eta)^{f_{\mathsf{max}}^{(i)}(x,z)}$
     (e) Let $\mathsf{h}^{(i)}$ be a randomized function such that $\Pr[\mathsf{h}^{(i)}(x) = z] \propto w_z^{(i)}(x)$.
  5. Return $\mathsf{h}^{(T)}(x)$
```

Note that since $f_{\mathsf{max}}^{(i)}$ depends on $\mathsf{h}^{(i-1)}(x)$ for all $x \in \{0,1\}^n$, we have to run the simulator h "in parallel" to properly define $f_{\mathsf{max}}^{(i)}$. However, $f_{\mathsf{max}}^{(i)}$ does not depend on any particular x. That is, no matter which $x \in \{0,1\}^n$ is the input of h, $f_{\mathsf{max}}^{(i)}$'s are the same. This is crucial since we will hard code $f_{\mathsf{max}}^{(i)}$'s as advice to the simulator h. It would be too long if it does depend on x.

Lemma 4.2.5. *Let X be a distribution over $\{0,1\}^n$ and $g : \{0,1\}^n \to \{0,1\}^\ell$ be a randomized function. For a given error parameter ε, the function h defined by Algorithm 4.2.3 satisfies*

$$\forall f \in \mathcal{F}' \,, \quad \mathop{\mathbb{E}}_{\mathsf{h}, x \xleftarrow{r} X}\Big[f(x, \mathsf{h}(x))\Big] - \mathop{\mathbb{E}}_{g, x \xleftarrow{r} X}\Big[f(x, g(x))\Big] \le \varepsilon \,.$$

[4] We say an online decision-making algorithm is *no-regret* if the average regret tends to zero as T approaches infinity. See, e.g., [RW16].

Proof. If there exists i such that $\mathsf{adv}(f^{(i)}_{\mathsf{max}}) \leq \varepsilon$, let i^* be the smallest one, then $\mathsf{h} = \mathsf{h}^{(i^*)}$ satisfies the requirement.

Otherwise, $\forall i \in [T]$, $\mathsf{adv}(f^{(i)}_{\mathsf{max}}) > \varepsilon$. By Corollary 4.2.4, for every $x \in \{0,1\}^n$ and $z \in \{0,1\}^\ell$,

$$\frac{1}{T}\sum_{i=1}^{T} \mathop{\mathbb{E}}_{\mathsf{h}^{(i)}(x)}\left[f^{(i)}(x, \mathsf{h}^{(i)}(x))\right] - \frac{1}{T}\sum_{i=1}^{T} f^{(i)}(x,z) \leq \varepsilon.$$

Taking (x,z) over the distribution $(X, g(X))$, we have

$$\frac{1}{T}\sum_{i=1}^{T}\mathsf{adv}(f^{(i)}_{\mathsf{max}}) = \frac{1}{T}\sum_{i=1}^{T} \mathop{\mathbb{E}}_{x \overset{\mathrm{r}}{\leftarrow} X, \mathsf{h}^{(i)}(x)}\left[f^{(i)}(x, \mathsf{h}^{(i)}(x))\right] - \frac{1}{T}\sum_{i=1}^{T} \mathop{\mathbb{E}}_{x \overset{\mathrm{r}}{\leftarrow} X, g(x)}\left[f^{(i)}(x, g(x))\right] \leq \varepsilon,$$

which contradict the assumption. □

4.2.2 Efficient approximation

Algorithm 4.2.3 provides a simulator which fools all distinguishers in $\mathcal{F}$ by error up to ε. However, we have only proved a bound for the number of iterations, but not for the complexity of h itself. In fact, the circuit complexity of a naive implementation of Algorithm 4.2.3 is not better than using additive boosting due to the precision issue. We provide another way to approximates h, which has complexity not much worse than evaluating the distinguishers T times.

Define $s(x,z) = \sum_{i=1}^{T} f^{(i)}(x,z)$. Then $\mathsf{h}(x)$ effectively outputs z with probability proportional to $(1-\eta)^{s(x,z)}$, Note that T and $f^{(1)}, \ldots, f^{(T)}$ can be provided by advice string and here we define T as the number of rounds that h actually runs since it may terminate early.

A natural way to approximate h is to compute $(1-\eta)^{s(x,z)}$ for each z and output the result randomly using rejection sampling. It takes $O(\log(1/\eta))$ bits to represent $(1-\eta)$, and thus it takes at most $O(k\log(1/\eta))$ to represent $(1-\eta)^k$ for $k \in \mathbb{N}$. Also, because $s(x,z)$ is at most T, it takes $O(Tt + T^2\log^2(1/\eta))$ complexity to compute $(1-\eta)^{s(x,z)}$ by naive multiplication, or $O(Tt + T^2\log T\log(1/\eta))$ via lookup table. Therefore, there exists an approximation of h_T of size $O((T^2\log^2(1/\eta) + Tt)\cdot 2^\ell)$, which is $\widetilde{O}(t\cdot 2^\ell\varepsilon^{-2} + 2^\ell\varepsilon^{-4}))$ after expanding T and η. This is the complexity claimed in [VZ13a]. As mentioned in [Sko16a], this bound is incomparable to the ones in [JP14, Sko16a, Sko16b], and the $\widetilde{O}(2^\ell\varepsilon^{-4})$ term may dominate in some settings.

Now we state the idea of approximating normalized weights efficiently. Observe that weights are of the form $(1-\eta)^{s(x,z)}$. If the total weight is guaranteed to be at least 1, then intuitively, truncating the weight at each $z \in \{0,1\}^{\ell}$ a little amount does not influence the result distribution too much. Hopefully, if the truncated values can be stored with a small number of bits, a lookup table which maps $s(x,z)$ to the truncated value of $(1-\eta)^{s(x,z)}$ is affordable.

By the above idea, we define $\widehat{\mathsf{h}}$ to approximate h as follows:

Algorithm 4.2.4: Simulator $\widehat{\mathsf{h}}$

Input: $x \in \{0,1\}^n$

Parameter: $\varepsilon > 0$

1. $\forall z \in \{0,1\}^{\ell}$, compute $s'(x,z) = s(x,z) - \min_{z'} s(x,z')$.
2. Let k be the smallest integer such that $2^{-k} \leq \frac{\eta}{2^{\ell}(1+\eta)}$. Then let $\widehat{w}_z(x)$ be $(1-\eta)^{s(x,z)}$ truncated down to the closest multiple of 2^{-k}.
3. Build a lookup table consists of truncated value of $(1-\eta)^j$ for $j \in [\tau]$ where τ is large enough such that it contains all $\widehat{w}_z(x)$.
4. Output $\widehat{\mathsf{h}}(x)$ such that $\Pr\left[\widehat{\mathsf{h}}(x) = j\right] \propto \widehat{w}_j(x)$ by rejection sampling up to η accuracy (in terms of statistical distance).

First we show that $\widehat{\mathsf{h}}$ can be implemented by a circuit of size $\widetilde{O}(2^{\ell}\varepsilon^{-2})$. Recall that $T = O(\ell/\varepsilon^2)$ and $\eta = O(\varepsilon)$. For the step 1, since $s(x,z) \in \{0\} \cup [T]$, $F'(x,z)$ for all $z \in [2^{\ell}]$ can be calculated by a circuit of size $O(2^{\ell} \cdot (tT + T\log T)) = t \cdot \widetilde{O}(2^{\ell}2^{\varepsilon})$. For step 2 and 3, we have that $k = O(\ell \log(1/\eta))$ and $\tau = O(k/\eta)$ and storing one truncated value takes $k \log \tau$ bits. Therefore, the size of the table is $O(k\tau \log \tau) = \widetilde{O}(\varepsilon^{-1})$. In the last step, to achieve the η accuracy, each rejection sampling step takes $O(\ell \log(1/\eta))$ time, so the total time complexity for this step is $\widetilde{O}(2^{\ell})$.

Now we want to show that $\widehat{\mathsf{h}}$ is indeed a good approximation of h.

Claim 4.2.6. *For every x, $\mathrm{d}_{\mathsf{TV}}(\widehat{\mathsf{h}}(x)\,,\,\mathsf{h}(x)) \leq 2\eta$.*

Proof. Since $\widehat{\mathsf{h}}(x)$ is η-close to the distribution defined by $\{\widehat{w}_z(x)\}_{z\in[2^\ell]}$, it suffices to show that $\{\widehat{w}_z(x)\}_{z\in[2^\ell]}$ is η-close to $\mathsf{h}(x)$, which is the distribution defined by $\{w_z(x)\}_{z\in[2^\ell]}$.

First we have the following subclaim.

Subclaim 4.2.7. *Suppose there are two sequences of positive real numbers $\{\gamma_z\}_{z\in[L]}$, $\{w_z\}_{z\in[L]}$ such that $\forall z \in [L]$, $\gamma_z \le w_z$. Let $r = \sum_z \gamma_z / \sum_z w_z$ and Z, Z' be a distribution on $[L]$ such that $\Pr[Z = z] \propto w_z$ and $\Pr[Z' = z] \propto (w_z - \gamma_z)$, respectively. Then $\mathrm{d}_{\mathsf{TV}}(Z\ , Z') \le \frac{r}{1-r}$.*

Proof. By the definition of statistical distance (total variance),

$$
\begin{aligned}
\mathrm{d}_{\mathsf{TV}}(Z\ , Z') &= \frac{1}{2}\sum_z \left|\frac{w_z}{\sum_{z'} w_{z'}} - \frac{w_z - \gamma_z}{\sum_{z'}(w_{z'} - \gamma_{z'})}\right| \\
&= \frac{1}{2}\sum_z \left|\frac{\gamma_z \sum_{z'} w_{z'} - w_z \sum_{z'} \gamma_{z'}}{(\sum_{z'} w_{z'})^2(1-r)}\right| \\
&\le \frac{1}{2}\sum_z \frac{w_z \sum_{z'} \gamma_{z'} + \gamma_z \sum_{z'} w_{z'}}{(\sum_i w_{z'})^2(1-r)} \\
&= \frac{\sum_{z'} w_{z'} \sum_{z'} \gamma_{z'}}{(\sum_{z'} w_{z'})^2(1-r)} = \frac{r}{1-r}
\end{aligned}
$$

□

By the definition of $\widehat{w}_z(x)$, $\widehat{w}_z(x) = w_z(x) - \gamma_z(x)$ where

$$
\gamma_z(x) \le \min\Big\{(1-\eta)^{s(x,z)}\ ,\ \frac{\eta}{2^\ell(1+\eta)} \cdot \sum_{z'} (1-\eta)^{s(x,z)}\Big\}\,.
$$

Clearly, $(\sum_z \gamma_z)/(\sum_z w_z) \le 2^\ell \cdot \frac{\eta}{2^\ell(1+\eta)} \le \eta$. Apply Subclaim 4.2.7, we conclude the claim. □

By Claim 4.2.6, we have

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{\widehat{\mathsf{h}},x\leftarrow X}\Big[f(x,\widehat{\mathsf{h}}(x))\Big] - \mathop{\mathbb{E}}_{g,x\leftarrow X}\Big[f(x,g(x))\Big] \\
\le &\mathop{\mathbb{E}}_{\mathsf{h},x\leftarrow X}\Big[f(x,\mathsf{h}(x))\Big] - \mathop{\mathbb{E}}_{g,x\leftarrow X}\Big[f(x,g(x))\Big] + 2\eta \\
\le &\,\varepsilon + 2\eta = O(\varepsilon)\,,
\end{aligned}
$$

which conclude Theorem 4.2.1.

4.3 Lower Bound for Leakage Simulation

We have seen that there exists an MWU algorithm which combines only $O(\ell\varepsilon^{-2})$ distinguishers to make a good simulator h. Besides, for every chosen distinguisher f the algorithm queries $f(x,z)$ for every $z \in \{0,1\}^\ell$ when computing $\mathsf{h}(x)$. Therefore the algorithm makes $O(\ell \cdot 2^\ell \varepsilon^{-2})$ queries in total. In the previous section, we also showed that evaluating the $O(\ell\varepsilon^{-2})$ chosen distinguishers is the bottleneck of the simulation. Then a natural question arises: can we construct a simulator which makes fewer queries? It might be possible to find a boosting procedure using fewer distinguishers, or maybe we can skip some $z \in \{0,1\}^\ell$ when querying $f(x,z)$ for some f. However, in this section we will show that the MWU approach is almost optimal: any *black-box* simulator which satisfies an "independence restriction" has to make $\Omega(2^\ell\varepsilon^{-2})$ queries to fool the distinguishers.

4.3.1 Black-box model

To show the optimality of the MWU approach, we consider black-box simulation, which means simulators only use the distinguishers as black-boxes. Note that all known results of leakage simulation ([VZ13a, JP14, Sko16a, Sko16b]) are black-box. Indeed, all the leakage simulation results are in the following form: first find a set of distinguishers $\{f_1,\ldots,f_q\}$ which does not depend on input x, then query $f_i(x,z)$ for every $z \in \{0,1\}^\ell$ and $i \in [q]$. Finally combine the results to obtain the distribution of $\mathsf{h}(x)$. Formally, we define a black-box simulator as follows.

Definition 4.3.1 (black-box simulator). *Let $\ell, m, a \in \mathbb{N}$ and $\varepsilon > 0$. A black-box construction of a simulator $\mathsf{h}^{(\cdot)} : \{0,1\}^n \times \{0,1\}^a \to \{0,1\}^\ell$ takes two inputs: $x \in \{0,1\}^n$ and an advice string $\alpha \in \{0,1\}^a$. $\mathsf{h}^{(\cdot)} : \{0,1\}^n \times \{0,1\}^a \to \{0,1\}^\ell$ is a ε-black-box simulator for $\mathcal{F}$ if for every joint distribution (X,Z) on $\{0,1\}^n \times \{0,1\}^\ell$, there exists $\alpha \in \{0,1\}^a$ such that*

$$\forall f \in \mathcal{F}\ ,\ \Big|\mathbb{E}[f(X,Z)] - \mathbb{E}[f(X,\mathsf{h}(X))]\Big| < \varepsilon\,.$$

We call a black-box simulator same-input *if for, for every $f \in \mathcal{F}, \alpha \in \{0,1\}^a$, $\mathsf{h}(\cdot;\alpha)$ only queries $f(x,\cdot)$.*

The lower bound we prove in this paper is for *same-input black-box simulator*. The same-input assumption is also made in related works including [LTW11, Zha11, PS16]. See the next section for more discussions about the black-box models in related results.

It is not hard to see that all the boosting approaches we mentioned above are in this model: the advice α is of length $O(q \log |\mathcal{F}|)$ and is used for indicating "which distinguishers should be queried", and h queries every chosen distinguisher f with input (x, z) for every $z \in \{0,1\}^\ell$ when computing $\mathsf{h}^{\mathcal{F}}(x; \alpha)$. Moreover, these simulation algorithms are non-adaptive. We can write the MWU approach as the following corollary:

Corollary 4.3.2. *For every $0 < \varepsilon < \frac{1}{2}$, $\ell, m \in \mathbb{N}$, there exists a non-adaptive same-input ε-black-box $\mathsf{h} : \{0,1\}^n \times \{0,1\}^a \to \{0,1\}^\ell$ for $\mathcal{F}$ with query complexity $q = O(\ell 2^\ell \varepsilon^{-2})$ and $a = \widetilde{O}\big(q \log|\mathcal{F}|\big)$.*

Besides capturing all known simulators, our lower bound also rules out the adaptive approaches. Whether there exists a faster simulation not satisfying the same-input restriction is left open, but intuitively, querying oracles on different inputs does not seem useful.

4.3.2 Main theorem and related results

Theorem 4.3.3. *Let n be the security parameter. For every $2^{-o(n)} < \varepsilon < 0.001$, $\ell = o(n)$, and $a = 2^{o(n)}$, there exists a family of distinguisher $\mathcal{F}$ of $\omega(2^\ell/\varepsilon^3) < |\mathcal{F}| < 2^{2^{o(n)}}$ such that a same-input ε-blakc-box simulator $\mathsf{h} : \{0,1\}^n \times \{0,1\}^a \to \{0,1\}^\ell$ for $\mathcal{F}$ must have query complexity $q = \Omega(2^\ell \varepsilon^{-2})$.*

Note that the lower bound of the size of $\mathcal{F}$ is needed so the simulator must "simulate" the function instead of fooling distinguishers one by one.

Before this paper, there were some lower bounds for Leakage Simulation Lemma itself or its implications. We discuss some of the results based on their models:

- **Non-adaptive same-Input black-box lower bounds.** Recall that Leakage Simulation Lemma implies Impagliazzo's Hardcore Lemma and Dense Model Theorem. Lu, Tsai and Wu [LTW11] proved an $\Omega(\log(1/\delta)/\varepsilon^2)$ query complexity lower bound for Impagliazzo's Hardcore Lemma where δ is the density of the hardcore set. Taking $\delta = \Theta(1)$ yields an $\Omega(1/\varepsilon^2)$ query complexity lower bound for Leakage Simulation. Similarly, Zhang [Zha11] proved a lower bound for query complexity in Dense Model Theorem proof which implies the $\Omega(1/\varepsilon^2)$ lower bound as well. The black-box models considered in both works have some restrictions. In fact, the black-box model in [LTW11] does not contain Holenstein's proof [Hol05]. Nevertheless, after converting to the lower bound for Leakage Simulation Lemma, their models fit into our setting. Also, Pietrzak and Skórski [PS16] proved a $\Omega(2^\ell/\varepsilon)$ lower bound for Leakage Chain Rule for relaxed HILL entropy, which also implies a $\Omega(2^\ell/\varepsilon)$ lower bound for Leakage Simulation Lemma. These lower bounds assume both the non-adaptivity and the independence of inputs. Interestingly, in the reduction from Leakage Chain Rule to Leakage Simulation, there exists a distinguisher in the reduction which only need to be queried on one adaptively chosen input. In this case the non-adaptivity causes a 2^ℓ additive loss. This can be viewed as an evidence that adaptivity might be useful.

- **Non-Adaptive Black-Box Lower Bounds.** Impagliazzo [Imp95] proved that the Hardcore Lemma implies Yao's XOR Lemma [Yao82, GNW11], which is an important example of hardness amplification. Since the reduction is black-box, it is not hard to see that the $\Omega(\log(\frac{1}{\delta})/\varepsilon^2)$ lower bound for hardness amplification proved by Shaltiel and Viola [SV10] is also applicable to Hardcore Lemma. Again, by setting $\delta = \Theta(1)$ we get a $\Omega(1/\varepsilon^2)$ lower bound for Leakage Simulation. Their model is incomparable to ours as they do not require the "same-input" assumption, but require the non-adaptivity.

- **General Black-Box Lower Bounds.** Artemenko and Shaltiel [AS14] proved an $\Omega(1/\varepsilon)$ lower bound for a simpler type of hardness amplification, and removed the non-adaptivity. Their result implies a general black-box lower bound for Leakage Simulation Lemma, but with less optimal parameters.

- **Non-Black-Box Lower Bounds.** Trevisan, Tulsiani and Vadhan show that the simulator cannot be much more efficient than the distinguishers [TTV09, Remark 1.6]. Indeed, for any large enough $t \in \mathbb{N}$ they construct a function g such that any simulator h of complexity t can be distinguished from g by a distinguisher of size $\widetilde{O}(nt)$. Jetchev and Pietrzak [JP14] also show an $\Omega(2^\ell \cdot t)$ lower bound under some hardness assumptions for one-way functions.

None of the existing results imply an optimal lower bound for Leakage Simulation. However, proving a lower bound for Leakage Simulation Lemma might be a simpler task, and it turns out that we can prove a lower bound of $\Omega(2^\ell \varepsilon^{-2})$. The ideas for proving the lower bound are as follows. To capture the 2^ℓ factor, for each distinguisher f and input x we hide information at $f(x,z)$ for a random z, similar to the idea in [PS16]. Then checking all z over $\{0,1\}^\ell$ is necessary. Although the claim is quite intuitive, a formal analysis is more involved in an adaptive model. The ε^{-2} factor is from the anti-concentration bound — it takes $\Omega(1/\varepsilon^2)$ samples to distinguish Bernoulli distributions $\mathrm{Bern}(1/2+\varepsilon)$ and $\mathrm{Bern}(1/2)$ with constant probability. The concept is also used in the some related lower bound, e.g., [Fre95, LTW11, PS16]. Note that in [PS16] they only require an advantage of ε when distinguishing such Bernoulli distribution from uniform, which causes an $O(1/\varepsilon)$ loss in complexity.

4.3.3 Proof of the lower bound

Overview. We would like to show the existence of function g and a set of distinguisher $\mathcal{F}$ such that any simulator h with limited queries to $\mathcal{F}$ cannot approximate g well by probabilistic method. More specifically, we first consider a randomly-chosen function g randomized distinguishers and show that with high probability, a black-box simulator with fixed advice cannot simulate well. Then show the existence of a fixed g and derandomize the distinguisher by union bound over all possible advice string of the black-box simulator.

Let G be the uniform distribution of a function $g : \{0,1\}^n \to \{0,1\}^\ell$. Given g, let $f_g(x,z)$ be a random bit drawn from $\mathrm{Bern}(1/2+c_1\varepsilon)$ for some constant c_1 if $z = g(x)$, or from $\mathrm{Bern}(\frac{1}{2})$ otherwise. In other word, $f_g(x,z)$ "leak" some information about g if and only if the "correct" z is provided. Providing g and distinguishers in this way, f provides very little information about g, yet it can distinguish g from trivial functions (the ones do not know what $g(x)$ is) with advantage ε. Intuitively, since $f(x, g(x))$ is only $\Theta(\varepsilon)$ away from uniform, f can distinguish g and any bad simulator h which does not approximate g with constant probability. To approximate g well, we need to test $O(2^\ell)$ keys to find the correct one. Besides, it requires $\Omega(1/\varepsilon^2)$ samples to distinguish Bern$(1/2 + \Theta(\varepsilon))$ and Bern$(1/2)$ with constant probability, so $\Omega(1/\varepsilon^2)$ queries are required for each key to make sure we can distinguish the real key from other fake keys. Therefore a successful simulator h should make at least $\Omega(\varepsilon^{-2}2^\ell)$ queries.

Now we proceed to the formal proof. Assume for contradiction that $\mathsf{h} : \{0,1\}^n \to \{0,1\}^\ell$ is a ε-black-box simulator with query complexity $q \le c_2 \cdot 2^\ell \varepsilon^{-2}$, where $c_2 = \frac{1}{360000}$. Let $g : \{0,1\}^n \to \{0,1\}^\ell$ be a function randomly chosen from G. Thus for every $x \in \{0,1\}^n$, $g(x)$ is chosen uniformly at random from $\{0,1\}^\ell$. Let $\mathcal{F}$ be a set of random function defined above with $c_1 = 30$. First, we prove that given any fixed advice string α, the $\mathsf{h}^{\mathcal{F}}(,\alpha)$ cannot guess g correctly with high enough probability over the choice of g and the randomness of $f \in \mathcal{F}$.

Lemma 4.3.4. *For every $x \in \{0,1\}^n$ and $\alpha \in \{0,1\}^a$, we have*

$$\Pr_{g \xleftarrow{r} G}\Big[\mathsf{h}^{\mathcal{F}}(x;\alpha) = g(x)\Big] \le 1 - \frac{3}{c_1},$$

where the probability is taken over the choice of $g(x)$, $f(x,\cdot)$ for every $f \in \mathcal{F}$ (abbreviated as $\mathcal{F}(x)$), and the randomness of h.

Proof. Without loss of generality, assume that h has no randomness other than oracle queries (We can obtain the same bound for probabilistic h by taking average over deterministic circuits.), and h always make q different queries. Now h is fully decided by the q query answered it makes to the oracle denoted as $b = (b_1, \ldots, b_q) \in \{0,1\}^q$ where b_i is the answer to the i-th query. Let B be a random variable for randomized b (due to the random distinguishers $f \in \mathcal{F}$). Here we use $h' : \{0,1\}^q \to \{0,1\}^\ell$ to denote the function maps query results to the output of $\mathsf{h}^{\mathcal{F}}(x;\alpha)$. Then we have

$$\Pr_{g \xleftarrow{r} G, \mathcal{F}}\Big[\mathsf{h}^{\mathcal{F}}(x;\alpha) = g(x)\Big] = \Pr_{B, g \xleftarrow{r} G}\Big[h'(B) = g(x)\Big] = \sum_{b,k} \Pr_{g \xleftarrow{r} G}\Big[B = b, g(x) = k, h'(b) = k\Big].$$

Use $\mathcal{F}^*$ to denote the set of distinguishers with uniform random function (with no bias) and let

B^* be the corresponding "ideal" transcript (basically U_q). For every $(b,k) \in \{0,1\}^q \times \{0,1\}^\ell$, $\Pr_{B^*, g \xleftarrow{r} G}[B^* = b, g(x) = k] = 2^{-(q+\ell)}$. Since for b uniquely determines $h'(b)$, only 2^q pairs (b,k) are *correct*. In the ideal case, we have $\Pr_{\mathcal{F}^*, g \xleftarrow{r} G}[\mathsf{h}^{\mathcal{F}^*}(x;\alpha) = g(x)] = 2^{-\ell}$ where h^* denotes the ideal variant of h. while in the real case, $\Pr_{B, g \xleftarrow{r} G}[B = b, g(x) = k]$ can be as large as $2^{-\ell}(\frac{1}{2} + c_1\varepsilon)^q$ (when h queries with correct key in every query and all the responses are 1). However, for most b that is not the case. Also, most of the pairs $(b,k) \in \{0,1\}^q \times \{0,1\}^\ell$ do not satisfy $h'(b) = k$. Therefore we can expect that a large fraction of pairs are chosen with probability $\Theta(2^{-(q+\ell)})$ and $h'(b) \neq k$. The above statement provides an intuition of the lower bound for $\Pr_{\mathcal{F}, g \xleftarrow{r} G}[\mathsf{h}^{\mathcal{F}}(x;\alpha) \neq g(x)]$.

Next we formally prove the above statement. Consider any $b = \{b_1, b_2, \ldots, b_q\}$. Recall that the queries made by h are uniquely determined by b: the first query is fixed, the second query is determined by the first bit of b, and so on. Let $(z_1, z_2, \ldots, z_q)$ be the sequence of key such that the i-th query is $f^{(i)}(x, z_i)$ for some $f^{(i)} \in \mathcal{F}$. For any $k \in \{0,1\}^\ell, b \in \{0,1\}^q$, let u_i denote the index of the i-th "useful query", which means the i-th index satisfying $z_{u_i} = k$. Then we define $N_\beta(b,k) \stackrel{\text{def}}{=} \sum_i [a_{u_i} = \beta]$ for $\beta \in \{0,1\}$, which represents the number of useful queries with response β. For convenience, we also define

$$N(b,k) \stackrel{\text{def}}{=} N_0(b,k) + N_1(b,k) \text{ and } N_\Delta(b,k) \stackrel{\text{def}}{=} N_0(b,k) - N_1(b,k)\,.$$

We further define more refined notation: for $j \leq N(b,k)$, we define $N_\beta(b,k,j) \stackrel{\text{def}}{=} \sum_{i=1}^{j} [a_{u_i} = \beta]$ for $\beta \in \{0,1\}$ and $N_\Delta(b,k,j) \stackrel{\text{def}}{=} N_0(b,k,j) - N_1(b,k,j)$. That is, only first j useful queries are considered. Recall that for $f_g \in \mathcal{F}$, $f_g(x,z)$ is uniform when $z \neq g(x)$ and biased when $z = g(x)$. For every fixed (b,k),

$$\begin{aligned}
\Pr_{B, g \xleftarrow{r} G}[g(x) = k, B = b] &= \left(\frac{1}{2}\right)^{(\ell+q-N(b,k))} \left(\frac{1}{2} - c_1\varepsilon\right)^{N_0(b,k)} \left(\frac{1}{2} + c_1\varepsilon\right)^{N_1(b,k)} \\
&= \left(\frac{1}{2}\right)^{(\ell+q)} (1 - 2c_1\varepsilon)^{N_\Delta(b,k)} \left(1 - 4c_1^2\varepsilon^2\right)^{N_1(b,k)} \\
&\geq \left(\frac{1}{2}\right)^{(\ell+q)} (1 - 2c_1\varepsilon)^{N_\Delta(b,k)} \left(1 - 4c_1^2\varepsilon^2\right)^{N(b,k)} \qquad (4.1)
\end{aligned}$$

We say a pair (b,k) is normal if $N_\Delta(b,k) = O(1/\varepsilon)$ and $N(b,k) = O(1/\varepsilon^2)$. We claim that a large enough fraction of pairs over $\{0,1\}^q \times \{0,1\}^\ell$ are normal and $h'(b) \neq k$:

Claim 4.3.5. *Let $q' = 5q/2^\ell \leq 5c_2\varepsilon^{-2}$. Then for at least $1/5$ fraction of pairs (b,k) over $\{0,1\}^q \times \{0,1\}^\ell$ satisfy (1) $h'(b) \neq k$, (2) $N(b,k) < q'$, and (3) $N_\Delta(b,k) < \sqrt{5q'}$.*

Proof of Claim. We consider each condition one by one.

(1) Only $2^{-\ell}$ of pairs satisfies $h'(b) = k$: This is obvious since h' is a deterministic function.

(2) At most $\frac{1}{5}$ of pairs (b,k) satisfy $N(b,k) \geq q'$: For any b we have $\mathbb{E}_{k\leftarrow U_\ell}[N(b,k)] = \frac{q}{2^\ell}$. By Markov's inequality, at most $\frac{q}{2^\ell q'} = \frac{1}{5}$ of pairs satisfy $N(b,k) \geq q'$.

(3) For at most $\frac{1}{10}$ of pairs (b,k), $N(b,k) < q'$ and $N_\Delta(b,k) > \sqrt{5q'}$:

Let B^* be a random transcript which is uniform over $\{0,1\}^q$. For a fixed k, consider a sequence of random variable $\{Y_j\}$ depending on B^* such that

$$Y_j = \begin{cases} N_\Delta(B^*,k,j) & \text{if } j < N(B^*,k) \\ N_\Delta(B^*,k) & \text{otherwise.} \end{cases}$$

It's not hard to see that $\{Y_i\}$ is a martingale with difference at most 1. By Azuma's inequality, we have $\Pr[Y_{q'} \geq \sqrt{5q'}] \leq e^{-5q'/2q'} < 0.1$. Since B^* is uniform, the statement above is the same as saying for at most 0.1 fraction of $t \in \{0,1\}^q$, $Y_{q'}(b) \geq \sqrt{5q'}$. Restricting b to satisfy $N(b,k) < q'$ we have $N_\Delta(b,k) = Y_{q'}(b) \geq \sqrt{5q'}$.

By union bound, all three conditions in the claim hold simultaneously for at least $\frac{1}{5}$ of pairs over $\{0,1\}^q \times \{0,1\}^\ell$. □

Now consider any pair (b,k) which satisfies condition (2) and (3) in the claim above, in other word a *normal* pair. By inequality (4.1), we have

$$\begin{aligned} \Pr[g(x) = k, B = b] &\geq (1/2)^{\ell+q}\,(1-2c_1\varepsilon)^{N_\Delta(b,k)}(1-4c_1^2\varepsilon^2)^{N(b,k)} \\ &\geq (1/2)^{\ell+q}\,(1-2c_1\varepsilon)^{\sqrt{5q'}}(1-4c_1^2\varepsilon^2)^{q'} \qquad (4.2) \\ &= (1/2)^{(\ell+q)}\,(1-2c_1\varepsilon)^{5\sqrt{c_2}\varepsilon^{-1}}(1-4c_1^2\varepsilon^2)^{5c_2\varepsilon^{-2}} \end{aligned}$$

$$\geq (1/2)^{\ell+q}\,(0.3)^{10c_1\sqrt{c_2}}(0.3)^{20c_1^2c_2} \tag{4.3}$$

$$\geq (1/2)^{\ell+q}\cdot 0.5\,. \tag{4.4}$$

The inequality (4.3) holds because $(1-\delta)^{1/\delta} \geq 0.3$ for any $0 < \delta \leq 0.1$. Since $\frac{1}{5}$ of pairs satisfy the conditions above, we have

$$\Pr[\mathsf{h}(x) \neq g(x)] = \sum_{k,b} g(x) = k, B = b, h'(b) \neq k \geq 0.1\,.$$

Therefore $\Pr[\mathsf{h}(x) = g(x)] \leq 0.9 = 1 - \frac{3}{c_1}$. □

With the lemma above, we can finish the proof by a concentration bound and probabilistic method. We extend the distinguisher class $\mathcal{F}$ such that each randomized distinguisher $f_g \in \mathcal{F}$ is replaced by many deterministic distinguisher $\{f_{g,r}\}_r$ with different random coins as input. Fix an advice α. For every $x \in \{0,1\}^n$, $f_g \in \mathcal{F}$ such that $f_{g,r}$ is not queried by $\mathsf{h}(x)$, we have $\mathbb{E}_{g \xleftarrow{r} G,r}[f_{g,r}(x,\mathsf{h}(x))] = \frac{1}{2} + \Pr_{g \xleftarrow{r} G}[\mathsf{h}^{\mathcal{F}}(x;\alpha) = g(x)] \cdot c_1\varepsilon$ by definition of f. Since h makes at most q query when computing $\mathsf{h}(x)$, when $f_{g,r}$ is sampled randomly, it chooses a query coincident with queries in h with probability $q/|\mathcal{F}|$. Even in the worst case that $f_{g,r}$ returns 1 in all these cases, we still have

$$\mathop{\mathbb{E}}_{g \xleftarrow{r} G,r}[f_{g,r}(x,\mathsf{h}(x;\alpha))] \leq \frac{1}{2} + \Pr_{g \xleftarrow{r} G}[g(x) = \mathsf{h}^{\mathcal{F}}(x;\alpha)] \cdot c_1\varepsilon + \frac{q}{|\mathcal{F}|} \tag{4.5}$$

$$\leq \frac{1}{2} + (c_1 - 2)\varepsilon \tag{4.6}$$

when $|\mathcal{F}| > q/\varepsilon$ by Lemma 4.3.4. Also we have $\mathbb{E}_{g \xleftarrow{r} G,r}[f_{g,r}(x,g(x))] = \frac{1}{2} + c_1\varepsilon$ by definition. Therefore, $\mathbb{E}_{g \xleftarrow{r} G,r}[f_{g,r}(x,g(x)) - f_{g,r}(x,\mathsf{h}(x;\alpha))] \geq 2\varepsilon$. Let X be the uniform distribution. Note that for different x, $g(x)$ and $\mathcal{F}(x)$ are chosen independently. Therefore $\mathbb{E}_{\mathsf{h},g,r}[f_{g,r}(x,g(x)) - f_{g,r}(x,\mathsf{h}(x))]$[5] for different x are independent since it is only decided by randomness of $g(x)$ and $\mathcal{F}(x)$. By Chernoff-Hoeffding bound, $\mathbb{E}_{x \xleftarrow{r} X}[f_{g,r}(x,g(x)) - f_R(x,\mathsf{h}^{\mathcal{F}}(x;\alpha))] < \varepsilon$ holds with probability $2^{-\Omega(\varepsilon^2 2^n)}$ over the choice of g and r. By taking

[5] The expectation is taken over the local randomness of h, which does not need to be considered in the probabilistic argument.

union bound over $\alpha \in \{0,1\}^a$, we have

$$\forall \alpha \in \{0,1\}^{2^{o(n)}} \ , \ \mathop{\mathbb{E}}_{x \overset{\mathrm{r}}{\leftarrow} X} [f_{g,r}(x, g(x)) - f_{g,r}(x, \mathsf{h}^{\mathcal{F}}(x;\alpha))] \leq \varepsilon \tag{4.7}$$

with probability $2^{-\Omega(\varepsilon^2 2^n)+2^{o(n)}}$, which is less than 1 for large enough n. Thus, there exists a function g and a set $\mathcal{F}$ such that

$$\mathop{\mathbb{E}}_{x \overset{\mathrm{r}}{\leftarrow} X} \left[f_{g,r}(x, g(x)) - f_{g,r}(x, \mathsf{h}^{\mathcal{F}}(x, \alpha)) \right] > \varepsilon. \tag{4.8}$$

By averaging argument, for all α, there exists $f_{g,r} \in \mathcal{F}$ such that $f_{g,r}$ can distinguish $(X, \mathsf{h}^{\mathcal{F}}(X, \alpha))$ and $(X, g(X))$. Therefore the simulation fails no matter what α is, which contradicts to the assumption. Thus there is no ε-simulator with query complexity $c_2(2^\ell \varepsilon^{-2})$.

To summarize, we proved an $\Omega(2^\ell \varepsilon^{-2})$ for black-box simulator, while the upper bound is only $O(\ell 2^\ell \varepsilon^{-2})$. Note that in order to apply Chernoff bound, we need the same-input assumption (i.e. $\mathsf{h}(x;\alpha)$ cannot query $\mathcal{F}(x')$ for $x' \neq x$) to guarantee the independence of different x, even though querying with different input does not seem helpful. A general black-box tight lower bound is left for future work.

Chapter 5

Computational Notions of Quantum Min-Entropy

Computational notions of entropy have many applications in cryptography and complexity theory. In particular, we consider the notions that measure how much (min-)entropy a source X has from the eyes of a computationally bounded party who may hold certain "leakage information" Z that is correlated with X. They have several applications in cryptography, such as leakage-resilient cryptography [DP08], memory delegation [CKLR11], deterministic encryption [FOR15], zero-knowledge [CLP15], pseudorandom generators [HILL99] and other cryptographic primitives [HRVW09, HHR+10, HRVW19], and also have close connections to important results in complexity theory, such as Impagliazzo's hardcore lemma [Imp95], and in additive number theory, such as the Dense Model Theorem [GT08, TZ08, RTTV08].

In this chapter, we initiate the study of computational entropy in the quantum setting, where X and Z may become quantum states and the computationally-bounded observer is modeled as a small quantum circuit. We find that some classical phenomena have (nontrivial) extensions to the quantum setting, but others in the quantum setting behave quite differently and we can even prove that the natural analogues of classical theorems are false. As an application of some of our results, we construct a quantum leakage-resilient stream cipher in the bounded-quantum-storage model, assuming the existence of a quantum-secure pseudorandom generator. We expect that computational notions of quantum entropy will find other natural applications in quantum cryptography. Moreover, by blending quantum information theory and quantum complexity theory, our study may provide new insights and perspectives in both of these areas.

5.1 Introduction

5.1.1 Brief review of quantum information and computation

Recall that a *pure state* in an n-qubit quantum system is a unit vector $|\psi\rangle \in \mathbb{C}^{2^n}$. The standard ("computational") basis is denoted by $\{|x\rangle : x \in \{0,1\}^n\}$ and it represents the set of classical bit strings $x \in \{0,1\}^n$. Until they are *measured* (observed), quantum systems evolve via unitary operations ($2^n \times 2^n$ complex matrices U such that $U^\dagger U = UU^\dagger = I$, where $U^\dagger$ is the conjugate transpose of U). A projective *binary measurement* on the quantum system is given by a linear subspace $A \subseteq \mathbb{C}^{2^n}$. If the system is in state $|\psi\rangle \in \mathbb{C}^{2^n}$, then the result of the measurement is determined by the decomposition $|\psi\rangle = |\psi\rangle_A + |\psi\rangle_{A^\perp}$, where $|\psi\rangle_A$ is the orthogonal projection of $|\psi\rangle$ to A. With probability $\||\psi\rangle_A\|_2^2$, the measurement returns 1 and the system collapses to state $|\psi\rangle_A / \||\psi\rangle_A\|_2$, and with probability $\||\psi\rangle_{A^\perp}\|_2^2$, the measurement returns 0 and the system collapses to state $|\psi\rangle_{A^\perp} / \||\psi\rangle_{A^\perp}\|_2$. We write $\mathsf{D}_A(|\psi\rangle)$ to denote the $\{0,1\}$ random variable that is the outcome of the measurement defined by the space A. Here D_A can be viewed as a (randomized) distinguisher. There is a more general form of binary measurement (described by a "projective operator value measurement" (POVM)), but we only need a projective binary measurement to discuss most concepts in the introduction, and defer the definition of POVMs to where we need it.

A *mixed state* ρ of a quantum system can be specified by a probability distribution $\{p_i\}$ over pure states $\{|\psi_i\rangle\}$. If we evolve ρ by applying a unitary transformation U, it will be in the mixed state given by distribution $\{p_i\}$ over the pure states $\{U|\psi_i\rangle\}$. If instead we perform a measurement defined by the space A on such a mixed state ρ, then by definition, $\Pr\left[\mathsf{D}_A(\rho) = 1\right] = \sum_i p_i \cdot \Pr\left[\mathsf{D}_A(|\psi_i\rangle) = 1\right] = \sum_i p_i \cdot \||\psi_i\rangle_A\|_2^2$. The representation of a mixed state as a probability distribution over pure states is not unique, in that two such representations can yield exactly the same behavior under all sequences of unitary transformations and measurements.[1] For example, the *maximally mixed state* ρ^{mm} is defined as the uniform distribution over the standard classical basis $\{|x\rangle : x \in \{0,1\}^n\}$, but using any orthonormal basis of $\mathbb{C}^{2^n}$ yields an equivalent mixed state (and thus all of them are regarded as the same mixed state ρ^{mm}).

Recall that the *min-entropy* of a classical random variable X is given by

$$\mathsf{H}_{\mathsf{min}}(X) = \min_x \log \frac{1}{\Pr[X = x]} = \log \frac{1}{\max_x \Pr[\mathsf{D}_x(X) = 1]},$$

where D_x is the indicator function for x. When we have a mixed quantum state ρ_X instead of a classical random variable X, we generalize from indicator functions to one-dimensional binary measurements [RW05]. That is, if ρ_X is a mixed quantum state, then:

$$\mathsf{H}_{\mathsf{min}}(X)_\rho = \log \frac{1}{\max_{|\psi\rangle} \Pr[\mathsf{D}_{|\psi\rangle}(\rho_X) = 1]},$$

where $\mathsf{D}_{|\psi\rangle}$ is the binary measurement given by the one-dimensional subspace spanned by $|\psi\rangle$. This generalizes the classical definition. If ρ is given by a distribution $\{p_x\}$ over the classical basis $\{|x\rangle\}$, then $\Pr[\mathsf{D}_{|\psi\rangle}(\rho) = 1] = \sum_x p_x |\langle\psi|x\rangle|^2$ where $\langle\psi|x\rangle$ denotes the standard (Hermitian) inner product between vectors $|\psi\rangle$ and $|x\rangle$. This is maximized by taking $|\psi\rangle = |x^*\rangle$ for $x^* = \arg\max_x p_x$. On the other hand, if ρ is a pure state, with all of its probability on a single unit vector $|\phi\rangle$, then the maximum probability is 1 (yielding *zero* min-entropy), obtained by taking $\psi = \phi$.

Informally, a *quantum circuit* computes on a quantum state (which may be a classical input $|x\rangle$ for $x \in \{0,1\}^n$) by applying a sequence of *local gates*, which are unitary transformations and measurements that apply to only a constant number of qubits in the state. Quantum circuits are also allowed extra *ancilla* qubits (in addition to the n input qubits). We usually require those ancilla qubits to be initialized to $|0^n\rangle$. The *size* of a quantum circuit is the

[1] A unique representation of a mixed state is given by its *density matrix* $\sum_i p_i |\psi_i\rangle\langle\psi_i|$, which is a $2^n \times 2^n$ positive semidefinite matrix of trace one, and thus we use the density matrix formalism in the technical sections of the paper.

number of gates in the circuit.

5.1.2 Quantum computational notions

Quantum indistinguishability. In many applications of cryptography and complexity theory, we only require the security against adversaries with restricted power. Here we consider adversaries with only polynomial-time circuits/algorithms.

In the classical world, there are two different computational models that are widely studied. First, in the *nonuniform computation* model, circuits can depend on the input size, while in the *uniform computation* model, the same algorithm is used for inputs of any size, or equivalently, there is a uniform algorithm that can generate the ensemble of circuits. Once the universal gate set is fixed, we can define the size of a circuit. Then both models can be extended to the quantum setting naturally by replacing circuits with quantum circuits. In this article, we mostly focus on the nonuniform settings, as adversaries have more power in this model. Consider two quantum state ensembles $\{\rho_n\}$ and $\{\sigma_n\}$ where n bounds the number of qubits in ρ_n and σ_n and serves as the security parameter. We say $\{\rho_n\}$ and $\{\sigma_n\}$ are *quantum-indistinguishable* if for every poly(n)-size family of quantum circuits nonuniform quantum algorithm $\{\mathsf{D}_n\}$, we have $\left|\Pr[\mathsf{D}_n(\rho_n) = 1] - \Pr[\mathsf{D}_n(\sigma_n) = 1]\right| \leq \text{negl}(n)$. Sometimes, we consider the asymptotic setting implicitly by omitting the index n. A quantum state ρ on n-qubits is *quantum-pseudorandom* if it is quantum-indistinguishable from the maximally mixed state ρ^{mm}.

Classically, an equivalent way to define a nonuniform circuit ensemble is giving a uniform algorithm (e.g., a Turing machine) an advice string that only depends on the input length. In the quantum setting, this formation of uniform algorithms with advice matches the above definition of nonuniform quantum circuits if we restrict the advice strings to be classical. But one can consider an even more general computational model by giving the circuits arbitrary advice, for example by allowing some of the *quantum* ancilla bits to be initialized to the quantum advice. In this model, the quantum analogue of the classical complexity class **P/poly** is **BQP/qpoly**, which was defined by Nishimura and Yamakami [NY04]. An intriguing and well-known question is whether quantum advice provides more power for solving decision problems. That is, does **BQP/qpoly** = **BQP/poly**?[2] Analogously, one can also define indistinguishability with respect to quantum advice. Some of our results hold in this model as well. However, for the sake of simplicity, in the rest of the introduction, we only consider classical advice.

An interesting fact about quantum indistinguishability (with classical advice) is that there exist *pure* states that are *p*seudorandom (i.e., indistinguishable from the maximally mixed state ρ^{mm}), as shown by Bremner, Mora and Winter [WW08], and Gross, Flammia and Eisert [GFE09]. This is a sharp contrast from the classical setting, as a classical distribution needs min-entropy at least $\omega(\log n)$ to be pseudorandom, but a pure quantum state has *zero* entropy.

Quantum pseudoentropies In this paper, we investigate computational notions of entropy in the quantum setting. One of the most natural ways to define pseudoentropy is that we say a state has *computational (min-)entropy at least k* if it is quantum-indistinguishable from a state with (min-)entropy at least k. If k equals n, the number of qubits of the state, then this is simply the definition of pseudorandomness described above, as the maximally mixed state is the unique state of (min-)entropy n. In the classical setting, this definition of pseudoentropy was proposed by Håstad, Impagliazzo, Levin and Luby [HILL99], who used it as an intermediate step for constructing pseudorandom generators from arbitrary one-way functions and thus it is hereafter referred to as "HILL-type entropy".

Metric-type entropy [BSW03] is another natural definition of computational entropy, which switches the quantifiers in the definition of HILL-type entropy. We say a state ρ_X has metric (min-)entropy at least k if, for every efficient distinguisher, there exists another quantum state $\sigma_{X'}$ with (min-)entropy at least k such that ρ_X and $\sigma_{X'}$ cannot be distinguished by a polynomial-size quantum distinguisher. In the classical case, it is known that the HILL and metric entropies are interchangeable up to some degradation in the size of

[2] A related questions is whether **QMA** = **QCMA**, i.e., whether quantum witnesses are more powerful than classical ones for quantum verifiers? [AK07]

distinguishers [BSW03]. With this equivalence, metric entropy is a useful intermediate notion to obtain tighter security proof in a number of cases (e.g., [DP08, FOR15]). We similarly can show the equivalence in the quantum setting (Theorem 5.3.12) using the "Quantum Nonuniform Min-max Theorem".

There are a number of notions of computational entropy (e.g., Yao-type pseudoentropy [BSW03], inaccessible entropy [HRVW09]) with many different applications and interesting connections to other fields. As the HILL and metric-type entropies are equivalent and they are more natural and widely used notions in the classical setting, we will focus primarily on the HILL-type computational notions.

5.1.3 Quantum nonuniform min-max theorem

Nonuniform Min-max Theorem is formalized in [Zhe13] and it is useful not only in proving the equivalence between HILL and metric (min-)entropies, but also the Leakage Simulation Lemma, Impagliazzo's Hardcore Theorem, and Dense Model Theorem.[3] Here we formulate the Quantum Nonuniform Min-Max Theorem and also use it to prove the equivalence between quantum HILL and metric (min-)entropies and the Quantum Leakage Simulation Lemma later.

Theorem 5.1.1 (informal version of Theorem 5.3.8)**.** *Consider a zero-sum game between two players where the strategy space of Player 1 is a convex set $\mathcal{A} \subseteq \{N\text{-dimensional density matrix}\}$ and the strategy space of Player 2 is $\mathcal{B}$, a set of N-dimensional Hermitian matrix Π with $0 \leq \Pi \leq \mathbb{1}_N$. For strategies $\rho \in \mathcal{A}$ and $\Pi \in \mathcal{B}$, the payoff to Player 2 is $g(\rho, \Pi) \overset{\text{def}}{=} \operatorname{Tr}(\Pi\rho)$. Suppose that for every strategy $\rho \in \mathcal{A}$ of Player 1, there exists a pure strategy $b \in \mathcal{B}$ such that $g(\Pi, \rho) \geq p$. Then for every $\varepsilon \in (0, 1/2)$, there exists a mixed strategy $\widehat{\Pi}$ of Player 2 such that for every strategy $\rho \in \mathcal{A}$ of Player 1, $\mathbb{E}_{\Pi \leftarrow \widehat{\Pi}}[g(\rho, \Pi)] \geq p - \varepsilon$. Moreover, $\widehat{\Pi}$ is the uniform distribution over a multi-set S consisting of at most $O\big(\log N/\varepsilon^2\big)$ strategies in $\mathcal{B}$.*

[3] In [Zhe13], they proved the uniform versions of these theorems using "Uniform Min-max Theorem", which is more general.

Note that, if we restrict the matrices in $\mathcal{A}$ to be diagonalized, namely $a \in \mathcal{A}$ represent a distribution over $[N]$, then the theorem reduces to the classical Nonuniform Min-Max Theorem [Zhe13].

By von Newmann's Min-max Theorem, we already know that there exists a mixed strategy B such that $\mathbb{E}_{\Pi \leftarrow \widetilde{\Pi}}[g(\rho, \Pi)] \geq p$ for all $\rho \in \mathcal{A}$. So the key is to show the existence of "low-complexity" strategy $\widehat{\Pi}$ to "approximate" $\widetilde{\Pi}$. Inspired by the connection between the Nonuniform Min-max Theorem and statistical learning theory observed by Skórski [Sko17], the task reduces to the problem "how many samples from quantum measurements are sufficient for learning quantum states." In [Aar07], Aaronson used the fat-shattering dimension to bound the number of samples needed, while in [CHY15] they used Rademacher complexity to yield a better bound. We use a different method to bound the Rademacher complexity and achieve an even tighter bound, especially when the entropy of $\rho \in \mathcal{A}$ is lower bounded.

5.1.4 Simulate quantum auxiliary input

As introduced in Section 1.3, The (classical) Leakage Simulation Lemma (Theorem 4.2.1) implies many theorems in computational complexity and cryptography which connect to computational entropies. We prove a generalize the Lemma, where the auxiliary input (namely the "Z part", cf., Theorem 4.1.1) becomes a quantum state till holds.

Theorem 5.1.2 (Quantum Leakage Simulating Lemma (informal))**.** *Let $\rho_{XZ} = \sum_{x \in \{0,1\}^n} p_x |x\rangle\langle x| \otimes \rho_Z^x$ be a cq-state consist of n classical bits and ℓ qubits. Leg $\mathcal{D}$ be a family of quantum distinguisher. There exists a quantum circuit C with relative complexity* $\mathrm{poly}(n, 2^\ell, \varepsilon)$ *mapping from n classical bit to ℓ qubits such that for all distinguisher $\mathsf{D} \in \mathcal{D}$,*

$$\left| \mathbb{E}\Big[\mathsf{D}\left(\textstyle\sum_{x \in \{0,1\}^n} p_x |x\rangle\langle x| \otimes \mathsf{C}(x)\right)\Big] - \mathbb{E}\Big[\mathsf{D}(\rho_{XZ})\Big] \right| \leq \varepsilon\,.$$

The main challenge of proving the lemma is that unlike ℓ classical bits, there are infinity many different ℓ-qubit pure states, so we cannot enumerate all possible outputs. We resolved the issue using techniques from *quantum tomography* and again, the generalization bound for Rademacher complexity as we proved the Quantum Nonuniform Min-max Theorem.

Despite in classical settings, the Leakage Simulation Lemma implies a rich class of theorems. Many of the quantum extensions of the implications do not come with the Quantum Leakage Simulating Lemma we have. One of the obstacles is that some implications rely on conditioning on Z part, which only makes sense when Z is classical. Still, the Quantum Leakage Simulating Lemma immediately implies the Leakage Chain Rule for (relaxed)-HILL min-entropy of ccq-state. Also, we have an application of the Quantum Leakage Simulating Lemma in leakage-resilient cryptography—the provable security of Dziembowski and Pietrzak's stream-cipher [DP08] against *quantum leakage* to a quantum adversary with logarithmic quantum storage.

Leakage Chain Rule

Most entropy notions H satisfies the chain rule in the following form

$$H(X|Z) \geq H(X) - \mathrm{len}(Z)$$

where $\mathrm{len}(Z)$ is the length of the variable Z measured in bits/qubits. It is called a "Leakage" Chain Rule because it quantifies how much uncertainty is left in a source X after a short piece of information Z is "leaked". In cryptographic applications, we often consider adversaries that have prior information Y, in which case the leakage chain rule is generalized to

$$H(X|YZ) \geq H(X|Y) - \mathrm{len}(Z).$$

In the classical case, computational analogues of Leakage Chain Rule have a number of applications in cryptography, such as leakage-resilient cryptography [DP08], memory delegation [CKLR11], and deterministic encryption [FOR15]. Before starting our new Leakage Chain Rule for Quantum HILL min-entropy, we review the conditional min-entropies in both classical and quantum setting.

Conditional Min-Entropy. A popular and useful measure of conditional min-entropy in the classical setting is the notion of *average min-entropy* by [DORS08], which has a nice operational interpretation in terms of the guessing probability: Let (X, Z) be a joint distribution over $\{0,1\}^{n+\ell}$. The *guessing probability* of X conditioned on Z is defined as the maximum probability that an algorithm can guess X correctly given Z. That is, $P^{\mathsf{guess}}(X|Z) \stackrel{\text{def}}{=} \max_A \Pr[A(Z) = X]$, where the maximum is taken over *all* (even computationally unbounded) algorithms A. Then the *conditional min-entropy* (also known as *average min-entropy*) of X given Z is defined as $\mathrm{H}_{\mathsf{min}}(X|Z) = -\log(P^{\mathsf{guess}}(X|Z))$.

The definition of conditional min-entropy $\mathrm{H}_{\mathsf{min}}(X|Z)_\rho$ for bipartite quantum states ρ_{XZ} was given by [RW04], which generalizes the aforementioned definition of average min-entropy. For the special case of classical X and quantum Z, which is called a classical-quantum-state (*cq-state*), König, Renner and Schaffner proved that the generalized guessing game described above captures conditional min-entropy [KRS09]. When two parts are quantum (a qq-state), the guessing probability may give higher entropy then Renner's definition. (Instead, an operational interpretation of Renner's definition is as the maximum achievable singlet fraction [KRS09].) In fact, when ρ_{XZ} is entangled, the conditional min-entropy can be negative, which is impossible to capture by a guessing probability.

The cq-state case is particularly useful in quantum cryptography, such as quantum key distribution (QKD) [BB14, Ren08, VV19], device-independent cryptography [VV12, MS16, CSW14] , and quantum-proof randomness extractors [DPVR12]. Also it has a more natural operational interpretation. Thus, we focus on conditional min-entropy for cq-states in this paper, and leave the study of conditional min-entropy for qq-states and computational analogues for future work.

Conditional Pseudoentropy. Classically, for a joint distribution (X, Z), we say that X conditioned on Z has *conditional relaxed HILL pseudo(min-)entropy* at least k if there exists a distribution (X', Z') that is computationally indistinguishable from (X, Z) with $\mathrm{H}_{\mathsf{min}}(X'|Z') \geq k$. (This definition is called *relaxed* HILL (min-)entropy because we do not require that Z' is identically distributed to Z. For short, we will write rHILL to indicate that we are working with the relaxed definition.)

In the quantum setting, let $\rho_{XZ} \in \mathcal{X} \otimes \mathcal{Z}$ be a bipartite state with $n + m$ qubits. We say that X conditioned on Z has *conditional quantum relaxed HILL min-entropy* at least

k (informally written as $\mathrm{H}_{\textsf{r-HILL-min}}(X|Z)_\rho \geq k$) if there exists a quantum state σ_{XZ} such that (i) $\mathrm{H}_{\textsf{min}}(X|Z)_\sigma \geq k$ and (ii) ρ_{XZ} and σ_{XZ} are computationally indistinguishable by all poly(κ)-size quantum distinguishers.

Leakage Chain Rule for quantum HILL entropy. The classical Leakage Chain Rule for relaxed HILL entropy, first proved by [DP08, RTTV08] states that for a joint distribution (X, Y, Z) where Z consists of $\ell = O(\log \kappa)$ bits,

$$\mathrm{H}_{\textsf{r-HILL-min}}(X|Y) \geq k \;\Rightarrow\; \mathrm{H}_{\textsf{r-HILL-min}}(X|Y, Z) \geq k - \ell.$$

(Note that under standard cryptographic assumptions, the analogous statement for (non-relaxed) HILL entropy is false [KPWW16].)

As the corollary of the Quantum Leakage Simulation Lemma, the Leakage Chain Rule can be generalized to handle quantum leakage Z when both the source X and the prior knowledge Y remain classical.

Theorem 5.1.3 (Quantum Leakage Chain Rule (Theorem 5.5.14); informal)**.** *Let ρ_{XYZ} be a ccq-state, where X and Y are classical, and Z consists of ℓ qubits, for $\ell = O(\log \kappa)$, where κ is the security parameter. Then*

$$\mathrm{H}_{\textsf{r-HILL-min}}(X|Y)_\rho \geq k \;\Rightarrow\; \mathrm{H}_{\textsf{r-HILL-min}}(X|Y, Z)_\rho \geq k - \ell.$$

An interesting open question is to prove the Leakage Chain Rule when the source X and/or the prior leakage Y are quantum. In particular, handling a prior quantum leakage Y seems important for applications to leakage-resilient cryptography with quantum leakage (see the following paragraph). This is not likely to be a direct generalization of Theorem 5.1.3 as the information theoretic Leakage Chain Rule loses 2ℓ rather than ℓ bits of entropy [WTHR11]. In Section 5.7.2, we discuss a general barrier to further generalize our proof to handle quantum X and Y as well as many other proofs of classical theorems.

Leakage-resilient stream-cipher against quantum adversary.

In leakage-resilient (or side-channel resilient) cryptography (see [KR19] for a survey), we seek to construct cryptographic protocols that maintain the security even if the side information about the honest parties' secrets leak to an adversary. Here we particularly consider a *quantum leakage.*

A stream cipher is an online and stateful analogue of a pseudorandom generator, where the output length is not determined in advance. It is defined by a function $\mathsf{SC} : \{0,1\}^m \to \{0,1\}^m \times \{0,1\}^n$. Initially, the internal state $S^{(0)}$ is uniformly random over $\{0,1\}^m$. In the i-th round, $(S^{(i)}, X^{(i)}) = \mathsf{SC}(S^{(i-1)})$ is computed, where the output of this round is $X^{(i)}$ and the internal state evolves to be $S^{(i)}$. The security requirement is that for all i, $X^{(i)}$ is pseudorandom given $X^{(1)}, \ldots, X^{(i-1)}$.

Classical leakage-resilient stream ciphers were investigated in the seminal work of Dziembowski and Pietrzak [DP08], where they consider the security of a stream cipher SC in the "only computation leaks" model [MR04] with continual leakage. Specifically, let $S^{(i-1)}$ be the secret state of SC at the beginning of the i-th round. When stream cipher evaluates $(S^{(i)}, X^{(i)}) = \mathsf{SC}(S^{(i-1)})$, an adversary can learn the leakage $\lambda^{(i)}$, which only depends on the part of $S^{(i-1)}$ involved in the computation of $\mathsf{SC}(S^{(i-1)})$. They assume that the leakage functions are efficient and of bounded output length $\ell = O(\log \kappa)$,[4] and proved the following security property: the output of the i-th round remains pseudorandom given the output and leakage from the first $i-1$ rounds. Note that even though the length of each leakage is bounded, the adversary can collect a long leakage accumulated over many rounds.

Dziembowski and Pietrzak [DP08] gave the first construction of leakage-resilient stream-cipher based on randomness extractors and pseudorandom generators, and proved its security using the classical Leakage Chain Rule for HILL entropy. Pietrzak [Pie09] gave a simpler construction based on any family of weak pseudorandom functions, and Jetchev and Pietrzak [JP14] gave an improved analysis of Pietrzak's construction using the (classical)

[4] Both assumptions are necessary. Without the efficiency assumption, the leakage function can invert the secret state and leak on the initial secret s_0 bit by bit; without the length bound, the adversary can learn the entire new secret state.

Leakage Simulation Lemma.

Now we consider the case where the leakage is quantum (the construction of the stream-cipher remains classical). Namely, the outputs of the leakage functions are bounded-length quantum states. It is conceivable that such an attack may exist in the future with the emergence of quantum computers. We prove that the construction of Dziembowski and Pietrzak [DP08] remains secure against quantum leakage in the bounded-quantum-storage model [DFSS08, KT08, WW08], where the adversary has a limited quantum memory (but no restriction on its classical memory).

The reason that we can only prove the security under the bounded-quantum-storage limitation is that we only know how to a simulate quantum auxiliary input when the given jointly distributed string is classical. If the adversary can accumulate the quantum information with unlimited quantum storage, then we need a leakage simulation lemma where the simulator takes quantum inputs.

When proving the quantum security of classical cryptographic constructions, it often suffices to assume the quantum security of the underlying primitives, since typically the security reductions can be directly carried through in the quantum setting. For example, using a classical construction from OWFs to PRGs, a "quantum-proof" OWFs also yields a quantum-proof PRG. Song gives a nice framework to formalize this observation and show a class of reductions satisfies this property [Son14]. However, the leakage-resilient stream-cipher is not the case here due to the presence of quantum side information.[5]

5.1.5 Dense Model Theorem

First, we recall the "computational complexity version" of the Dense Model Theorem by Reingold, Trevisan, Tulsiani, and Vadhan.

Theorem 5.1.4 (Dense Model Theorem [RTTV08]). *Let X, Y, Y' be three distributions on $\{0,1\}^n$ such that Y, Y' are computationally indistinguishable and X is δ-dense in Y'. That*

[5] There are several other challenging cases such as when the reduction needs to rewind the adversary [Wat09, Unr12], or when the setting involves oracles [BDF$^+$11, Zha12] (Leakages from previous rounds).

is, for all $x \in \operatorname{Supp}(X)$, $\Pr[X = x] \leq \frac{1}{\delta} \Pr[Y' = x]$. Then there exists a distribution X' such that X and X' are computationally indistinguishable and X' is δ-dense in Y.

The Dense Model Theorem can also be used to prove the Leakage Chain Rule for conditional HILL min-entropy, and thus the security of leakage-resilient stream-cipher by [DP08]. Another application is in the study of computational differential privacy [MPRV09]. See [Tre11] for more discussions about the applications of Dense Model Theorem.

Using the definitions of HILL-type relative max-entropies (see Definition 5.1.5 for generalized definitions), the Dense Model Theorem can be equivalently stated as

$$\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(X \,\|\, Y) \leq \lambda \quad \Rightarrow \quad \mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(X \,\|\, Y) \leq \lambda$$

for $\lambda = O(\log \kappa)$, where κ is the security parameter. Also, we can prove the converse (Lemma 5.4.14): $\mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(X \,\|\, Y) \leq \lambda \Rightarrow \mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(X \,\|\, Y) \leq \lambda$. Therefore, the two HILL-type computational relative max-entropies are equivalent in the classical setting.

In the quantum settings, first, the definitions of HILL-type computational relative max-entropies can be naturally generalized to quantum states.

Definition 5.1.5 (HILL-type relative max-entropies)**.** *We say the* HILL-1 relative max-entropy *between quantum states ρ and σ is at most λ (written $\mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$) if there exists a quantum state ρ' such that ρ and ρ' are quantum-indistinguishable and $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda$; we say the* HILL-2 relative max-entropy *between quantum states ρ and σ is at most λ (written $\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$) if there exists a quantum state σ' such that σ and σ' are quantum-indistinguishable and $\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma') \leq \lambda$.*

Interestingly, we can show a separation between those two notions, which can be interpreted as the impossibility of the “Quantum Dense Model Theorem”: there exists quantum states ρ, σ such that

$$\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq 1 \text{ but } \mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) = \infty.$$

The counterexample is based on the existence of a *pure* states that is pseudorandom [BMW09, GFE09], which is an interesting phenomenon. As in the classical settings, a random variable must have entropy $\omega(\kappa)$ to be pseudorandom where κ is the security parameter.[6] Note that out negative result does not contradict to the existence of the Leakage Chain rule for conditional relaxed HILL-entropy (Theorem 5.5.14) since in our counterexample, σ is not a maximally mixed state.

5.2 Preliminaries

5.2.1 Quantum information

Quantum state. We begin with some notation. Let $\mathcal{X}$ be a finite-dimensional complex vector space with a Hermitian inner product. A vector in $\mathcal{X}$ is denoted by $|v\rangle$ and its conjugate transpose is denoted by $\langle v| = |v\rangle^\dagger$. The inner product and outer product of two vectors $|v\rangle, |w\rangle \in \mathcal{X}$ are denoted by $\langle v|w\rangle$ and $|v\rangle\langle w|$, respectively. The *norm* of $|v\rangle$ is defined by

$$\left\| |v\rangle \right\|_2 = \sqrt{\langle v|v\rangle}\,.$$

The set of all unit vectors in $\mathcal{X}$ is denoted by $\mathrm{Ball}(\mathcal{X})$. Let $\mathrm{Lin}(\mathcal{X})$ denote the set of all linear operators on $\mathcal{X}$. Let $\mathrm{Herm}(\mathcal{X})$ denote the set of all Hermitian operators on space $\mathcal{X}$, i.e., $\mathrm{Herm}(\mathcal{X}) \overset{\text{def}}{=} \{T \in \mathrm{Lin}(\mathcal{X}) : T^\dagger = T\}$, where $T^\dagger$ is the conjugate transpose of T. The *Hilbert-Schmidt inner product* on $\mathrm{Lin}(\mathcal{X})$ is defined by

$$\langle S\,,\,T\rangle = \mathrm{Tr}\left(S^\dagger T\right)\,,\ \forall S, T \in \mathrm{Lin}(\mathcal{X})\,.$$

A Hilbert space of a quantum system X is denoted by the corresponding calligraphic letter $\mathcal{X}$. When the quantum system consists of m qubits, the space is the complex Euclidean vector space $\mathcal{X} = \mathbb{C}^{2^m}$. An m-qubit quantum state is represented by a *density operator* $\rho \in \mathrm{Herm}(\mathcal{X})$, which is a positive semidefinite Hermitian operator on $\mathcal{X}$ with trace one. When ρ is of rank one, it refers to a *pure* quantum state, which can also be represented by a unit vector $|v\rangle$ in $\mathrm{Ball}(\mathcal{X})$. In that case, the density operator ρ can be written as $|v\rangle\langle v|$. Otherwise, the density operator ρ refers to a *mixed* quantum state. Thus in any basis that diagonalizes ρ, we can

[6] If we consider quantum distinguishers with quantum advice, then similar to the classical case, a random variable must have entropy $\omega(\kappa)$ to be quantum pseudorandom.

think of ρ as a classical distribution on the pure states corresponding to the basis elements. In general, the expression is not unique. The set of all quantum density operators on $\mathcal{X}$ is denoted by

$$\mathrm{Dens}(\mathcal{X}) \stackrel{\text{def}}{=} \{\rho \in \mathrm{Herm}(\mathcal{X}) : \rho \geq 0, \mathrm{Tr}(\rho) = 1\} = \mathrm{Conv}\Big(\{|v\rangle\langle v| : |v\rangle \in \mathrm{Ball}(\mathcal{X})\}\Big),$$

where $\mathrm{Conv}(\mathcal{S})$ is the convex hull of set $\mathcal{S}$ and $\rho \geq 0$ means that ρ is positive semidefinite. Likewise, $\sigma \geq \rho$ means that $\sigma - \rho$ is positive semidefinite. Let $\mathbb{1}_{\mathcal{X}}$ denote the identity operator on $\mathcal{X}$ (or $\mathbb{1}_d$ when the dimension of $\mathcal{X}$ (denoted as $\dim(\mathcal{X})$) is known to be d), $\rho_{\mathcal{X}}^{\mathsf{mm}} = \frac{1}{\dim(\mathcal{X})}\mathbb{1}_{\mathcal{X}}$ denotes the maximally mixed state in $\mathrm{Dens}(\mathcal{X})$ (or ρ_d^{mm} when $\dim(\mathcal{X}) = d$).

Composite system and partial trace. The Hilbert space of the composite system of two quantum systems X and Y is their tensor product space $\mathcal{X} \otimes \mathcal{Y}$, and similarly for multiple systems. For a multi-partite state, e.g., $\rho_{XYZ} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z})$, its reduced state on some subsystem is represented by the same state with the corresponding subscript. For example, the reduced (marginal) state on system X of ρ_{XYZ} is $\rho_X = \mathrm{Tr}_{YZ}(\rho_{XYZ})$, where $\mathrm{Tr}_{YZ}(\cdot)$ denotes the *partial trace* operation over the composite system YZ. That is, $\mathrm{Tr}_{YZ}(|x_1\rangle\langle x_2| \otimes |y_1\rangle\langle y_2| \otimes |z_1\rangle\langle z_2|) = |x_1\rangle\langle x_2| \in \mathrm{Dens}(\mathcal{X})$, where $|x_i\rangle, |y_i\rangle, |z_i\rangle$ for $i = 1, 2$ are vectors in $\mathrm{Ball}(\mathcal{X}), \mathrm{Ball}(\mathcal{Y}), \mathrm{Ball}(\mathcal{Z})$, respectively, and Tr_{YZ} is a trilinear map on YZ. When all subscript letters are omitted, the notation represents the original state (e.g., $\rho = \rho_{XYZ}$). A bipartite state ρ_{XY} is call a *product state* if and only if $\rho_{XY} = \rho_X \otimes \rho_Y$. A bipartite state ρ_{XY} is *separable* if and only if ρ_{XY} can be written as $\sum_k p_k \cdot \rho_X^k \otimes \rho_Y^k$.

A classical discrete random variable X with distribution $p_x = \Pr[X = x]$ can be represented by a density operator $\rho = \sum_x p_x |x\rangle\langle x|$ over state space $\mathcal{X}$ with orthonormal basis $\{|x\rangle\}$ of $\mathcal{X}$. When restricted to the basis $|x\rangle$, we will say that the system X is classical. A *classical-quantum*-state, or *cq-state* $\rho \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ indicates that subsystem X is classical and subsystem Y is quantum. We use lower case letters to denote specific values assigned to the classical part of a state. Then a cq-state can be represented (uniquely) in the form $\rho_{XY} = \sum_x p_x |x\rangle\langle x| \otimes \rho_Y^x$, where $p_x = \Pr[X = x]$ and $\rho_Y^x \in \mathrm{Dens}(\mathcal{Y})$. The marginal state ρ_Y is $\sum_x p_x \rho_Y^x$.

Quantum measurements. A *positive-operator valued measure (POVM)* on the Hilbert space $\mathcal{X}$ with outcomes in $[k]$ is a collection of Hermitian and positive semidefinite operators $\{\Pi_i\}_{i\in[k]}$ such that $\sum_{i\in[k]} \Pi_i = \mathbb{1}_{\mathcal{X}}$. Each POVM element Π_i can serves as an instrument to perform a yes-no measurement. We denote the space of possible POVM elements Π as

$$\mathrm{Meas}(\mathcal{X}) \stackrel{\text{def}}{=} \left\{\Pi : \Pi \in \mathrm{Herm}(\mathcal{X}), 0 \leq \Pi \leq \mathbb{1}_{\mathcal{X}}\right\}.$$

When this POVM is applied to a quantum state $\rho \in \mathrm{Dens}(\mathcal{X})$, the probability of obtaining outcome $i \in [k]$ is $\langle \Pi_i \, , \, \rho \rangle$. If outcome i is observed, the quantum state ρ will collapse to the state $\sqrt{\Pi_i}\rho\sqrt{\Pi_i}/\langle \Pi_i \, , \, \rho \rangle$, where $\sqrt{\Pi}$ is the unique positive semidefinite operator T such that $T^2 = \Pi$.

Matrix Norms. The *trace norm* of $T \in \mathrm{Lin}(\mathcal{X})$ is defined as

$$\left\|T\right\|_1 \stackrel{\text{def}}{=} \mathrm{Tr}\left(\sqrt{T^\dagger T}\right).$$

One important measure on the distance between two quantum states $\rho, \sigma \in \mathrm{Dens}(\mathcal{X})$ is the *trace distance*, defined as

$$\mathrm{d}_{\mathsf{Tr}}(\rho \, , \, \sigma) \stackrel{\text{def}}{=} \frac{1}{2}\left\|\rho - \sigma\right\|_1 ,$$

which equals the total variation distance between ρ and σ if they are both classical. Similar to the classical case, the trace distance of two quantum states is an upper bound on the difference of their probabilities of obtaining the same measurement outcome [NC02]:

$$\mathrm{d}_{\mathsf{Tr}}(\rho \, , \, \sigma) = \max_{\Pi \in \mathrm{Meas}(\mathcal{X})} \mathrm{Tr}\left(\Pi(\rho - \sigma)\right).$$

Also, trace distance is contractive under applying a general quantum circuits (a.k.a. TPCP maps or quantum operations).

Proposition 5.2.1. *Let* $\rho, \sigma \in \mathrm{Dens}(\mathcal{X})$ *and let* C *be a general quantum circuits mapping from* $\mathrm{Dens}(\mathcal{X})$ *to* $\mathrm{Dens}(\mathcal{Y})$*. Then we have*

$$\mathrm{d}_{\mathsf{Tr}}(C(\rho) \, , \, C(\sigma)) \leq \mathrm{d}_{\mathsf{Tr}}(\rho \, , \, \sigma).$$

The *operator norm* of $T \in \mathrm{Lin}(\mathcal{X})$ is

$$\|T\|_{\mathsf{op}} \stackrel{\text{def}}{=} \sup\Big\{\big\|T|v\rangle\big\|_2 : |v\rangle \in \mathrm{Ball}(\mathcal{X})\Big\}\,.$$

When T is Hermitian, the operator norm of T equals the magnitude of the largest eigenvalue of T. Once we fix an orthonormal basis $\{|i\rangle\}$ of $\mathcal{X}$, the *max norm* of $T \in \mathrm{Lin}(\mathcal{X})$ is defined as

$$\|T\|_{\mathsf{max}} \stackrel{\text{def}}{=} \max_{i,j} |T_{ij}|\,,$$

where $T_{ij} = \langle i|T|j\rangle$. We can connect $\|T\|_{\mathsf{max}}$ to $\|T\|_{\mathsf{op}}$ by the following inequality.

$$\|T\|_{\mathsf{op}} \leq \dim(\mathcal{X}) \cdot \|T\|_{\mathsf{max}}\,. \tag{5.1}$$

For operational interpretation of these norms, see [HJ12].

Quantum circuits. The evolution of a closed quantum system X is described by a unitary operator $U \in \mathrm{Lin}(\mathcal{X})$, i.e., an operator U satisfying $UU^\dagger = U^\dagger U = \mathbb{1}_{\mathcal{X}}$. The quantum system then evolves from state $\rho \in \mathrm{Dens}(\mathcal{X})$ to $U\rho U^\dagger \in \mathrm{Dens}(\mathcal{X})$. If $\rho = |\psi\rangle\langle\psi|$, then $U\rho U^\dagger = |\phi\rangle\langle\phi|$ for $|\phi\rangle = U|\psi\rangle \in \mathrm{Ball}(\mathcal{X})$. Herein we consider a multipartite system, where each subsystem is a two-dimensional quantum system $\mathbb{C}^2$ with an ordered computational basis $\{|0\rangle, |1\rangle\}$. A quantum state in $\mathrm{Dens}(\mathbb{C}^2)$ is called a *qubit* (quantum bit), as opposed to a classical bit 0 or 1. Thus an m-qubit state space is $\mathbb{C}^{2^m}$ with a computational basis $\big\{|x\rangle : x \in \{0,1\}^m\big\}$. Simple unitary operators that act non-trivially on a constant number of qubits are called *elementary quantum gates*. A set of elementary quantum gates is called *universal* if any unitary operator can be approximated arbitrarily closely by a composition of gates from this set. We fix one universal gate set for the remainder of this paper.

Let $\mathcal{W} = \mathcal{X} \otimes \mathcal{A} = \mathbb{C}^{2^m}$ denote the work space of a quantum circuit C, which is an m-qubit space that consists of both an ℓ-qubit input space $\mathcal{X} = \mathbb{C}^{2^\ell}$, taking some quantum/classical input $\rho \in \mathrm{Dens}(\mathcal{X})$, and some $m - \ell$ ancilla qubits initialized as $\tau \in \mathrm{Dens}(\mathcal{A})$. Usually, we assume $\tau = |0_{\mathcal{A}}\rangle$, meaning that the circuit has only classical nonuniform advice (corresponding to gates and wires). Occasionally, we allow for quantum advice, where τ could be an arbitrary quantum state. A quantum circuit C is a sequence of elementary quantum gates from the universal gate set, followed by some measurements. (In general, measurements can be deferred to the end of quantum circuits [NC02]). That is, C applies a unitary $U_{\mathsf{C}} = U_1 U_2 \cdots U_t$ where U_i denotes the ith gate and s is the number of elementary quantum gates, and the performs some measurements. We say the size of the quantum circuit C is t. The number of quantum circuits of size t is $t^{O(t)}$.

Quantum distinguishers. In cryptography, we usually have a circuit with binary output as a distinguisher between two random variables. Here we define the quantum analogue. A *quantum distinguisher* is a quantum circuit with binary measurement outcome 0 or 1. Without loss of generosity, we assume that we measure after applying a unitary U_{C}. That is, we measure $\rho' = U_{\mathsf{C}}(\rho \otimes \tau)U_{\mathsf{C}}^{\dagger}$ according to the POVM $\{\Pi_0, \Pi_1\}$, where τ is the initial ancilla state and $\Pi_i = |i\rangle\langle i| \otimes \mathbb{1}_{2^{m-1}} \in \mathrm{Meas}(\mathcal{W})$. Thus

$$\begin{aligned}
\Pr[\mathsf{C} \text{ outputs } i \text{ on input } \rho] =& \langle \rho' \,,\, \Pi_i \rangle \\
=& \left\langle U_C(\rho \otimes \tau)U_{\mathsf{C}}^{\dagger} \,,\, |i\rangle\langle i| \otimes \mathbb{1}_{2^{m-1}} \right\rangle \\
=& \left\langle \rho \otimes \tau \,,\, U_C^{\dagger}(|i\rangle\langle i| \otimes \mathbb{1}_{2^{m-1}})U_{\mathsf{C}} \right\rangle \\
=& \langle \rho \,,\, \Pi_i' \rangle,
\end{aligned}$$

where $\Pi_i' = \mathrm{Tr}_A\big((\mathbb{1}_{\mathcal{X}} \otimes \tau)U_{\mathsf{C}}^{\dagger}(|i\rangle\langle i| \otimes \mathbb{1}_{2^{m-1}})U_{\mathsf{C}}\big)$.

Consequently, applying this quantum circuit is equivalent to applying a corresponding POVM $\{\Pi_0', \Pi_1'\}$ on the input space $\mathcal{X}$ as above. For our purpose, a *quantum distinguishers* will be considered as binary POVMs on the space $\mathcal{X}$. Since $\Pi_0 + \Pi_1 = \mathbb{1}_{\mathcal{X}}$, the POVM can be fully determined by Π_1. Therefore, we can describe a quantum distinguisher by a measurement operator Π and vice versa. In particular, we say the corresponding measurement operator of the distinguisher $\mathsf{D} : \mathrm{Dens}(\mathcal{X}) \to \{0,1\}$ is $\Pi \in \mathrm{Meas}(\mathcal{X})$ if

$$\forall \rho \in \mathrm{Dens}(\mathcal{X}) \ , \ \mathbb{E}[\mathsf{D}(\rho)] = \langle \Pi \,,\, \rho \rangle \,.$$

One can easily generalize the binary output to larger domains. In that case, any quantum circuit can still be effectively deemed as a general POVM with a large outcome set. We also consider more general quantum circuits that output general quantum states. These circuits can be deemed as mappings from $\mathrm{Dens}(\mathcal{X})$ to $\mathrm{Dens}(\mathcal{Y})$, where $\mathcal{X}$ is the input space and $\mathcal{Y}$ is the output space. (In general, these mappings are called super-operators from $\mathrm{Lin}(\mathcal{X})$ to $\mathrm{Lin}(\mathcal{Y})$.) Similar to quantum distinguishers, a general quantum circuit C applies a unitary U_{C} on the space $\mathcal{W} = \mathcal{X} \otimes \mathcal{A}$ consisting of an input and ancillas, and perform some measurements on $\mathcal{W}$. Then it outputs a state in space $\mathcal{Y}$ where $\mathcal{W} = \mathcal{Y} \otimes \mathcal{B}$ is the decomposition of the space after applying U_{C}. We abuse the notation for convenience as

$$\rho \mapsto \mathsf{C}(\rho) \in \mathrm{Dens}(\mathcal{Y})$$

for input $\rho \in \mathrm{Dens}(\mathcal{X})$, so

$$\mathsf{C}(\rho) = \mathrm{Tr}_B\Big(U_{\mathsf{C}}\,(\rho \otimes \tau)\,U_{\mathsf{C}}^{\dagger}\Big),$$

where τ is the ancilla state.

Uniformity. A family of uniform quantum circuits $\{\mathsf{C}_n\}_{n\in\mathbb{N}}$ is a set of circuits indexed by n where the inputs length (measured in bit/qubit) of C_n is n, and there exists a polynomial time classical algorithm A such that $\mathsf{A}(1^n) = \mathsf{C}_n$. **BQP** is a class of language $\mathcal{L}$ for which there exists a (family of) uniform quantum circuits $\{\mathsf{C}_n\}_{n\in\mathbb{N}}$ with binary outputs such that for all $n \in \mathbb{N}$ and $x \in \{0,1\}^n$, if $x \in \mathcal{L}$, then $\Pr[\mathsf{C}_n(x) = 1] > 2/3$, Otherwise, $\Pr[\mathsf{C}_n(x) = 1] < 1/3$.

5.2.2 Information-theoretic notions

Definition 5.2.2 (Quantum min-entropy)**.** *Let ρ be a density operator on state space $\mathcal{X}$. Then min-entropy of ρ is defined as*

$$\mathrm{H}_{\mathsf{min}}(\rho) = \mathrm{H}_{\mathsf{min}}(X)_\rho \stackrel{\text{def}}{=} \log \frac{1}{\lambda_{\max}}\,,$$

where $\lambda_{\max}$ is the largest eigenvalue of ρ.

To define the conditional quantum min-entropy [RW05], we first define the max-divergence (a.k.a. max-relative entropy) between two quantum states. Max-divergence can be seen as a distance between two quantum states, which measures, in log-scale, how much more likely an event happens for one state than for the other.

Definition 5.2.3 ((quantum) max-divergence)**.** *Let ρ and σ be two density operators on a space $\mathcal{X}$. The max-relative entropy between two quantum states ρ and σ is defined as*

$$\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma) \overset{\text{def}}{=} \inf\Big\{\lambda \in \mathbb{R} : \rho \leq 2^{\lambda}\sigma\Big\}.$$

Equivalently, quantum max-divergence can be defined in an operational way using binary measurement.

Proposition 5.2.4. *Let ρ and σ be density operators on a state space $\mathcal{X}$. Then*

$$\begin{aligned}\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma) &= \log\Big(\sup\Big\{\frac{\langle \Pi\,,\, \rho\rangle}{\langle \Pi\,,\, \sigma\rangle} : \Pi \in \mathrm{Meas}(\mathcal{X})\Big\}\Big)\\ &= \log\Big(\sup\Big\{\frac{\Pr[\mathsf{D}(\rho)=1]}{\Pr[\mathsf{D}(\sigma)=1]} : \mathsf{D} \textit{ is a quantum distinguisher}\Big\}\Big)\end{aligned}$$

Proof. It suffices to show that for $\gamma > 0$, $\gamma\cdot\sigma \geq \rho$ iff $\gamma\cdot\langle \Pi\,,\, \sigma\rangle \geq \langle \Pi\,,\, \rho\rangle$ for every $\Pi \in \mathrm{Meas}(\mathcal{X})$.

If $\gamma\cdot\sigma - \rho \geq 0$, then $\langle \Pi\,,\, \gamma\cdot\sigma - \rho\rangle \geq 0$, since $\langle A\,,\, B\rangle \geq 0$ for A and B are positive semidefinite and Hermitian (Let $A = \sum_i \lambda_i |\psi_i\rangle\langle\psi_i|$, then $\langle A\,,\, B\rangle = \sum_i \lambda_i \langle\psi_i|B|\psi_i\rangle \geq 0$). On the other hand, suppose $\langle \Pi\,,\, \gamma\cdot\sigma - \rho\rangle \geq 0$ for every $\Pi \in \mathrm{Meas}(\mathcal{X})$. For every $|\psi\rangle \in \mathrm{Ball}(\mathcal{X})$, taking $\Pi = |\psi\rangle\langle\psi|$, we have

$$\Big\langle |\psi\rangle\langle\psi|\,,\, \gamma\cdot\sigma - \rho\Big\rangle = \Big\langle \psi\Big|(\gamma\cdot\sigma - \rho)\Big|\psi\Big\rangle \geq 0.$$

Thus $\gamma\cdot\sigma - \rho \geq 0$. The formulation in terms of quantum distinguishers D follows from the fact that every distinguisher D has an associated measurement operator and conversely every measurement operator can be approximated arbitrarily well by a distinguisher. □

The definition of quantum max-divergence agrees with the definition of classical max-divergence definition when the two quantum states are equivalent to classical random variables.

Proposition 5.2.5. *If ρ and σ are mixed quantum states corresponding to two discrete classical random variables X_ρ and X_σ respectively. Then*

$$\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma) = \log\Big(\max_{x\in\mathrm{Supp}(X_\rho)} \frac{\Pr[X_\rho = x]}{\Pr[X_\sigma = x]}\Big).$$

Proof. By the assumption, we can write $\rho = \sum_x p_x |x\rangle\langle x|$ and $\sigma = \sum_x q_x |x\rangle\langle x|$ for $\Pr[X_\rho = x] = p_x, \Pr[X_\sigma = x] = q_x$. Then

$$\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma) = \inf\left\{\lambda : \forall x \;\; p_x \leq 2^\lambda q_x\right\} = \log\left(\max_{q_x > 0} \frac{p_x}{q_x}\right) = \log\left(\max_{x \in \mathrm{Supp}(X_\sigma)} \frac{\Pr[X_\rho = x]}{\Pr[X_\sigma = x]}\right).$$

□

Definition 5.2.6 (Conditional quantum min-entropy)**.** *Let $\rho = \rho_{XY} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ be a density operator describing a bipartite quantum system (X, Y). The min-entropy of system X conditioned on system Y is defined as*

$$\mathrm{H}_{\mathsf{min}}(X|Y)_\rho \overset{\mathrm{def}}{=} \log(\dim(\mathcal{X})) - \inf_{\sigma_Y \in \mathrm{Dens}(\mathcal{Y})} \left\{\mathrm{D}_{\mathsf{max}}(\rho_{XY} \,\|\, \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \sigma_Y)\right\}.$$

Similar to the *conditional von Newman entropy*, the quantum min-entropy can be negative as opposed to the classical cases [CA97]. In particular, when a bipartite state has entanglement, the conditional min-entropy may be negative. For instance, let $\rho \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ where its systems X and Y are maximally entangled, $\mathrm{H}_{\mathsf{min}}(X|Y)_\rho = -\log\dim(\mathcal{X})$ [CA97, KRS09].

Proposition 5.2.7. *If X and Y are discrete classical random variables, then*

$$\mathrm{H}_{\mathsf{min}}(X|Y)_\rho = \frac{1}{\log \sum_y \max_x p_{xy}} = \mathrm{H}_{\mathsf{min}}(X|Y),$$

where $\mathrm{H}_{\mathsf{min}}$ is the definition of average min-entropy in the classical case (Definition 1.5.2).

Proof. Since X, Y are classical random variables, we abuse the notation $\mathcal{X}, \mathcal{Y}$ to be the finite spaces that X and Y are distributed over, respectively. Let $U_{\mathcal{X}}$ be the uniform distribution over the set $\mathcal{X}$. The following claim shows that it suffices to only consider diagonal density operator σ_Y for the infimum in Definition 5.2.6 when ρ_{XY} is diagonal.

Claim 5.2.8. *Let ρ_{XY} and σ_X be diagonal density matrices*

$$\inf_{\sigma_y \in \mathrm{Dens}(\mathcal{Y})} \mathrm{D}_{\mathsf{max}}\left(\rho_{XY} \,\Big\|\, \sigma_X \otimes \sigma_Y\right) = \inf_{\substack{\sigma_y \in \mathrm{Dens}(\mathcal{Y}) \\ \sigma_y \text{ is diagonalized}}} \mathrm{D}_{\mathsf{max}}\left(\rho_{XY} \,\Big\|\, \sigma_X \otimes \sigma_Y\right)$$

Proof of Claim 5.2.8. Suppose $\sigma_Y \in \mathrm{Dens}(\mathcal{Y})$ satisfies that $\gamma \cdot \sigma_X \otimes \sigma_Y - \rho_{XY} \geq 0$, Let σ'_Y be the diagonal matrix having the same diagonal entries as σ_y. Clearly, $\sigma'_Y \in \mathrm{Dens}(\mathcal{Y})$.

$\gamma \cdot \sigma_X \otimes \sigma'_Y - \rho_{XY}$ is diagonal. Moreover, $\gamma \cdot \sigma_X \otimes \sigma_Y - \rho_{XY}$ is positive semidefinite implies the diagonal entries are non-negative. Thus, $\gamma \cdot \sigma_X \otimes \sigma'_Y - \rho_{XY} \geq 0$. □

By Definition 5.2.6,

$$\begin{aligned} \mathrm{H}_{\mathsf{min}}(X|Y)_\rho &= \log(\dim(\mathcal{X})) - \inf_{\sigma_Y \in \mathrm{Dens}(\mathcal{Y})} \mathrm{D}_{\mathsf{max}}\Big(\rho_{XY} \,\Big\|\, \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \sigma_Y\Big) \\ &= \log(\dim(\mathcal{X})) - \inf_{Y'\text{: dist. over } \mathcal{Y}} \mathrm{D}_{\mathsf{max}}\Big(X, Y \,\Big\|\, U_{\mathcal{X}}, Y'\Big) \qquad \text{(Claim 5.2.8)} \\ &= \log(\dim(\mathcal{X})) - \log \inf_{Y'\text{: dist. over } \mathcal{Y}} \Big\{ \max_{x,y} \frac{p_{xy}}{\Pr[(U_{\mathcal{X}}, Q) = (x,y)]} \Big\} \\ &= -\log \inf_{\sum_y q_y = 1} \Big\{ \max_y \frac{\max_x p_{xy}}{q_y} \Big\} \\ &= -\log \Big(\sum_y \max_x p_{xy} \Big) \end{aligned}$$

where $q_y = \Pr[Y' = y]$. The last equality is by

$$\max_y \frac{\max_x p_{xy}}{q_y} \geq \frac{\sum_y \max_x p_{xy}}{\sum_y q_y} = \sum_y \max_x p_{xy} \,,$$

with equality if and only if $(\max_x p_{xy})/q_y$ is constant. □

Another way to define min-entropy is through guessing probability. Here we only consider the case that ρ_{XY} is a cq-state: $\rho_{XY} = \sum_x p_x |x\rangle\langle x| \otimes \rho_Y^x \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$. Recall that a quantum circuit with classical output can be seen as a POVM. The probability of guessing X correctly given Y by a given quantum circuit C is

$$P_{\mathsf{C}}^{\mathsf{guess}}(X|Y)_\rho \stackrel{\text{def}}{=} \sum_x p_x \langle \Pi_x \,,\, \rho_Y^x \rangle \,,$$

where $\{\Pi_x\}$ is the effective POVM for C, demonstrating the guessing strategy. Accordingly, the probability of guessing X correctly given Y is defined as

$$P^{\mathsf{guess}}(X|Y)_\rho = \sup_{\mathsf{C}} P_{\mathsf{C}}^{\mathsf{guess}}(X|Y)_\rho,$$

where the maximization is taken over arbitrary quantum circuits C of unbounded size. As in the purely classical case [DORS08], the guessing probability captures the conditional min-entropy of X given Y:

Lemma 5.2.9 ([KRS09]). *Suppose ρ_{XY} is a cq-state on the space $\mathcal{X} \otimes \mathcal{Y}$. Then*

$$\mathrm{H}_{\mathsf{min}}(X|Y)_\rho = \log \frac{1}{P^{\mathsf{guess}}(X|Y)_\rho} .$$

5.3 Quantum Pseudoentropy

One of the first proposed notions of (classical) pseudoentropy is by Yao [Yao82], which based on efficient compression. Then, Håstad, Impagliazzo, Levin, and Luby introduced another class of pseudoentropies [HILL99] based on *computational indistinguishability.* We call them HILL-type entropies. Also, there is another class of pseudoentropy notion called metric-type entropy defined by Barak, Shaltiel, and Wigderson [BSW03], which is also based on computational indistinguishability. In this section, we extend those types of pseudoentropy to quantum settings and study their properties. First, we define *quantum (computational) indistinguishability.*

5.3.1 Quantum indistinguishability

Computational indistinguishability is a fundamental concept in computational complexity and cryptography. It provides a relaxed way to describe the similarity of two random objects. Informally, computational indistinguishability only requires that two random objects cannot be distinguished by efficient algorithms/circuits. Two objects may be indistinguishable by bounded algorithms even if they are statistically very far from each other (e.g., have very different entropies).

We consider two variants of indistinguishability in the quantum setting, depending on whether the ancilla bits are initialized to 0 (so the circuit can only have classical nonuniform advice corresponding to the gates and wires) or whether the ancilla qubits can be initialized to an arbitrary quantum state (corresponding to quantum advice).

Definition 5.3.1. *Quantum states ρ and σ on* $\mathrm{Dens}(\mathcal{X})$ *are* (t, ε)-quantum-indistinguishable *if for all quantum distinguishers* D *of size* t *with ancilla qubits all initialized to* $|0\rangle$*'s,*

$$\Big|\Pr[\mathsf{D}(\rho) = 1] - \Pr[\mathsf{D}(\sigma) = 1]\Big| \leq \varepsilon.$$

Moreover, we say that ρ *is an* (t, ε)-quantum-pseudorandom *state if* ρ *is* (t, ε)*-quantum-indistinguishable from the maximally mixed state on* $\mathrm{Dens}(\mathcal{X})$*.*

Definition 5.3.2. *Quantum states* ρ *and* σ *on* $\mathrm{Dens}(\mathcal{X})$ *are* (t, ε)-quantum$^+$-indistinguishable *if for all quantum distinguishers* D *of size* t *with arbitrary ancilla qubits,*

$$\Big|\Pr[\mathsf{D}(\rho) = 1] - \Pr[\mathsf{D}(\sigma) = 1]\Big| \leq \varepsilon.$$

Moreover, we say that ρ *is an* (t, ε)-quantum$^+$-pseudorandom *state if* ρ *is* (t, ε)*-quantum*$^+$*-indistinguishable from the maximally mixed state on* $\mathrm{Dens}(\mathcal{X})$*.*

Now we give an asymptotic formulation of the above definitions.

Definition 5.3.3. *Let* $t : \mathbb{N} \to \mathbb{N}$ *and* $\varepsilon : \mathbb{N} \to \mathbb{R}$ *be two functions. Let* $\{\rho_n\}_{n\in\mathbb{N}}$ *and* $\{\sigma_n\}_{n\in\mathbb{N}}$ *be two quantum state ensembles where* $\rho_n, \sigma_n \in \mathrm{Dens}(\mathbb{C}^{2^n})$*. We say* $\{\rho_n\}_{n\in\mathbb{N}}$ *and* $\{\sigma_n\}_{n\in\mathbb{N}}$ *are* $(t(n), \varepsilon(n))$-quantum-indistinguishable *(resp.,* $(t(n), \varepsilon(n))$-quantum$^+$-indistinguishable*), if for every* $n \in \mathbb{N}$*,* ρ_n *and* σ_n *are* $(t(n), \varepsilon(n))$*-quantum-indistinguishable (resp.,* $(t(n), \varepsilon(n))$*-quantum*$^+$*-indistinguishable).*

We say that ρ_n *and* σ_n *are* quantum-indistinguishable *(resp.,* quantum$^+$-indistinguishable) if they are $(t(n), \varepsilon(n))$-quantum-indistinguishable (resp., $(s(n), \varepsilon(n))$-quantum$^+$-indistinguishable) for some functions $t(n) = n^{\omega(1)}$, $\varepsilon(n) = n^{-\omega(1)}$.

5.3.2 Pseudo (min-)entropy

Similar to the definition of pseudorandomness as a computational analogue of the uniform distribution, one can naturally generalize the concept of entropy in information theory to computational notions of entropy. In the classical setting, the definition of HILL-type entropy says that a random variable X has HILL (min-)entropy at least k if it is indistinguishable from some random variable X' with (min-)entropy at least k. Another natural definition of computational entropy is *metric-type entropy* which switches the quantifiers in the definition of HILL-type entropy. That is, X has metric (min-)entropy at least k if for every efficient distinguisher D, there exists a random variable X' with (min-)entropy at least k such that X and X' cannot be distinguished by the D.

Recall that one can equivalently define the conditional min-entropy using guessing probability (cf., Lemma 5.2.9). We can also get a relaxed notion by restricting the complexity of guessing algorithms, and we call it *guessing pseudoentropy.*

Below, we formally define the quantum analogues of those relaxed notions.

Definition 5.3.4 (Conditional (relaxed-)HILL (min-)entropy)**.** *Let* $\rho = \rho_{XY}$ *be a bipartite quantum state in* $\mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$*. We say* X *conditioned on* Y *has* (t, ε)-relaxed-HILL (min-)entropy *at least* k *(written "*$\mathrm{H}^{t,\varepsilon}_{\mathsf{r\text{-}HILL\text{-}min}}(X|Y)_\rho \geq k$*") if there exists a bipartite quantum state* $\sigma_{XY} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ *such that*

1. $\mathrm{H}_{\mathsf{min}}(X|Y)_\sigma \geq k$*.*

2. ρ_{XY} *and* σ_{XY} *are* (t, ε)*-quantum-indistinguishable.*

In addition, if $\mathrm{Tr}_X(\rho_{XY}) = \mathrm{Tr}_X(\sigma_{XY})$*, we say* X *conditioned on* Y *has (standard)* HILL (min-)entropy *at lease* k *(written "*$\mathrm{H}^{t,\varepsilon}_{\mathsf{HILL\text{-}min}}(X|Y)_\rho \geq k$*").*

As in the classical case [HLR07], we do not require the reduced states ρ_Y and σ_Y being equal in relaxed-HILL (min-)entropy. In the classical case, the relaxed HILL notion satisfies a chain rule even when a prior knowledge is present, while for standard HILL (min-)entropy, a counterexample exists (under a standard assumption) [KPWW16]. Also, in the classical case, when the length of Y is $O(\log n)$, the two definitions are equivalent up to a $\mathrm{poly}(n)$ factor in s. However, we do not know whether that is still the case if Y is a quantum state of $O(\log n)$ qubits.

We now state the quantum analogues of metric entropy and guessing pseudoentropy.

Definition 5.3.5 (conditional (relaxed-)metric (min-)entropy)**.** *Let* $\rho = \rho_{XY}$ *be a bipartite quantum state in* $\mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$*. We say that* X *conditioned on* Y *has* (t, ε)-relaxed-metric (min-)entropy *at lease k (written "$\mathrm{H}^{t,\varepsilon}_{\textsf{r-metric-min}}(X|Y)_\rho \geq k$") if for every quantum distinguisher* D *of size t, there exists a bipartite quantum state $\sigma_{XY} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ such that*

1. $\mathrm{H}_{\textsf{min}}(X|Y)_\sigma \geq k$ *and*

2. $\left|\mathbb{E}[\mathsf{D}(\rho_{XY})] - \mathbb{E}[\mathsf{D}(\sigma_{XY})]\right| \leq \varepsilon$.

In addition, if $\mathrm{Tr}_X(\rho_{XY}) = \mathrm{Tr}_X(\sigma_{XY})$, we say X conditioned on Y has (standard) metric (min-)entropy *at least k (written "$\mathrm{H}^{t,\varepsilon}_{\textsf{metric-min}}(X|Y)_\rho \geq k$").*

Definition 5.3.6 (guessing pseudoentropy (cq-state))**.** *Let $\rho_{XY} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho^x_Y \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ be a cq-state. We say that X conditioned on Y has* (t, ε)-quantum guessing pseudoentropy *at least k (written "$\mathrm{H}^{t,\varepsilon}_{\textsf{guess}}(X|Y)_\rho \geq k$") if for every quantum circuit D of size t, $P^{\textsf{guess}}_{\mathsf{D}}(X|Y)_\rho \leq 2^{-k} + \varepsilon$.*

HILL entropy v.s. metric entropy By definition, the metric (min-)entropy of the quantum state is at least as large as its HILL (min-)entropy. In the classical case, it is known that metric entropy implies HILL entropy [BSW03]. We will show the analogous implication in the quantum setting in Section 5.3.4. As a useful intermediate step, we introduce the quantum min-max theorem in Section 5.3.3 first, which is also an essential tool for proving the Quantum Leakage Simulation Lemma (Theorem 5.5.1).

Guessing pseudoentropy v.s. HILL min-entropy As in the classical case, guessing pseudoentropy implies HILL entropy.

Proposition 5.3.7. *Let $\rho_{XY} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho^x_Y$ be a cq-state. If $\mathrm{H}^{t,\varepsilon}_{\textsf{HILL-min}}(X|Y)_\rho \geq k$ then $\mathrm{H}^{t-O(n),\varepsilon}_{\textsf{guess}}(X|Y)_\rho \geq k$.*

Proof. Suppose for contradiction, there exists a quantum circuit $\mathsf{A} : \mathrm{Dens}(\mathcal{Y}) \to \{0,1\}^n$ of size t such that $P^{\textsf{guess}}_{\mathsf{A}}(X|Y)_\rho > 2^{-k} + \varepsilon$. Define $A' : \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y}) \to \{0,1\}$ to be a quantum distinguisher $\mathsf{A}'(\rho_{XY}) = 1$ iff $\mathsf{A}(\rho_Y) = X$, then $\mathbb{E}[\mathsf{A}'(\rho_{XY})] \geq 2^{-k} + \varepsilon$. Also, A' can be implemented by a size $t + O(n)$ circuit.

For every σ_{XY} with $\mathrm{H}_{\mathsf{min}}(X|Y)_\sigma \geq k$, by Lemma 5.2.9, we know that $P^{\mathsf{guess}}_{\mathsf{A}}(X|Y)_\rho \leq 2^{-k}$, which implies $\mathbb{E}[\mathsf{A}'(\rho_{XY})] \leq 2^{-k}$. Therefore, σ_{XY} and ρ_{XY} are not $(t+O(n),\varepsilon)$-quantum-indistinguishable. That is, $\mathrm{H}^{t+O(n),\varepsilon}_{\mathsf{HILL\text{-}min}}(X|Y)_\rho < k$. □

Vadhan and Zheng showed that in the classical case, HILL entropy and guessing pseudoentropy are equivalent when n is logarithmic in the security parameter [VZ12]. Also, when $n = 1$, the equivalence between HILL entropy and guessing pseudoentropy implies Impagliazzo's Hardcore Theorem [Imp95] and *vice versa.* [Zhe13]

However, in the quantum case, we do not know whether these two definitions are equivalent. All the proofs suffer the same barrier discussed in Section 5.7.2. Briefly speaking, a proof cannot be extended to the quantum case if it relies on estimating the acceptance probability of a given quantum state. Therefore, connections between guessing pseudoentropy and other pseudoentropy notions remain as interesting open problems.

5.3.3 Quantum nonuniform min-max theorem

We begin with von Neumann's Min-Max Theorem for zero-sum game with two players. Let the strategy spaces of Player 1 and Player 2 be $\mathcal{A}$ and $\mathcal{B}$, respectively, and the payoff function be $g : \mathcal{A} \times \mathcal{B} \to [-1, 1]$. The theorem says that if for every mixed strategy $A \in \mathrm{Conv}(\mathcal{A})$, Player 2 can respond $b \in \mathcal{B}$ so that the expected payoff $\mathbb{E}_{a\leftarrow A}[g(a, b)]$ is at least p, then Player 2 has an universal mixed strategy $B \in \mathrm{Conv}(\mathcal{B})$ that guarantees the same payoff regardless of the strategy of Player 1. Namely, for all $a \in \mathcal{A}, \mathbb{E}_{b\leftarrow B}[g(a, b)] \geq p$. In many applications in cryptography and complexity theory, (e.g., [Imp95, RTTV08, DP08, GW11, VZ12], the strategy space $\mathcal{A}$ is taken to be a convex set of distributions over $\{0, 1\}^n$. Also, those applications require not only the existence of a universal mixed strategy B, but also with low complexity (measured in support size). In such settings, the theorem is called *Nonuniform Min-max Theorem* [Zhe13] (contrary to the Uniform Min-Max Theorem where it further requires an explicit construction of the universal mixed strategy B.). In this section, we generalize the classical Nonuniform Min-max Theorem to the quantum setting where the strategy space $\mathcal{A}$ becomes a set of quantum states.

Theorem 5.3.8 (Quantum Non-uniform Min-Max Theorem)**.** *Consider a zero-sum game between two players where the strategy space of Player 1 is a convex set $\mathcal{A} \subseteq \mathrm{Dens}(\mathbb{C}^d)$ and the strategy space of Player 2 is $\mathcal{B}$. For strategies $a \in \mathcal{A}$ and $b \in \mathcal{B}$, the payoff to Player 2 is $g(a,b) = \langle a\,,\, M(b)\rangle$ where $M : \mathcal{B} \to \mathrm{Meas}(\mathbb{C}^d)$. Suppose that for every strategy $a \in \mathcal{A}$ of Player 1, there exists a pure strategy $b \in \mathcal{B}$ such that $g(a,b) \geq p$. Then for every $\varepsilon \in (0,1/2)$, there exists a mixed strategy $\widehat{B}$ of Player 2 such that for every strategy $a \in \mathcal{A}$ of Player 1, $\mathbb{E}_{b\leftarrow \hat{B}}[g(a,b)] \geq p - \varepsilon$. Moreover, $\widehat{B}$ is the uniform distribution over a multi-set $\mathcal{S}$ consisting of at most*

$$O\left(\frac{\log d - \min_{a\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(a)}{\varepsilon^2}\right)$$

strategies in $\mathcal{B}$.

Note that if we restrict Player 1's strategies to be diagonal and set $d = 2^n$, then the above theorem replicates the classical Non-uniform Min-max Theorem.

Proof. By von Newmann's Min-max Theorem, there exists a distribution B on $\mathcal{B}$ such that for all $a \in \mathcal{A}$, $\mathbb{E}_{b\leftarrow B}\Big[g(a,b)\Big] \geq p$. Therefore, it suffices to show that there exists a "small" multi-set $\mathcal{S} = \{b_1, \ldots, b_T\}$ such that for all $a \in \mathcal{A}$,

$$\left|\mathop{\mathbb{E}}_{b\leftarrow B}[g(a,b)] - \mathop{\mathbb{E}}_{b\leftarrow \widehat{B}}[g(a,b)]\right| = \left|\mathop{\mathbb{E}}_{b\leftarrow B}[g(a,b)] - \frac{1}{T}\sum_{i=1}^{T} g(a,b_i)\right| \leq \varepsilon\,.$$

Observed by Skórski [Sko17], the above statement can be obtained from a generalization error bounds in statistical learning theory. We particularly use the bound for Rademacher complexity.

Recall the definitions of Rademacher complexity and the generalization error bound for it:

Definition 5.3.9 (Rademacher complexity ([BM02]))**.** *Let $\mathcal{F}$ be a class of functions from $\mathcal{W} \to \mathbb{R}$, and $(w_1, \ldots, w_T) \in \mathcal{W}^T$. The empirical Rademacher complexity of $\mathcal{F}$ is defined as*

$$\widehat{\mathfrak{R}}_{w_1,\ldots,w_T}(\mathcal{F}) \stackrel{\text{def}}{=} \mathbb{E}\left[\sup_{f\in\mathcal{F}} \frac{1}{T}\sum_{i=1}^{T} \gamma_i f(w_i)\right],$$

where the expectation is over γ_i's, which are sampled from Rademacher distribution (uniformly *over* $\{1, -1\}$*) independently.*

Let W *be a distribution on* $\mathcal{W}$*. Then the Rademacher complexity of* $\mathcal{F}$ *is defined as*

$$\mathfrak{R}_{W,T}\left(\mathcal{F}\right) \stackrel{\text{def}}{=} \mathbb{E}\Big[\widehat{\mathfrak{R}}_{w_1,\ldots,w_T}\left(\mathcal{F}\right)\Big]\,,$$

where the expectation is over w_i*'s sampled from* W *independently, and independent to* γ_i*'s.*

Theorem 5.3.10 (generalization bounds via Rademacher complexity [BM02])**.** *Let* $\mathcal{F}$ *be a class of functions from* $\mathcal{W} \to [0,1]$ *and* W *be a distribution on* $\mathcal{W}$*. If* $w_1, \ldots, w_T$ *are drew i.i.d. from* W*, then for every* $\delta \in (0,1)$*,*

$$\Pr_{w_1,\ldots,w_T}\left[\forall f \in \mathcal{F}\,,\ \left|\mathop{\mathbb{E}}_{w\sim W}[f(w)] - \frac{1}{T}\sum_{i=1}^{T} f(w_i)\right| \le \mathfrak{R}_{W,T}\left(\mathcal{F}\right) + O\!\left(\sqrt{\frac{\log(1/\delta)}{T}}\right)\right] > 1-\delta\,.$$

In particular, there exists $w_1, \ldots, w_T \in \mathcal{W}$ *such that*

$$\forall f \in \mathcal{F}\,,\ \left|\mathop{\mathbb{E}}_{w\sim W}[f(w)] - \frac{1}{T}\sum_{i=1}^{T} f(w_i)\right| \le \mathfrak{R}_{W,T}\left(\mathcal{F}\right) + O\!\left(\sqrt{1/T}\right).$$

Now we bound the Rademacher complexity of $\mathcal{F}$ by the following theorem:

Theorem 5.3.11. *For* $d \in \mathbb{N}$*, let* $\mathcal{W} = \mathrm{Meas}(\mathbb{C}^d)$*,* $\mathcal{A} \subseteq \mathrm{Dens}(\mathbb{C}^d)$*, and* $\mathcal{F} = \{\langle \cdot\,,\, \rho\rangle : \rho \in \mathcal{A}\}$*. For every distribution* W *on* $\mathcal{W}$ *and* $T \ge \log d - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho)$*,*

$$\mathfrak{R}_{W,T}\left(\mathcal{F}\right) = 2 \cdot \sqrt{\frac{\log d - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho)}{T}}\,.$$

Taking $W = M(B), T = O\big((\log d - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho))/\varepsilon^2\big), \delta = 0.5$, and by Theorem 5.3.11 and Theorem 5.3.10, we have

$$\begin{aligned}&\Pr_{w_1,\ldots,w_T}\left[\forall f \in \mathcal{F}\,,\ \left|\mathop{\mathbb{E}}_{w\sim W}[f(w)] - \frac{1}{T}\sum_{i=1}^{T} f(w_i)\right| \le \varepsilon\right]\\ =&\Pr_{b_1,\ldots,b_T}\left[\forall a \in \mathcal{A}\,,\ \left|\mathop{\mathbb{E}}_{b\sim B}\Big[\langle a\,,\, M(b)\rangle\Big] - \frac{1}{T}\sum_{i=1}^{T}\langle a\,,\, M(b_i)\rangle\right| \le \varepsilon\right] > 0.5,\end{aligned}$$

which implies there exists $\{b_1, \ldots, b_T\}$ such that for all $a \in \mathcal{A}$

$$\left|\mathop{\mathbb{E}}_{b\leftarrow B}[g(a,b)] - \frac{1}{T}\sum_{i=1}^{T} g(a,b_i)\right| \le \varepsilon.$$

□

Proof of Theorem 5.3.11. It suffices to bound the empirical Rademacher complexity for every $w_1, \ldots, w_T \in \mathcal{W}$:

$$\widehat{\mathfrak{R}}_{w_1,\ldots,w_T}(\mathcal{F}) = \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sup_{f\in\mathcal{F}} \frac{1}{T}\sum_{j=1}^{T} \gamma_j f(w_j)\right] = \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sup_{\rho\in\mathcal{A}} \operatorname{Tr}(\Pi\rho)\right],$$

where $\Pi = \frac{1}{T}\sum_{j=1}^{T}\gamma_j w_j$. Note that Π depends on random variables γ_j's.

By the non-negativity of KL-divergence, we have for every $t > 0$,

$$\begin{aligned}
&\mathrm{D}_{\mathsf{KL}}\left(\rho \,\middle\|\, \frac{\exp(t\Pi)}{\operatorname{Tr}(\exp(t\Pi))}\right) \geq 0 \\
\Rightarrow\ &\operatorname{Tr}\left(\rho \log\left(\frac{\exp(t\Pi)}{\operatorname{Tr}(\exp(t\Pi))}\right)\right) \leq \operatorname{Tr}\left(\rho\log\rho\right) \\
\Rightarrow\ & t\cdot \operatorname{Tr}(\Pi\rho) \leq \log(\operatorname{Tr}(\exp(t\Pi))) + \operatorname{Tr}(\rho\log\rho). \qquad (5.2)
\end{aligned}$$

By the Inequality (5.2), we have that for every $t > 0$,

$$\begin{aligned}
\widehat{\mathfrak{R}}_{w_1,\ldots,w_T}(\mathcal{F}) &\leq \frac{1}{t}\mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sup_{\rho\in\mathcal{A}}\left\{\log\left(\operatorname{Tr}(\exp(t\Pi))\right) + \operatorname{Tr}(\rho\log\rho)\right\}\right] \\
&= \frac{1}{t}\left(\mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\log\left(\operatorname{Tr}(\exp(t\Pi))\right)\right] - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho)\right) \\
&\leq \frac{1}{t}\left(\log\left(\mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\operatorname{Tr}(\exp(t\Pi))\right]\right) - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho)\right), \qquad (5.3)
\end{aligned}$$

where the last inequality is by Jensen inequality. Now we bound the term $\mathbb{E}_{\gamma_1,\ldots,\gamma_T}\left[\operatorname{Tr}(\exp(t\Pi))\right]$ by Golden Thompson inequality:

$$\begin{aligned}
\mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\operatorname{Tr}(\exp(t\Pi))\right] &\leq \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\operatorname{Tr}\left(\exp\left(\frac{t}{T}\gamma_1 w_1\right)\cdots\exp\left(\frac{t}{T}\gamma_T w_T\right)\right)\right] \\
&= \operatorname{Tr}\left(\mathop{\mathbb{E}}_{\gamma_1}\left[\exp\left(\frac{t}{T}\gamma_1 w_1\right)\right]\cdots\mathop{\mathbb{E}}_{\gamma_T}\left[\exp\left(\frac{t}{T}\gamma_T w_T\right)\right]\right) \qquad (5.4)
\end{aligned}$$

Then by Taylor expansion, we have for all $i \in [T]$

$$\mathop{\mathbb{E}}_{\gamma_i}\left[\exp\left(\frac{t}{T}\gamma_i w_i\right)\right] = \mathop{\mathbb{E}}_{\gamma_i}\left[\sum_{j=0}^{\infty}\left(\frac{t}{T}\gamma_i w_j\right)^j\right] = \sum_{j=0}^{\infty}\left(\frac{t}{T}w_j\right)^{2j} \leq \mathbb{1}_d \cdot \sum_{j=0}^{\infty}\left(\frac{t}{T}\right)^{2j} \qquad (5.5)$$

By the fact that $B > C$ implies $\operatorname{Tr}(AB) > \operatorname{Tr}(AC)$ when A, B, C are positive definite matrices, we can plug in Inequality (5.5) to Equation (5.4) to get

$$\begin{aligned}
\mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\Big[\mathrm{Tr}\big(\exp(t\Pi)\big)\Big] &\leq \mathrm{Tr}\left(\left(\mathbb{1}_d \sum_{j=0}^{\infty}\Big(\frac{t}{T}\Big)^{2j}\right)^T\right) \\
&= d \cdot \left(\frac{\exp(t/T)+\exp(-t/T)}{2}\right)^T \\
&\leq d \cdot \Big(1 + t^2/T^2\Big)^T \qquad (5.6) \\
&\leq d \cdot \Big(\exp(t^2/T^2)\Big)^T = d \cdot \exp\Big(t^2/T\Big) \qquad (5.7)
\end{aligned}$$

for $t/T \in [0,1]$. Finally, we put Inequality (5.7) back to Inequality (5.3), then have

$$\mathfrak{R}_{w_1,\ldots,w_T}(\mathcal{F}) \leq \frac{1}{t}\left(\log d - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho) + t^2/T\right).$$

Take $t = \sqrt{(\log d - \min_{\rho\in\mathcal{A}} \mathrm{H}_{\mathsf{Sh}}(\rho))/T}$, we concludes the proof. □

5.3.4 Metric entropy implies HILL entropy

In the classical case, it is known that the HILL and metric pseudoentropies are interchangeable up to some degradation in the size of distinguishers [BSW03]. With the equivalence, metric entropy is a useful intermediate notion to obtain tighter security proof in a number of cases (e.g., [DP08, FOR15]). Here we will show the equivalence in the quantum setting.

Theorem 5.3.12 ((relaxed-)HILL ⇔ (relaxed-)metric)**.** *Let ρ_{XY} be a bipartite quantum system in $\mathrm{Dens}(\mathcal{X}\otimes\mathcal{Y})$ and $\dim(\mathcal{X}\otimes\mathcal{Y}) = d$. If $\mathrm{H}^{t,\varepsilon}_{\mathsf{metric\text{-}min}}(X|Y)_\rho \geq k$ (resp., $\mathrm{H}^{t,\varepsilon}_{\mathsf{r\text{-}metric\text{-}min}}(X|Y)_\rho \geq k$), then for every $\delta > 0$, we have $\mathrm{H}^{t',\varepsilon'}_{\mathsf{HILL\text{-}min}}(X|Y)_\rho \geq k$ (resp., $\mathrm{H}^{t',\varepsilon'}_{\mathsf{r\text{-}HILL\text{-}min}}(X|Y)_\rho \geq k$), where $\varepsilon' = \varepsilon + \delta$ and $t' = t/O\left((\log d - k)/\delta^2\right)$.*

Proof. Consider the following zero-sum game between Player 1 and Player 2:

- The strategy space of Player 1 $\mathcal{A} = \{\sigma_{XY} \in \mathrm{Dens}(\mathcal{X}\otimes\mathcal{Y}) : \mathrm{H}_{\mathsf{min}}(X|Y)_\sigma \geq k, \sigma_Y = \rho_Y\}$.
- The strategy space of Player 2 $\mathcal{B}$ is a set of all quantum distinguishers $\mathsf{D} : \mathrm{Dens}(\mathcal{X}\otimes\mathcal{Y}) \to \{0,1\}$ of size at most t'.
- For the payoff function $g : \mathcal{A}\times\mathcal{B} \to [0,1]$, we first define the auxiliary mapping M. For an input distinguisher $\mathsf{D} \in \mathcal{B}$, let Π_D be the corresponding measurement operator, and

define

$$M(\mathsf{D}) = \frac{1}{2}\Big((\mathbb{E}[\mathsf{D}(\rho_{XY})] + 1) \cdot \mathbb{1}_d - \Pi_{\mathsf{D}}\Big).$$

Then for $\sigma_{XY} \in \mathcal{A}$ and $\mathsf{D} \in \mathcal{B}$,

$$g(\sigma_{XY}, \mathsf{D}) = \Big\langle \sigma_{XY}\,,\, M(\mathsf{D}) \Big\rangle = \frac{1}{2}\Big(\mathbb{E}\big[\mathsf{D}(\rho_{XY})\big] + 1 - \mathbb{E}\big[\mathsf{D}(\sigma_{XY})\big]\Big).$$

Note that the strategy space $\mathcal{A}$ is convex. Also, since $0 \le \Pi_{\mathsf{D}} \le \mathbb{1}_d$, we have $0 \le M(\mathsf{D}) \le \mathbb{1}_d$, and so $M(\mathsf{D}) \in \mathrm{Meas}(\mathcal{X} \otimes \mathcal{Y})$. Therefore, the above game satisfies the requirements in Theorem 5.3.8.

Now suppose for contradiction, let $\mathrm{H}^{t',\varepsilon'}_{\mathsf{HILL\text{-}min}}(X|Y)_\rho < k$ Then for all $\sigma_{XY} \in \mathcal{A}$, there exists a quantum distinguisher $\mathsf{D} : \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y}) \to \{0,1\}$ of size t' such that

$$\mathbb{E}\Big[\mathsf{D}(\rho_{XY})\Big] - \mathbb{E}\Big[\mathsf{D}(\sigma_{XY})\Big] > \varepsilon',$$

namely, $g(\sigma_{XY}, \mathsf{D}) > (1+\varepsilon')/2$. By Theorem 5.3.8, there exists a quantum circuit $\widehat{\mathsf{D}}$ of size $t' \cdot O\big((\log d - k)/\delta^2\big)$ such that for all σ_{XY} with $\mathrm{H}_{\mathsf{min}}(X|Y)_\sigma \ge k$,

$$g(\sigma_{XY}, \widehat{\mathsf{D}}) > (1+\varepsilon')/2 - \delta/2\,.$$

That is,

$$\mathbb{E}\Big[\widehat{\mathsf{D}}(\rho_{XY})\Big] - \mathbb{E}\Big[\widehat{\mathsf{D}}(\sigma_{XY})\Big] > \varepsilon' - \delta = \varepsilon,$$

which contradicts the assumption.

The proof for the case of $\mathrm{H}_{\mathsf{r\text{-}HILL\text{-}min}}$ is identical except the requirement of $\sigma_Y = \rho_Y$ in $\mathcal{A}$ is removed. □

Remark 5.3.13. *In the above discussion, we define the computational entropies and state the theorems only respect to quantum distinguishers with classical advice. One can also consider HILL/metric entropy respect to quantum distinguishers with quantum advice. The transformation between metric and HILL entropy can be extended to this model. Indeed, in the proof of Theorem 5.3.12, we only use the fact that distinguishers (strategies of Player 2)* *can be given as measurement operator and that taking a distribution over a small number of such operators incurs a small blow-up in circuit size.*

5.4 Computational Quantum Max-divergence

In this section, we consider computational analogues of max-divergence, a.k.a max-relative entropy in the quantum setting. Recall that the max-divergence is a generalization of min-entropy. That is, the max-divergence between a quantum state $\rho \in \mathrm{Dens}(\mathbb{C}^{2^n})$ and a 2^n-dimensional maximally mixed state $\rho_{2^n}^{\mathsf{mm}}$ is exactly n minus the min-entropy of ρ. Similar to min-entropy, we can also consider computational relaxations of max-divergence. Since max-divergence involves between *two* states, there are more ways to define its computational relaxations.

Classically, relations between some computational notions of relative min-entropies are given by the Dense Model Theorem [RTTV08]. In Section 5.4.2, we review the theorem and prove a variation that establishes more connections among the various notions. For the quantum case, we show in Section 5.4.3 that some computational notions are not equivalent, which can be interpreted as saying that a "Quantum Dense Model Theorem" does not hold.

5.4.1 Definition

Following the idea of defining HILL-type entropy, there are already two ways to relax max-divergence (Definition 5.2.3) to computational notions. First, we can say ρ has small computational max-divergence with respect to σ if there exists ρ' that is indistinguishable from ρ, but has small max-divergence with respect to σ entropy. Alternatively, we can ask that there exists σ' indistinguishable from σ such that ρ has small max-divergence with respect to σ'.

Definition 5.4.1 (HILL-1 max-divergence)**.** *Let ρ and σ be density operators of the same system. We say $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ if there exists ρ' that is (t, ε)-quantum-indistinguishable from ρ and $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda$.*

Definition 5.4.2 (HILL-2 max-divergence)**.** *Let ρ and σ be density operators of the same system. We say $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ if there exists σ' that is (t, ε)-quantum-indistinguishable from σ and $\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma') \leq \lambda$.*

By switching the quantifiers, we can also have two metric-type generalizations:

Definition 5.4.3 (metric-1 max-divergence)**.** *Let ρ and σ be density operators of the same system. We say $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ if for all t-size quantum distinguishers A, there exists ρ' such that (i) $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda$, and (ii) $|\mathbb{E}[\mathsf{A}(\rho)] - \mathbb{E}[\mathsf{A}(\rho')]| < \varepsilon$.*

Definition 5.4.4 (metric-2 max-divergence)**.** *Let ρ and σ be density operators of the same system. We say $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ if for all t-size quantum distinguishers A, there exists σ' such that (i) $\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma') \leq \lambda$, and (ii) $|\mathbb{E}[\mathsf{A}(\sigma)] - \mathbb{E}[\mathsf{A}(\sigma')]| < \varepsilon$.*

Another approach is to directly compare the behavior of distinguisher on the states ρ and σ, by restricting the distinguishers in Proposition 5.2.4 to be small quantum circuits:

Definition 5.4.5 (pseudo max-divergence)**.** *Let ρ and σ be density operators of the same system. Then $\mathrm{D}^{t,\varepsilon}_{\mathsf{pseudo\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ if for all t-size quantum distinguishers A, we have $\Pr[\mathsf{A}(\rho) = 1] \leq 2^{\lambda} \cdot \Pr[\mathsf{A}(\sigma) = 1] + \varepsilon$.*

Remark 5.4.6. *When we restrict $\rho, \rho', \sigma, \sigma'$ to be classical discrete random variables, and distinguishers to be classical, we get the definitions of computational relative min-entropy notions in the classical case.*

Taking σ to be the maximally mixed states in $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma)$ and $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma)$, recovers our computational analogues of min-entropy.

Proposition 5.4.7. *For $s \in \mathbb{N}$, $\varepsilon > 0$, $\rho \in \mathrm{Dens}(\mathbb{C}^d)$, and $k \in [\log d]$, we have*

1. *$\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \rho^{\mathsf{mm}}_d) \leq \log d - k$ if and only if $\mathrm{H}^{t,\varepsilon}_{\mathsf{HILL\text{-}min}}(\rho) \geq k$.*

2. *$\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \rho^{\mathsf{mm}}_d) \leq \log d - k$ if and only if $\mathrm{H}^{t,\varepsilon}_{\mathsf{metric\text{-}min}}(\rho) \geq k$.*

We also have the following relations:

Proposition 5.4.8. *For $s \in \mathbb{N}$, $\varepsilon, \lambda > 0, \rho \in \mathrm{Dens}(\mathbb{C}^d)$, we have*

1. $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda \Rightarrow \mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda \Rightarrow \mathrm{D}^{t,\varepsilon}_{\mathsf{pseudo\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$.

2. $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda \Rightarrow \mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda \Rightarrow \mathrm{D}^{t,\varepsilon'}_{\mathsf{pseudo\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$, *where* $\varepsilon' = 2^\lambda \cdot \varepsilon$.

Proof. Suppose $\mathrm{D}^{t,\varepsilon}_{\mathsf{pseudo\text{-}max}}(\rho \,\|\, \sigma) > \lambda$. Let A be such that

$$\Pr[\mathsf{A}(\rho) = 1] \;>\; 2^\lambda \cdot \Pr[\mathsf{A}(\sigma) = 1] + \varepsilon.$$

Then for all ρ' with $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda$,

$$\Pr[\mathsf{A}(\rho) = 1] \;>\; 2^\lambda \cdot \Pr[\mathsf{A}(\sigma) = 1] + \varepsilon \;\geq\; \Pr[\mathsf{A}(\rho') = 1] + \varepsilon,$$

which implies $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \geq \lambda$. On the other hand, for all σ' with $\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma') \leq \lambda$,

$$\Pr[\mathsf{A}(\sigma') = 1] \;\geq\; \frac{1}{2^\lambda} \cdot \Pr[\mathsf{A}(\rho) = 1] \;\geq\; \Pr[\mathsf{A}(\sigma) = 1] + \frac{\varepsilon}{2^\lambda},$$

which implies $\mathrm{D}^{t,\varepsilon/2^\lambda}_{\mathsf{metric\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \geq \lambda$. □

Similarly to Theorem 5.3.12, the HILL-type and metric-type relative min-entropies are also interchangeable up to a small parameter loss.

Theorem 5.4.9. *Let σ and ρ be quantum states in* $\mathrm{Dens}(\mathcal{X})$ *where* $\dim(\mathcal{X}) = d$. *If* $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ *(resp.,* $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$*), then* $\mathrm{D}^{t',\varepsilon'}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$ *(resp.,* $\mathrm{D}^{t',\varepsilon'}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \lambda$*), where $\varepsilon' = 2\varepsilon$ and $t' = t \cdot O(\varepsilon^2/\log d)$.*

Proof. Suppose for contradiction that $\mathrm{D}^{t',\varepsilon'}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) > \lambda$. That is for all ρ' with $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda$, there exists a distinguisher A of size t' such that $\mathbb{E}[\mathsf{A}(\rho)] - \mathbb{E}[\mathsf{A}(\rho')] > \varepsilon'$. We consider the following zero-sum game:

- The strategy space of Player 1 $\mathcal{A} = \{\rho' \in \mathrm{Dens}(\mathcal{X}) : \mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \leq \lambda\}$.

- The strategy space of Player 2 $\mathcal{B}$ a set of all distinguishers $\mathsf{A} : \mathrm{Dens}(\mathcal{X}) \to \{0, 1\}$ of size at most t'.

- For the payoff function $g : \mathcal{A} \times \mathcal{B} \to \{0,1\}$, we first define the auxiliary mapping M. For an input distinguisher $\mathsf{A} \in \mathcal{B}$, let Π_D be its corresponding measurement operator, and define
$$M(\mathsf{A}) = \frac{1}{2}\Big((\mathbb{E}[\mathsf{A}(\rho)] + 1) \cdot \mathbb{1}_d - \Pi_{\mathsf{A}} \Big).$$
Then for $\rho' \in \mathcal{A}$ and $\mathsf{A} \in \mathcal{B}$,
$$g(\rho', D) = \Big\langle \rho' \,,\, M(\mathsf{A}) \Big\rangle = \frac{1}{2}\Big(\mathbb{E}[\mathsf{A}(\rho)] + 1 - \mathbb{E}[\mathsf{A}(\rho')] \Big).$$

Note that the strategy space $\mathcal{A}$ is convex. Also, since $0 \le \Pi_{\mathsf{A}} \le \mathbb{1}_d$, we have $0 \le M(\mathsf{A}) \le \mathbb{1}_d$, and so $M(\mathsf{A}) \in \mathrm{Meas}(\mathcal{X})$. Therefore, the above game satisfies the requirements in Theorem 5.3.8.

By the nonuniform Quantum Min-max Theorem (Theorem 5.3.8), there exists a universal distinguisher $\widehat{\mathsf{A}}$ of size $t = t' \cdot O(\log d)/\varepsilon^2$ such that for all ρ' with $\mathrm{D}_{\mathsf{max}}(\rho' \,\|\, \sigma) \le \lambda$,
$$\mathbb{E}\Big[\widehat{\mathsf{A}}(\rho')\Big] - \mathbb{E}\Big[\widehat{\mathsf{A}}(\rho)\Big] > \varepsilon' - \varepsilon = \varepsilon.$$

By the definition of metric relative entropy, we get $\mathrm{D}^{t,\varepsilon}_{\mathsf{metric\text{-}1\text{-}max}}(\rho \,\|\, \sigma) > \lambda$, which yields a contradiction.

Similarly for the type-2 notions, the strategy space of Player 1 becomes a convex set $\mathcal{A} = \{\sigma' = \mathrm{Dens}(\mathcal{X}) : \mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma') \le \lambda\}$, $\mathcal{B}$ remain the same, and for the payoff function, we replace ρ, ρ' by σ, σ', respectively. The conclusion for type-2 notions follows the same argument. □

Because of this equivalence, in the rest of the section, we focus on the HILL-type and pseudo notions.

5.4.2 Classical Dense Model Theorem

In the classical case, relations between HILL-1, HILL-2, and pseudo max-divergence are given captured by the Dense Model Theorem [RTTV08, GT08] and variants. Specifically, the form of the Dense Model Theorem by Reingold, Trevisan, Tulsiani, and Vadhan [RTTV08] says

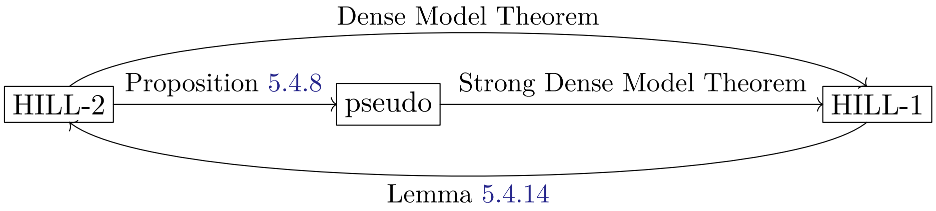


Figure 5.4.1: *Relationships between computational relative min-entropies in the classical setting*

that HILL-2 max-divergence implies HILL-1 when the divergence is $\lambda = O(\log \kappa)$ where κ is the security parameter. The *Strong Dense Model Theorem* [MPRV09] says that pseudo max-divergence implies HILL-1 max-divergence. Here we additionally show that HILL-1 max-divergence also implies HILL-2 max-divergence (Lemma 5.4.14). Therefore, all three notions are equivalent in the classical setting. (See Figure 5.4.1 for their relationships)

Definition 5.4.10 (density (classical))**.** *Let X and Y be distributions over $\mathcal{X}$. For $0 < \delta < 1$, we say X is δ*-dense *in Y if*

$$\forall x \in \mathcal{X}\ ,\ \Pr[X = x] \leq \frac{1}{\delta} \cdot \Pr[Y = x].$$

Equivalently, $\mathrm{D}_{\mathsf{max}}(X \,\|\, Y) \leq \log(1/\delta)$*.*

Definition 5.4.11 (pseudo-density (classical))**.** *Let X and Y be distributions over $\mathcal{X}$. For $0 < \delta < 1$, we say X is* $(\delta, (t, \varepsilon))$-pseudo-dense *in Y* $\mathrm{D}^{t,\varepsilon}_{\mathsf{pseudo\text{-}max}}(X \,\|\, Y) \leq \log(1/\delta)$*.*

The statement of the Strong Dense Model Theorem is as follows.

Theorem 5.4.12 (Strong Dense Model Theorem [MPRV09])**.** *For every $t, n \in \mathbb{N}$ and $0 < \varepsilon, \delta < 1$, let X, Y be distributions over $\mathcal{X}$ such that X is $(\delta, (t, \varepsilon))$-pseudo-dense in Y. Then there exists a distribution X' over $\mathcal{X}$ such that X' is δ-dense in Y, and X' is (t', ε')-indistinguishable from X where $t' = t/\operatorname{poly}(1/\varepsilon, \log(1/\delta))$ and $\varepsilon' = O(\varepsilon/\delta)$.*

Corollary 5.4.13. *For any $t, n \in \mathbb{N}$, $0 < \varepsilon < 1$ and $\lambda > 0$, let X, Y be two distributions over $\mathcal{X}$ such that $\mathrm{D}^{t,\varepsilon}_{\mathsf{pseudo\text{-}max}}(X \,\|\, Y) \leq \lambda$, then $\mathrm{D}^{t',\varepsilon'}_{\mathsf{HILL\text{-}1\text{-}max}}(X \,\|\, Y) \leq \lambda$ where $t' = t/\operatorname{poly}(1/\varepsilon, \lambda)$ and $\varepsilon' = O(\varepsilon \cdot 2^{\lambda})$.*

Note that the dependency on λ in ε' is exponential. Therefore we usually limit $\lambda = \log(\kappa)$ where κ is the security parameter to maintain the negligibility of ε'.

Here we observe that that HILL-1 max-divergence also implies HILL-2 max-divergence without any parameter loss, which is also true in the quantum case:

Lemma 5.4.14. *Let $\rho, \rho', \sigma \in \mathrm{Dens}(\mathcal{X})$ such that ρ' is δ-dense in σ, and ρ' is (t, ε)-quantum-indistinguishable from ρ. That is, $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) < \log(1/\delta)$. Then there exists $\sigma' \in \mathrm{Dens}(\mathcal{X})$ such that ρ is δ-dense in σ', and σ' is (t, ε)-quantum-indistinguishable from σ. That is $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) < \log(1/\delta)$.*

Proof. Define a state $\tau = (\sigma - \delta \cdot \rho')/(1 - \delta)$. Since ρ' is δ-dense in σ, $\tau > 0$, and so $\tau \in \mathrm{Dens}(\mathcal{X})$. Let $\sigma' = \delta \cdot \rho + (1 - \delta) \cdot \tau \in \mathrm{Dens}(\mathcal{X})$. Clearly $\rho \leq \frac{1}{\delta}\sigma'$. Also, σ and σ' are (t, ε)-quantum-indistinguishable due to the quantum indistinguishability between ρ and ρ'. □

Therefore, by Theorem 5.4.12 (Strong Dense Model Theorem), Lemma 5.4.14, and Proposition 5.4.8, all the three notions, pseudo, HILL-1 and HILL-2 max-divergence are equivalent up to some parameter losses in the classical case.

5.4.3 Impossibility of Quantum Dense Model Theorem

In this section, we will show a separation between the $\mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}$ and $\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}$ max-divergence for quantum states. More specifically, we show that there exist quantum states ρ and σ such that $\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq 1$ but $\mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma)$ is unbounded. To this end, we use the language of density. We first generalize the notion of density for quantum states:

Definition 5.4.15 (density (quantum))**.** *Let ρ and σ be quantum states on $\mathrm{Dens}(\mathcal{X})$. For $0 < \delta \leq 1$, we say ρ is* δ-dense *in σ if $\rho \leq \frac{1}{\delta}\sigma$. Equivalently, $\mathrm{D}_{\mathsf{max}}(\rho \,\|\, \sigma) \leq \log(1/\delta)$.*

Recall the Dense Model Theorem statement and what the counterexample should achieve to show the non-existence of Quantum Dense Model Theorem. Suppose σ and σ' are two computationally indistinguishable quantum states and ρ is a quantum state that is δ-dense in σ'. That is $\mathrm{D}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \leq \log(1/\delta)$. A Quantum Dense Model Theorem would imply that there exists ρ' that is δ-dense in σ and indistinguishable from ρ. That is $\mathrm{D}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) \leq \log(1/\delta)$.

However, we show that this is false by constructing ρ, σ, and σ' such that for every ρ' that is δ-dense in σ, it can be distinguished from ρ. That is $\mathrm{D}_{\textsf{HILL-1-max}}(\rho \,\|\, \sigma) = \infty$.

Our counterexample is based on the following two observations: 1) the only state that is dense in a pure state is the pure state itself; 2) there exists a pure state that is pseudorandom. We state the observations formally as follows.

Lemma 5.4.16. *Let $\mathcal{X} \otimes \mathcal{Y}$ be a bipartite quantum state space. Suppose $\sigma = |x\rangle\langle x| \otimes \sigma_Y \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ where $|x\rangle \in \mathrm{Ball}(\mathcal{X})$ and $\sigma_Y \in \mathrm{Dens}(\mathcal{Y})$. For every $0 < \delta \leq 1$, a density operator ρ that is δ-dense in σ must be of the form $|x\rangle\langle x| \otimes \rho_Y$, where ρ_Y is δ-dense in σ_Y.*

Proof. Let $\sigma_Y = \sum_i p_i |y_i\rangle\langle y_i|$ be the spectral decomposition of σ_Y. Then $\sigma = |x\rangle\langle x| \otimes \sigma_Y = \sum_i p_i |x, y_i\rangle\langle x, y_i|$. Suppose for contradiction, the spectral decomposition of $\rho \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ is $\sum_i q_i |\psi_i\rangle\langle\psi_i|$ but for some j, $q_j > 0$ and $\mathrm{Tr}_Y(|\psi_j\rangle\langle\psi_j|) \neq |x\rangle\langle x|$.

Let $|v\rangle = |\psi_j\rangle - \sum_i \langle x, y_i|\psi_j\rangle \cdot |x, y_i\rangle$ Then $\langle v|x, y_i\rangle = 0$ for all i, and since $\mathrm{Tr}_Y(|\psi_j\rangle\langle\psi_j|) \neq |x\rangle\langle x|$, $|v\rangle$ is non-zero and is not orthogonal to ψ_j. Let $|\phi\rangle = |v\rangle / \||v\rangle\| \in \mathrm{Ball}(\mathcal{X} \otimes \mathcal{Y})$. Then $\langle\phi|\rho|\phi\rangle \geq \|\langle\phi|\psi_j\rangle\|^2 > 0$, but $\langle\phi|\sigma|\phi\rangle = \sum_i \langle\phi| \left(|x, y_i\rangle\langle x, y_i|\right) |\phi\rangle = 0$, which contradicts the assumption that $\rho \leq \frac{1}{\delta}\sigma$ for some $\delta > 0$. □

Theorem 5.4.17 ([BMW09, GFE09])**.** *There is a constant $c > 0$ such that for all $t, m \in \mathbb{N}$, $\varepsilon > 0$ such that $m \geq c \cdot \log(t/\varepsilon)$, there exists a pure state $\rho = |\psi\rangle\langle\psi| \in \mathrm{Dens}(\mathbb{C}^{2^m})$ on m qubits that is (t, ε)-quantum-pseudorandom.*

Remark 5.4.18. *In [BMW09, GFE09], they showed that a uniformly random pure state $\rho = |\psi\rangle\langle\psi| \in \mathrm{Dens}(\mathbb{C}^{2^m})$ is (t, ε)-quantum-indistinguishable with all but $2^{-\Omega(2^m)}$ probability. We can show that sampling a pure state $|\psi\rangle$ uniformly at random from $\left\{\sum_{i \in \{0,1\}^m} \alpha_i |i\rangle : \alpha_i \in \{\pm 2^{-m/2}\}\right\}$ is (t, ε)-quantum-indistinguishable with all but $2^{-\Omega(2^m)}$ probability. See Appendix 5.7.1 for the formal statement and proof.*

The following theorem says that a Quantum Dense Model Theorem does not exist.

Theorem 5.4.19. *For $t, n \in \mathbb{N}, \varepsilon, \delta \in (0, 1)$, and integers $m_1, m_2 > O(\log(t/\varepsilon))$ with $m_1 + m_2 = n$. Let $\mathcal{X} = \mathbb{C}^{2^{m_1}}, \mathcal{Y} = \mathbb{C}^{2^{m_2}}$. There exist quantum states ρ, σ, $\sigma' \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ with $\mathrm{H}_{\textsf{min}}(\sigma) = m_2, \mathrm{H}_{\textsf{min}}(\sigma') = m_1$ such that*

1. ρ is δ-dense in σ'.

2. σ' and σ are (t, ε)-quantum-indistinguishable.

3. ρ and ρ' are not $(O(n), \varepsilon')$-quantum-indistinguishable where $\varepsilon' = \frac{1}{2\delta} - \frac{1}{2} - \varepsilon$.

Proof. Let $d_1 = 2^{m_1} = \dim(\mathcal{X})$ and $d_2 = 2^{m_2} = \dim(\mathcal{Y})$. First, we have the following claim:

Claim 5.4.20. *Let $\rho_X \in \mathrm{Dens}(\mathcal{X})$ and $\sigma_Y \in \mathrm{Dens}(\mathcal{Y})$ be two quantum states that are (t, ε)-quantum pseudorandom. Then the quantum states $(\rho_X \otimes \rho^{\mathsf{mm}}_{d_2}), (\rho^{\mathsf{mm}}_{d_1} \otimes \sigma_Y) \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y})$ are $(t - O(\log(\max\{d_1, d_2\})), 2\varepsilon)$-quantum-indistinguishable.*

Proof of Claim 5.4.20. Since it only takes $O(m)$ ancilla qubits and $O(m)$ many Hadamard gates to prepare a 2^m-dimensional maximally mixed state, $\rho_X \otimes \rho^{\mathsf{mm}}_{d_2}$ and $\rho^{\mathsf{mm}}_{d_1} \otimes \rho^{\mathsf{mm}}_{d_2}$ are $(s - O(\log(d_1)), \varepsilon)$-quantum-indistinguishable. Similarly, $\rho^{\mathsf{mm}}_{d_1} \otimes \rho_2$ and $\rho^{\mathsf{mm}}_{d_1} \otimes \rho^{\mathsf{mm}}_{d_2}$ are $(s - O(\log(d_2)), \varepsilon)$-quantum-indistinguishable. Therefore, $\rho_1 \otimes \rho^{\mathsf{mm}}_{d_2}$ and $\rho^{\mathsf{mm}}_{d_1} \otimes \rho_2$ are $(s - O(\log(\max\{d_1, d_2\})), 2\varepsilon)$-quantum-indistinguishable from each other. □

By Theorem 5.4.17, there exists pure states $\sigma_X \in \mathrm{Dens}(\mathcal{X})$ and $\sigma'_Y \in \mathrm{Dens}(\mathcal{Y})$ that both are $(t + O(\log(d_1 d_2)), \varepsilon/2)$-quantum-pseudorandom. Then by Claim 5.4.20,

$$\sigma = \sigma_X \otimes \rho^{\mathsf{mm}}_{d_2} \quad \text{and} \quad \sigma' = \rho^{\mathsf{mm}}_{d_1} \otimes \sigma'_Y$$

are (t, ε)-quantum-indistinguishable. Moreover, the min-entropies of σ and σ' are $\log d_2 = m_2$ and $\log d_1 = m_1$, respectively. Let

$$\rho = \frac{1}{2\delta}\Big(|0\rangle\langle 0| \otimes \rho^{\mathsf{mm}}_{d_1/2}\Big) \otimes \sigma'_Y + \Big(1 - \frac{1}{2\delta}\Big)\Big(|1\rangle\langle 1| \otimes \rho^{\mathsf{mm}}_{d_1/2}\Big) \otimes \sigma'_Y \,.$$

Then ρ is δ-dense in σ'. By Lemma 5.4.16, for every ρ' that is δ-dense in σ, ρ' must be of the form $\sigma_X \otimes \rho'_Y$ for some $\rho'_Y \in \mathrm{Dens}(\mathcal{Y})$. Now we define a quantum distinguisher A whose corresponding measurement operator is

$$\Pi = |0\rangle\langle 0| \otimes \mathbb{1}_{d_1/2} \otimes \mathbb{1}_{d_2},$$

which essentially measures the first input qubit in the standard basis and output its complement. Thus it can be implemented by a circuit of size $O(\log d_1 + \log d_2) = O(n)$. Then

$$\Pr\big[\mathsf{A}(\rho) = 1\big] = \langle \Pi \,,\, \rho \rangle = \frac{1}{2\delta}\,.$$

On the other hand,

$$\begin{aligned}
\Pr\Big[\mathsf{A}(\rho') = 1\Big] &= \Pr\Big[\mathsf{A}(\sigma_X \otimes \rho'_Y) = 1\Big] \\
&= \Pr\Big[\mathsf{A}(\sigma_X \otimes \rho^{\mathsf{mm}}_{d_2}) = 1\Big] \\
&\le \Pr\Big[\mathsf{A}(\rho^{\mathsf{mm}}_{d_1} \otimes \rho^{\mathsf{mm}}_{d_2}) = 1\Big] + \varepsilon = \frac{1}{2} + \varepsilon\,.
\end{aligned}$$

Therefore, $\big|\Pr[\mathsf{A}(\rho) = 1] - \Pr[\mathsf{A}(\rho')] = 1\big| > \frac{1}{2\delta} - \frac{1}{2} - \varepsilon$. □

Corollary 5.4.21. *Given $t \in \mathbb{N}, n > O(\log(t/\varepsilon))$, and $\varepsilon \in (0, 1/4)$, there exist quantum states $\rho, \sigma \in \mathrm{Dens}(\mathbb{C}^{2^n})$ such that $\mathrm{D}^{t,\varepsilon}_{\mathsf{HILL\text{-}2\text{-}max}}(\rho \,\|\, \sigma) \le 1$ but $\mathrm{D}^{O(n),1/4}_{\mathsf{HILL\text{-}1\text{-}max}}(\rho \,\|\, \sigma) = \infty$.*

Summarily, in the quantum setting, HILL-1 max-divergence being small does imply HILL-2 max-divergence being small (Lemma 5.4.14, and then pseudo max-divergence being small (Proposition 5.4.8. However, we show an counter example where HILL-2 max-divergence is small, but HILL-2 max-divergence is unbounded.

Remark 5.4.22. *The existence of a pseudorandom pure state (Theorem 5.4.17) only holds when we consider quantum distinguishers without quantum advice. Otherwise, for a pure state ρ, one can hardwire the same state as advice, allowing it to be distinguished from a maximally mixed state by using a Swap Test. (See Appendix 5.7.1 for more about pseudorandom pure states against quantum distinguishers with quantum advice.) Therefore, the separation between HILL-1 and HILL-2 type of computational relative entropies only holds when quantum distinguishers do not have quantum advice.*

5.5 Simulating Quantum Auxiliary Input

Let (X, Z) be a classical joint distribution over $\{0,1\}^n \times \{0,1\}^\ell$. The classical *Leakage Simulation Lemma* asserts the existence of "low complexity" simulator function $\mathsf{h} : \{0,1\}^n \to$

$\{0,1\}^\ell$ such that (X, Z) and $(X, \mathsf{h}(X))$ are indistinguishable by a family of distinguishers.

The Leakage Simulation Lemma implies many theorems in computational complexity and cryptography. For cryptographic applications, Jetchev and Pietrzak [JP14] used the lemma to give a simpler and quantitatively better proof for the leakage-resilient stream cipher by Pietrzak [Pie09]. Chung, Lui, and Pass [CLP15] also apply the lemma to study connections between various notions of Zero Knowledge. Moreover, the Leakage Simulation Lemma can be used to deduce the technical lemma of Gentry and Wichs [GW11] (for establishing lower bounds for succinct arguments), and the Leakage Chain Rule [JP14] for relaxed-HILL pseudoentropy [HILL99, GW11, Rey11]. For complexity theory, the Leakage Simulation Lemma implies the Regularity Lemma [TTV09], thus also the Impagliazzo's Hardcore Lemma [Imp95] and the Dense Model Theorem [RTTV08].

Here we generalize the Leakage Simulation Lemma to the quantum setting where the simulated system is quantum.

Theorem 5.5.1. *Let $\rho_{XZ} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho_Z^x \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Z})$ with $\dim(\mathcal{X}) = 2^n$ and $\dim(\mathcal{Z}) = d$. For every $t \in \mathbb{N}$ and $\varepsilon > 0$, there exists a quantum circuit $\mathsf{C} : \{0,1\}^n \to \mathrm{Dens}(\mathcal{Z})$ of size $t' = \mathrm{poly}(t, n, d, 1/\varepsilon)$ such that the cq-state $\sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \mathsf{C}(x)$ and ρ_{XZ} are (t, ε)-quantum-indistinguishable.*

We will prove the theorem in Section 5.5.2. Before that, we introduce some basic lemmas that will be used in the proof. In Section 5.5.3, we derive the Leakage Chain Rule for quantum relaxed-HILL entropy as a corollary of the Quantum Leakage Simulation Lemma.

5.5.1 Basic Lemmas

Rademacher Complexity

Cheng, Hsieh, and Yeh showed the bound on the Rademacher complexity of quantum measurements:

Theorem 5.5.2 ([CHY15, Theorem 4.2] (implicit))**.** *For $d \in \mathbb{N}$, let $\mathcal{W} = \mathrm{Dens}(\mathbb{C}^d)$, and $\mathcal{F} = \{\langle \Pi\,, \cdot\rangle : \Pi \in \mathrm{Meas}(\mathbb{C}^d)\}$. Then for every $w_1, \ldots, w_T \in \mathcal{W}$, the empirical Rademacher* *complexity of $\mathcal{F}$ is $\widehat{\mathfrak{R}}_{w_1,\ldots,w_T}(\mathcal{F}) = O(\sqrt{d/T})$.*

It is straightforward to show the linear property of (empirical) Rademacher complexity:

Proposition 5.5.3. *Given classes of functions $\mathcal{F}_1, \ldots, \mathcal{F}_N$ mapping from $\mathcal{W}$ to $\mathbb{R}$, use $p_1\mathcal{F}_1 + \cdots + p_N\mathcal{F}_N$ to denote the class*

$$\left\{ g : \mathcal{W}^N \to \mathbb{R} : g(w_1, \ldots, w_N) = \sum_{j=1}^{N} p_j \cdot f_j(w_j) \text{ where } f_j \in \mathcal{F}_j \ \forall j \in [N] \right\}.$$

Let $\vec{w}_i = (w_{i,1}, \ldots, w_{i,N}) \in \mathcal{W}^N$ for all $i \in [T]$. If $\widehat{\mathfrak{R}}_{w_{1,j},\ldots,w_{T,j}}(\mathcal{F}_j) \leq r_j$ for all $j \in [N]$, then $\widehat{\mathfrak{R}}_{\vec{w}_1,\ldots,\vec{w}_T}(\mathcal{G}) \leq \sum_{j=1}^{N} p_j r_j$, where $\mathcal{G} = p_1\mathcal{F}_1 + \cdots + p_N\mathcal{F}_N$.

Proof.

$$\begin{aligned}
\widehat{\mathfrak{R}}_{\vec{w}_1,\ldots,\vec{w}_T}(\mathcal{G}) &= \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sup_{g \in \mathcal{G}} \frac{1}{T} \sum_{i=1}^{T} \gamma_i \cdot g(\vec{w}_i)\right] \\
&= \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sup_{\forall j \in [N], f_j \in \mathcal{F}_j} \frac{1}{T} \sum_{i=1}^{T} \gamma_i \cdot \sum_{j=1}^{N} p_j \cdot f_j(w_{i,j})\right] \\
&= \mathop{\mathbb{E}}_{\gamma_1,\ldots,\gamma_T}\left[\sum_{j=1}^{N} p_j \cdot \sup_{f_j \in \mathcal{F}_j} \frac{1}{T} \sum_{i=1}^{T} \gamma_i f_j(w_{i,j})\right] \\
&= \sum_{j=1}^{N} p_j \cdot \widehat{\mathfrak{R}}_{w_{j,1},\ldots,w_{j,T}}(\mathcal{F}_j) \leq \sum_{j=1}^{N} p_j r_j \,.
\end{aligned}$$

□

Tomography

In a quantum tomography problem, the goal is to learn the behavior or even a description of a quantum circuit or quantum state. Our form of this will come up in the proof of the Quantum Leakage Simulation Lemma (Theorem 5.5.1), where we would like to find a quantum state that maximizes the acceptance probability of a given quantum distinguisher. This task is formulated in Definition 5.5.9 below. Here we provide a solution with runtime poly(t, d), which suffices for our applications.

Our tomography algorithm also uses a solution to the QCkt-Value Problem (Definition 5.5.4) and the QCkt-Tomography Problem (Definition 5.5.7), described as follows.

Definition 5.5.4 (QCkt-Value Problem)**.** *The* QCkt-Value(t, ε) *problem is a computational problem defined as follows:*

- *Input: a description of a quantum distinguisher* D *with no input (only ancillas) of size* t *with binary output* $\{0, 1\}$*, and an error parameter* $0 < \varepsilon < 1$*.*
- *Task: output an estimate* $\widehat{p}$ *of the probability* $p = \Pr[\mathsf{D}() = 1]$ *such that* $|\widehat{p} - p| \le \varepsilon$*.*

Lemma 5.5.5. *There exists a uniform quantum algorithm* A *that solves* QCkt-Value(t, ε) *with probability at least* $1 - \gamma$ *in time* $O\big(t \cdot \log(1/\gamma)/\varepsilon^2)\big)$*.*

Proof. The algorithm independently runs the circuit D t times (with fresh ancilla qubits each time) and set $\widehat{p}$ to be the fraction of times D outputs 1. By a Chernoff bound, we have

$$\Pr\big[|p - \widehat{p}| > \varepsilon\big] < 2^{-\Omega(t\varepsilon^2)} \le \gamma.$$

for $t = O\big(\log(1/\gamma)/\varepsilon^2\big)$. Each trial takes $O(t)$ time. Therefore, the total running time is $O\big(t \cdot \log(1/\gamma)/\varepsilon^2\big)$. □

Remark 5.5.6. *It is worth mentioning that by using a quantum speed-up (e.g., [Mon15]), one can improve the dependence on* $1/\varepsilon$ *quadratically, although this improvement is not crucial for our purposes. On the other hand, there is a lower bound [HHJ⁺16] saying a significant improvement on the dependency on* t *is impossible.*

Definition 5.5.7 (QCkt-Tomography Problem)**.** *The* QCkt-Tomography(t, d, ε) *problem is a computational problem defined as follows:*

- *Input: a description of a quantum distinguisher* $\mathsf{D} : \mathrm{Dens}(\mathbb{C}^d) \to \{0, 1\}$ *of size* t*, and an error parameter* $0 < \varepsilon < 1$*.*
- *Task: let* Π *be the corresponding quantum measurement of* C*. Output an explicit description (as a* $d \times d$ *matrix) of a quantum measurement* $\widehat{\Pi}$ *such that* $\big\|\Pi - \widehat{\Pi}\big\|_{\mathsf{op}} \le \varepsilon$*.*

Lemma 5.5.8. *There exists a quantum algorithm running in time* $\mathrm{poly}(t, d, 1/\varepsilon, \log(1/\gamma))$ *that solves* QCkt-Tomography(t, d, ε) *Problem with probability at least* $1 - \gamma$*. .*

Proof. The strategy is to estimate each entry of the matrix Π by feeding special input states to circuit C and observing the statistics of the output bit (i.e., a tomography process for the POVM Π (e.g., [LFC$^+$09])).

It only costs $O(t \log s)$ time for a quantum algorithm to execute C once [Val76]. The total running time then depends on the number of executions of C for the desired efficiency. To that end, we will leverage the following set of special input states, which suffice to determine the value of any positive semidefinite operator over the input space. Let $\{|1\rangle, \cdots, |d\rangle\}$ be any orthonormal basis in $\mathbb{C}^d$. Define the following set of density operators:

$$\forall n = 1, \cdots, d, \qquad A_{n,n} = |n\rangle\langle n|, \tag{5.8}$$

$$\forall 1 \leq n < m \leq d, \quad A^{\mathrm{re}}_{n,m} = |\psi_{n,m}\rangle\langle\psi_{n,m}|, |\psi_{n,m}\rangle = \tfrac{1}{\sqrt{2}}(|n\rangle + |m\rangle), \tag{5.9}$$

$$\forall 1 \leq n < m \leq d, \quad A^{\mathrm{im}}_{n,m} = |\phi_{n,m}\rangle\langle\psi_{n,m}|, |\phi_{n,m}\rangle = \tfrac{1}{\sqrt{2}}(|n\rangle + i|m\rangle). \tag{5.10}$$

Also let

$$\alpha_{n,n}(\Pi) = \mathrm{Tr}(A_{n,n}\Pi)$$
$$\alpha^{\mathrm{re}}_{n,m}(\Pi) = \mathrm{Tr}(A^{\mathrm{re}}_{n,m}\Pi)$$
$$\alpha^{\mathrm{im}}_{n,m}(\Pi) = \mathrm{Tr}(A^{\mathrm{im}}_{n,m}\Pi)$$

The collection of values $\alpha_{n,n}(\Pi)$ for $n = 1, \cdots, d$, and $\alpha^{\mathrm{re}}_{n,m}(\Pi)$ and $\alpha^{\mathrm{im}}_{n,m}(\Pi)$ for $1 \leq n < m \leq d$ uniquely determines any positive semidefinite operator Π.[7] It suffices to collect these α values to within small error to approximate Π. We will use Lemma 5.5.5 for this purpose. Overall, by a union bound, with probability $1 - \gamma$, we can collect a set of $\tilde{\alpha}$ values that approximate the original α values each within an additive error η in time $d^2 \cdot O\left(t \cdot \log(d/\gamma)/\eta^2\right) =$

[7] It is not hard to see that $\alpha_{n,n}(\Pi)$ determines all the diagonal entries. Every off-diagonal entries (n, m) (or its conjugate at (m, n)) is then determined by $\alpha^{\mathrm{re/im}}_{n,m}(\Pi)$ together with the information about the diagonal entree (n, n) and (m, m).

$\mathrm{poly}(t, d, \log(1/\gamma), 1/\eta)$. Namely, for all n, m, we have

$$\begin{cases} |\tilde{\alpha}_{n,n} - \alpha_{n,n}(\Pi)| \leq \eta \\ |\tilde{\alpha}^{\mathrm{re}}_{n,m} - \alpha^{\mathrm{re}}_{n,m}(\Pi)| \leq \eta \\ |\tilde{\alpha}^{\mathrm{im}}_{n,m} - \alpha^{\mathrm{im}}_{n,m}(\Pi)| \leq \eta \end{cases} . \tag{5.11}$$

We can thus solve the following semidefinite program (SDP) to recover an approximate $\widehat{\Pi}$:

$$\text{Goal: find a } \widehat{\Pi}$$

$$\text{Subject to:} \begin{cases} |\tilde{\alpha}_{n,n} - \alpha_{n,n}(\widehat{\Pi})| \leq \eta, \\ |\tilde{\alpha}^{\mathrm{re/im}}_{n,m} - \alpha^{\mathrm{re/im}}_{n,m}(\widehat{\Pi})| \leq \eta, \\ 0 \leq \widehat{\Pi} \leq \mathbb{1}_d \end{cases}$$

By Equation 5.11, this SDP is feasible. We claim that any feasible solution $\widehat{\Pi}$ is a good approximate of Π. Specifically, by Equation 5.11, the definition of the SDP and the triangle inequality, we have

$$\begin{cases} |\tilde{\alpha}_{n,n}(\widehat{\Pi}) - \alpha_{n,n}(\Pi)| \leq 2\eta \\ |\tilde{\alpha}^{\mathrm{re}}_{n,m}(\widehat{\Pi}) - \alpha^{\mathrm{re}}_{n,m}(\Pi)| \leq 2\eta \\ |\tilde{\alpha}^{\mathrm{im}}_{n,m}(\widehat{\Pi}) - \alpha^{\mathrm{im}}_{n,m}(\Pi)| \leq 2\eta \end{cases} ,$$

which implies $\|\widehat{\Pi} - \Pi\|_{\mathsf{max}} \leq \sqrt{(2\eta)^2 + (2\eta)^2} = O(\eta)$. By Equation (5.1), we have

$$\left\|\widehat{\Pi} - \Pi\right\|_{\mathsf{op}} \leq d \cdot \left\|\widehat{\Pi} - \Pi\right\|_{\mathsf{max}} = O(d\eta) .$$

It then suffices to choose $\eta = O(\varepsilon/d)$. Overall, the above algorithm succeeds with probability at least $1 - \gamma$ and runs in $\mathrm{poly}(t, d, 1/\varepsilon, \log(1/\gamma))$ time. □

Once we know how to approximate the quantum effect matrix of a given quantum distinguisher, we are ready to solve the our main tomography problem:

Definition 5.5.9 (QCkt-Max-Sat Problem)**.** *The* QCkt-Max-Sat(t, d, ε) *problem is a computational problem defined as follows:*

- *Input: a description of a quantum distinguisher* $\mathsf{D} : \mathrm{Dens}(\mathbb{C}^d) \to \{0,1\}$ *of size* t*, and an error parameter* $0 < \varepsilon < 1$*.*

- *Task: output an explicit description (as a density matrix) of a quantum state* $\rho \in \mathrm{Dens}(\mathbb{C}^d)$ *such that* $\mathsf{D}(\rho) > \max_\sigma \mathsf{D}(\sigma) - \varepsilon$*.*

Theorem 5.5.10. *There exists a (uniform) quantum algorithm* A *that solves* $\textsc{QCkt-Max-Sat}(t, d, \varepsilon)$ *with probably at least* $1 - \gamma$ *in time* $\mathrm{poly}(t, d, 1/\varepsilon, \log(1/\gamma))$*.*

Proof. The proof follows from Lemma 5.5.8 and an application of a spectrum decomposition. Specifically, let Π be the corresponding quantum measurement of D. By Lemma 5.5.8, there exists a quantum circuit of time complexity $\mathrm{poly}(t, d, 1/\varepsilon, \log(1/\gamma))$ and outputs a description of $\widehat{\Pi}$ such that $\left\|\widehat{\Pi} - \Pi\right\|_{\mathsf{op}} \leq \varepsilon/2$ with probability $1 - \gamma$. That means for all $\tau \in \mathrm{Dens}(\mathbb{C}^d)$,

$$\left|\langle \widehat{\Pi}\,,\, \tau\rangle - \langle \Pi\,,\, \tau\rangle\right| \leq \varepsilon/2\,. \tag{5.12}$$

We then run a spectrum decomposition on $\widehat{\Pi}$ and choose $\rho = |\psi\rangle\langle\psi|$ to be the density operator corresponding to the eigenvector $|\psi\rangle$ with the largest eigenvalue of $\widehat{\Pi}$. This step can be done in $\mathrm{poly}(d)$ given that dimension of $\widehat{\Pi}$ is d. Thus, we have

$$\langle \widehat{\Pi}\,,\, \rho\rangle \geq \max_\sigma \langle \widehat{\Pi}\,,\, \sigma\rangle. \tag{5.13}$$

By Equation (5.12), we have

$$\begin{aligned}
\langle \Pi\,,\, \rho\rangle &\geq \langle \widehat{\Pi}\,,\, \rho\rangle - \varepsilon/2 \\
&\geq \max_\sigma \langle \widehat{\Pi}\,,\, \sigma\rangle - \varepsilon/2 \\
&\geq \max_\sigma \langle \Pi\,,\, \sigma\rangle - \varepsilon/2 - \varepsilon/2 \\
&= \max_\sigma \langle \Pi\,,\, \sigma\rangle - \varepsilon.
\end{aligned}$$

The overall complexity is $\mathrm{poly}\big(t, d, 1/\varepsilon, \log(1/\gamma)\big)$, which completes the proof.

□

5.5.2 Proof of Quantum Leakage Simulation Lemma

Theorem 5.5.1 (restatement)**.** *Let* $\rho_{XZ} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho_Z^x \in \mathrm{Dens}(\mathcal{X}\otimes\mathcal{Z})$ *with* $\dim(\mathcal{X}) = 2^n$ *and* $\dim(\mathcal{Z}) = d$*. For every* $t \in \mathbb{N}$ *and* $\varepsilon > 0$*, there exists a quantum circuit* $\mathsf{C} : \{0,1\}^n \to \mathrm{Dens}(\mathcal{Z})$ *of size* $t' = \mathrm{poly}(t, n, d, 1/\varepsilon)$ *such that the cq-state* $\sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x|\otimes\mathsf{C}(x)$ *and* ρ_{XZ} *are* (t,ε)*-quantum-indistinguishable.*

Proof. Suppose for contradiction that for all size t' quantum circuits $\mathsf{C} : \{0,1\}^n \to \mathrm{Dens}(\mathbb{C}^d)$, there exists a quantum distinguisher on the space $\mathrm{Dens}(\mathbb{C}^{2^n}) \times \mathrm{Dens}(\mathbb{C}^d)$ of size t such that

$$\mathbb{E}\Big[\mathsf{D}(\rho_{XZ})\Big] - \mathbb{E}\Big[\mathsf{D}\Big(\textstyle\sum_x p_x|x\rangle\langle x| \otimes \mathsf{C}(x)\Big)\Big] \geq \varepsilon\,. \tag{5.14}$$

We can characterize a quantum distinguisher D by a set of measurement operators $\{\Pi_x\}_{x\in\{0,1\}^n}$ by letting the corresponding measurement operator of $\mathsf{D}(x,\cdot)$ be Π_x.

Then Equation (5.14) can be written as

$$\sum_x p_x\langle\Pi_x\,,\,\rho_Z^x\rangle - \sum_x p_x\langle\Pi_x\,,\,\mathsf{C}(x)\rangle \geq \varepsilon.$$

First, we extend the above statement about circuits C of bounded size to distributions of circuits $\bar{\mathsf{C}}$ of bounded size via the following claim.

Claim 5.5.11. *For every distribution* $\bar{\mathsf{C}}$ *over size* t'' *quantum circuit with* $t'' = t'/O(d/\varepsilon^2)$*, there exists a distinguisher* D *of size* t *such that*

$$\sum_x p_x\Big\langle\Pi_x\,,\,\rho_Z^x\Big\rangle - \mathop{\mathbb{E}}_{\mathsf{C}\overset{r}{\leftarrow}\bar{\mathsf{C}}}\Big[\textstyle\sum_x p_x\langle\Pi_x\,,\,C(x)\rangle\Big] \geq \varepsilon/2,$$

where Π_x *is the measurement operator of* $\mathsf{D}(x,\cdot)$*.*

Proof of Claim 5.5.11. Suppose for contradiction, there is a distribution $\bar{\mathsf{C}}$ over size t'' circuit such that for all distinguisher D of size t,

$$\sum_x p_x\Big\langle\Pi_x\,,\,\rho_Z^x\Big\rangle - \mathop{\mathbb{E}}_{\mathsf{C}\overset{r}{\leftarrow}\bar{\mathsf{C}}}\Big[\textstyle\sum_x p_x\langle\Pi_x\,,\,\mathsf{C}(x)\rangle\Big] < \varepsilon/2\,. \tag{5.15}$$

As in the proof of Theorem 5.3.8, we will use the generalization bound of Rademacher complexity (5.3.10) to argue the existence of a low complexity circuit $\widehat{\mathsf{C}}$ that approximates $\bar{\mathsf{C}}$.

Define $g_{\{\Pi_x\}} : \left(\mathrm{Dens}(\mathbb{C}^d)\right)^{2^n} \to [0,1]$ to be $g_{\{\Pi_x\}}(\{\rho^x\}) \stackrel{\text{def}}{=} \sum_x p_x \langle \Pi_x \,,\, \rho^x \rangle$. Let $\mathcal{G}_t$ contains the set of functions $g_{\{\Pi_x\}}$ where $\{\Pi_x\}$ is the corresponding set of measurement operator of some D on $\{0,1\}^n \times \mathrm{Dens}(\mathbb{C}^d)$ of size $\le t$, and $\mathcal{W} = \left(\mathrm{Dens}(\mathbb{C}^d)\right)^{2^n}$. By Theorem 5.5.2 and Lemma 5.5.3, we know that for every $w_1 \dots w_T \in \mathcal{W}$,

$$\widehat{\mathfrak{R}}_{w_1,\dots,w_T}\left(\mathcal{G}_\infty\right) = \widehat{\mathfrak{R}}_{w_1,\dots,w_T}\left(\textstyle\sum_x p_x \mathcal{F}\right) = \sum_x p_x \cdot O\big(\sqrt{d/T}\big) = O\big(\sqrt{d/T}\big),$$

where $\mathcal{F} = \{\langle \Pi \,,\, \cdot \rangle : \Pi \in \mathrm{Meas}(\mathbb{C}^d)\}$

Thus, for every distribution W on $\mathcal{W}$, we have $\mathfrak{R}_{W,T}\left(\mathcal{G}_\infty\right) = O\big(\sqrt{d/T}\big)$. In particular, letting $W = \bar{C}(x)$, and by the monotonicity of Rademacher complexity, we have $\mathfrak{R}_{\bar{\mathsf{C}}(x),T}\left(\mathcal{G}_s\right) = O\big(\sqrt{d/T}\big)$. Applying Theorem 5.3.10, there exist circuits $\mathsf{C}_1, \dots, \mathsf{C}_T$ of size at most t'' such that

$$\left| \frac{1}{T} \sum_{i=1}^{T} \sum_x p_x \Big\langle \Pi_x \,,\, \mathsf{C}_i(x) \Big\rangle - \mathop{\mathbb{E}}_{\mathsf{C} \xleftarrow{r} \bar{\mathsf{C}}} \left[\sum_x p_x \Big\langle \Pi_x \,,\, \mathsf{C}(x) \Big\rangle \right] \right| < \varepsilon/2$$

for some $T = O(d/\varepsilon^2)$. Let $\widehat{\mathsf{C}}$ be the quantum circuits that choose uniformly at random from $\mathsf{C}_1, \dots, \mathsf{C}_T$ to run. Then the circuit size of $\widehat{\mathsf{C}}$ is $t'' \cdot O(d/\varepsilon^2) = t'$, and

$$\left| \sum_x p_x \Big\langle \Pi_x \,,\, \widehat{\mathsf{C}}(x) \Big\rangle - \mathop{\mathbb{E}}_{\mathsf{C} \xleftarrow{r} \bar{\mathsf{C}}} \left[\sum_x p_x \Big\langle \Pi_x \,,\, \mathsf{C}(x) \Big\rangle \right] \right| < \varepsilon/2,$$

which together with 5.15 contradicts our assumption above that every circuit C of size t' has a distinguisher D of size t such that Equation 5.14 holds. □

Once we have Claim 5.5.11, we apply the Nonuniform Quantum Min-Max Theorem (Theorem 5.3.8) to the following game:

- The strategy space $\mathcal{A}$ of Player 1 is the convex hull of

$$\left\{ \text{cq-states } \textstyle\sum_x p_x |x\rangle\langle x| \otimes C(x) \;\middle|\; \mathsf{C} : \{0,1\}^n \to \mathrm{Dens}(\mathbb{C}^{2^\ell}) \text{ of size } t'' \right\}.$$

- The strategy space $\mathcal{B}$ of Player 2 is the set of all distinguishers of size at most t.

- For the payoff function $g : \mathcal{A} \times \mathcal{B} \to [0,1]$, we first define the mapping M to be

$$M(\mathsf{D}) = \frac{1}{2}\Big(\left(\mathbb{E}[\mathsf{D}(\rho_{XZ})] + 1\right) \cdot \mathbb{1}_{\mathcal{X} \otimes \mathcal{Z}} - \Pi_{\mathsf{D}} \Big).$$

where Π_{D} is the corresponding measurement operator of D. Then for $\sigma_{XZ} \in \mathcal{A}$ and $\mathsf{D} \in \mathcal{B}$,

$$g(\sigma_{XZ}, \mathsf{D}) = \Big\langle M(D)\,,\, \rho_{XZ} \Big\rangle = \frac{1}{2}\Big(\mathbb{E}[\mathsf{D}(\rho_{XZ})] + 1 - \mathbb{E}[\mathsf{D}(\sigma_{XZ})]\Big)$$

Claim 5.5.11 tells us that for all $\sigma_{XZ} \in \mathcal{A}$, there exists $D \in \mathcal{B}$ such that $g(\sigma_{XZ}, \mathsf{D}) > (1+\varepsilon/2)/2$. By the Nonuniform Quantum Min-Max Theorem, we deduce that there exists a quantum distinguisher $\widehat{\mathsf{D}}$ of size $t_{\widehat{\mathsf{D}}} = t \cdot O\left(\frac{1}{\varepsilon^2}(n + \log d)\right)$ such that for all $\sigma_{XZ} \in \mathcal{A}$, $g(\sigma_{XZ}, \widehat{\mathsf{D}}) > (1+\varepsilon/2)/2 - \varepsilon/8$. That is, for all quantum circuit $\widehat{\mathsf{C}} : \{0,1\}^n \to \mathrm{Dens}(\mathbb{C}^d)$ of size t'',

$$\mathbb{E}\Big[\widehat{\mathsf{D}}\Big(\rho_{XZ}\Big)\Big] - \mathbb{E}\Big[\widehat{\mathsf{D}}\Big(\textstyle\sum_x p_x |x\rangle\langle x| \otimes \widehat{\mathsf{C}}(x)\Big)\Big] > \varepsilon/8. \tag{5.16}$$

Writing the corresponding measurement operator of $\widehat{\mathsf{D}}(x,\cdot)$ as Π_x for $x \in \{0,1\}^n$, we have

$$\mathbb{E}\Big[\widehat{\mathsf{D}}(\rho_{XZ})\Big] = \sum_{x \in \{0,1\}^n} p_x \Big\langle \Pi_x\,,\, \rho_Z^x \Big\rangle\,.$$

Now, define the quantum circuit $\mathsf{C} : \{0,1\}^n \to \mathrm{Dens}(\mathbb{C}^d)$ as follows on input $x \in \{0,1\}^n$.

1. Apply Lemma 5.5.10 to solve the $(t_{\widehat{\mathsf{D}}}, d, \varepsilon/32)$-QCkt-Max-Sat Problem (Definition 5.5.9) with the quantum circuit $\widehat{\mathsf{D}}(x,\cdot)$. Therefore, there exists an algorithm with running time $\mathrm{poly}\big(t_{\widehat{\mathsf{D}}}, d, 1/\varepsilon)$ outputting the description (in density matrix) of σ_x such that with probability at least $1 - \varepsilon/32$,

$$\Big\langle \Pi_x\,,\, \sigma_x \Big\rangle \geq \max_\rho \langle \Pi_x, \rho\rangle\,[2] - \frac{\varepsilon}{16}.$$

2. Construct and output the quantum state σ_x based on the its density matrix description, which can be done by a circuit of size $\mathrm{poly}(d)$ [SBM05].

The total running time of C is $\mathrm{poly}(t, n, d, 1/\varepsilon)$. Suppose the running time of C is at most t'', we have

$$\mathbb{E}\Big[\widehat{\mathsf{D}}\left(\textstyle\sum_x p_x |x\rangle\langle x| \otimes C(x)\right)\Big] = \sum_{x \in \{0,1\}^n} p_x \Big\langle \Pi_x\,,\, \sigma_x \Big\rangle$$

$$
\begin{aligned}
&\geq \left(1 - \frac{\varepsilon}{32}\right) \left(\sum_{x \in \{0,1\}^n} p_x \max_{\rho} \left\langle \Pi_x \, , \, \rho \right\rangle - \frac{\varepsilon}{32} \right) \\
&\geq \left(1 - \frac{\varepsilon}{32}\right) \left(\max_{\mathrm{Tr}_Z(\rho'_{XZ}) = \mathrm{Tr}_Z(\rho_{XZ})} \mathbb{E}\left[\widehat{\mathsf{D}}(\rho'_{XZ})\right] - \frac{\varepsilon}{32} \right) \\
&\geq \mathbb{E}\left[\widehat{\mathsf{D}}(\rho_{XZ})\right] - \frac{\varepsilon}{16},
\end{aligned}
$$

which contradicts Equation (5.16). □

5.5.3 Leakage Chain Rule

The Leakage Chain Rule for (classical) relaxed-HILL entropy have number of applications in cryptography such as leakage-resilient cryptography [DP08], memory delegation [CKLR11], and deterministic encryption [FOR15]. In this section, we will prove the leakage chain rule for quantum relaxed-HILL pseudoentropy with quantum leakage.

First, we recall the generic statement of the leakage chain rule [DP08]. Let (X, Y, Z) be a joint distribution (which will be a quantum state ρ_{XYZ} in the quantum setting) where X, Y and Z are viewed as a source, prior knowledge, and leakage, respectively. The leakage chain rule says that, if the entropy of X conditioned on Y is at least k, then the entropy of X conditioned on both Y and Z retains at least $k - \mathrm{len}(Z)$, where $\mathrm{len}(Z)$ is the length (measured in bit/qubit) of Z. That is

$$
\mathrm{H}(X|YZ) \geq \mathrm{H}(X|Y) - \mathrm{len}(Z) \, .
$$

In the asymptotic setting, we will focus on the case that $\mathrm{len}(Z) = O(\log \kappa)$ and the length of $\mathrm{len}(X)$ and $\mathrm{len}(Y)$ could be $\mathrm{poly}(\kappa)$, where κ is the security parameter.

The Leakage Chain Rule holds for quantum min-entropy when Z is separable from XY, which is a necessary step in our proof of the Leakage Chain Rule for computational notions.

Theorem 5.5.12 ([WTHR11, Lemma 13])**.** *Let $\rho = \rho_{XYZ} = \sum_k p_k \rho^k_{XY} \otimes \rho^k_Z$ be a separable state on the space $\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z}$. Then*

$$
\mathsf{H}_{\mathsf{min}}(X|YZ)_\rho \geq \mathsf{H}_{\mathsf{min}}(X|Y)_\rho - \mathrm{len}(Z) \, .
$$

Remark 5.5.13. *In the general case — System XY may not be separable from Z, [WTHR11] also showed that* $\mathrm{H}_{\mathsf{min}}(X|YZ)_\rho \geq \mathrm{H}_{\mathsf{min}}(X|Y)_\rho - 2|\mathcal{Z}|$. *Furthermore, the equality holds if and only if ρ_{XZ} are maximally entangled.*

For pseudoentropies, it is known that the Leakage Chain Rule holds for classical relaxed-HILL min-entropy [DP08, RTTV08, GW11]. When there is no prior knowledge Y and Z is short (logarithmic in the security parameter), classical HILL pseudoentropy and classical relaxed-HILL pseudoentropy are equivalent. Thus the Leakage Chain Rule holds for classical HILL pseudoentropy also holds when there is no prior knowledge Y. However, if prior information Y is allowed, Krenn et al. [KPWW16] showed that the leakage lemma is unlikely to hold for standard HILL pseudoentropy. Specifically, assuming the existence of a perfectly binding commitment scheme, they constructed a joint distribution (X, Y, Z), where Z is a single random bit, such that $\mathrm{H}_{\mathsf{HILL\text{-}min}}(X|Y) \geq n$, but $\mathrm{H}_{\mathsf{HILL\text{-}min}}(X|Y,Z) \leq 1$. Therefore, we aim to prove a quantum leakage chain rule for relaxed-HILL pseudoentropy. In particular, we consider the case that only the leakage is quantum, namely the joint quantum state $\rho_{XYZ} = \sum_{xy} p_{xy}|xy\rangle\langle xy| \otimes \rho_Z^{xy}$ is a ccq-state.

Theorem 5.5.14. *Given $n, m, \ell, t' \in \mathbb{N}$ and $\varepsilon > 0$, let $\rho_{XYZ} = \sum_{(x,y)\in\{0,1\}^{n+m}} p_{xy}|xy\rangle\langle xy| \otimes \rho_Z^{xy} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z})$ be a ccq-state with $\dim(\mathcal{X}) = 2^n, \dim(\mathcal{Y}) = 2^m$, and $\dim(\mathcal{Z}) = 2^\ell$. If $\mathrm{H}^{t,\varepsilon}_{\mathsf{r\text{-}HILL\text{-}min}}(X|Y)_\rho \geq k$, then we have $\mathrm{H}^{t',\varepsilon'}_{\mathsf{r\text{-}HILL\text{-}min}}(X|YZ)_\rho \geq k - \ell$ where $t' = (t/\operatorname{poly}(n, m, 2^\ell, 1/\varepsilon))^{O(1)}$ and $\varepsilon' = 2\varepsilon$.*

We use the following lemma as an intermediate step to derive the Leakage Chain Rule.

Lemma 5.5.15 (quantum generalization of [GW11, Lemma 3.2]). *Given $n, \ell, t \in \mathbb{N}$ and $\varepsilon > 0$, let $\rho_{XZ} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho_Z^x \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Z})$ be a cq-state with $\dim(\mathcal{X}) = 2^n$ and $\dim(\mathcal{Z}) = 2^\ell$. For every X' that is (t, ε)-quantum-indistinguishable from X, there exists a quantum circuit C of size at most $s/2$ such that the cq-state*

$$\sigma_{X'Z'} = \sum_{x\in\{0,1\}^n} q_x|x\rangle\langle x| \otimes \mathsf{C}(x) \qquad \text{and} \qquad \rho_{XZ}$$

are (t', ε')-quantum-indistinguishable where $q_x = \Pr[X' = x]$, $t' = (t/\operatorname{poly}(n, 2^\ell, 1/\varepsilon))^{O(1)}$, *and* $\varepsilon' = 2\varepsilon$.

Proof. By Theorem 5.5.1, there exists a circuit $C : \{0,1\}^n \to \mathrm{Dens}(\mathbb{C}^{2^\ell})$ with size $s/2 = \mathrm{poly}(t', n, 2^\ell, 1/\varepsilon)$ such that $\rho_{XZ} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x|\otimes\rho_Z^x$ and $\sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x|\otimes\mathsf{C}(x)$ are (t', ε)-quantum-indistinguishable. Since X and X' are (t, ε)-quantum-indistinguishable, $\sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \mathsf{C}(x)$ and $\sigma_{X'Z'} = \sum_{x\in\{0,1\}^n} q_x|x\rangle\langle x| \otimes \mathsf{C}(x)$ are $(t/2, \varepsilon)$-quantum-indistinguishable. By the transitivity of indistinguishability, $\rho_{X'Z'}$ and σ_{XZ} are (t', ε')-quantum-indistinguishable. □

Once we have Lemma 5.5.15, we can derive the chain rule for quantum relaxed HILL pseudoentropy from the chain rule for quantum min-entropy of separable states (Theorem 5.5.12).

Proof of Theorem 5.5.14. $\mathrm{H}^{t,\varepsilon}_{\textsf{r-HILL-min}}(X|Y) \geq k$ implies there exists a joint distribution (X', Y') such that (X, Y) and (X', Y') are (t, ε)-indistinguishable and $\mathrm{H}_{\textsf{min}}(X'|Y') \geq k$. By Lemma 5.5.15, there exists an ℓ-qubit quantum circuit $\mathsf{C} : \{0,1\}^{n+m} \to \mathrm{Dens}(\mathbb{C}^\ell)$ such that

$$\rho_{XYZ} = \sum_{(x,y)\in\{0,1\}^{n+m}} p_{xy}|xy\rangle\langle xy| \otimes \rho_Z^{xy} \text{ and } \sigma_{X'Y'Z'} = \sum_{(x,y)\in\{0,1\}^{n+m}} q_{xy}|xy\rangle\langle xy| \otimes C(x, y),$$

where $q_{xy} = \Pr[X' = x, Y' = y]$ are (t', ε')-indistinguishable for some $t' = \left(t/\mathrm{poly}(n, 2^\ell, 1/\varepsilon)\right)^{O(1)}$ and $\varepsilon' = 2\varepsilon$. By the chain rule for quantum min-entropy of separable states (Theorem 5.5.12), $\mathrm{H}_{\textsf{min}}(X'|Y'Z')_\sigma \geq k - \ell$, which implies $\mathrm{H}^{t',\varepsilon'}_{\textsf{r-HILL-min}}(X|YZ)_\rho \geq k - \ell$. □

We have proved the quantum leakage chain rule for ccq-states. However, due to some barriers that we will mention in Appendix 5.7.2, our proof techniques do not extend to the case of open for the cqq-states (where the prior knowledge Y is also quantum).

Open Problem 5.5.16. *Let* $\rho_{XYZ} = \sum_{x\in\{0,1\}^n} p_x|x\rangle\langle x| \otimes \rho_{YZ}^x \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z})$ *be a cqq-state with* $\dim(\mathcal{X}) = 2^n, \dim(\mathcal{Y}) = 2^m$, *and* $\dim(\mathcal{Z}) = 2^\ell$ *If* $\mathrm{H}^{t,\varepsilon}_{\textsf{r-HILL-min}}(X|Y)_\rho \geq k$, *can we show that* $\mathrm{H}^{t',\varepsilon'}_{\textsf{r-HILL-min}}(X|YZ)_\rho \geq k - \ell$ *for some* $t' = \left(t/\mathrm{poly}(n, m, 2^\ell, 1/\varepsilon)\right)^{O(1)}$ *and* $\varepsilon' = O(\varepsilon)$*?*

Additionally, we do not know whether quantum HILL and relaxed-HILL entropies are equivalent, even for the case that Z is a single qubit. Thus our result does not imply a chain rule for quantum HILL entropies without prior knowledge.

Open Problem 5.5.17. *Let* $\rho_{XZ} \in \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Z})$ *be a cq-state with* $\dim(\mathcal{X}) = 2^n, \dim(\mathcal{Z}) = 1$ *If* $\mathrm{H}^{t,\varepsilon}_{\mathsf{r\text{-}HILL\text{-}min}}(X|Z)_\rho \geq k$*, can we show that* $\mathrm{H}^{t',\varepsilon'}_{\mathsf{HILL\text{-}min}}(X|Z)_\rho \geq k$ *for some* $t' = (t/\operatorname{poly}(n, 1/\varepsilon))^{O(1)}$ *and* $\varepsilon' = O(\varepsilon)$*?*

5.6 Application to Quantum Leakage-Resilient Cryptography

Classically, the Leakage Simulation Lemma and the Leakage Chain Rule have important applications in Leakage-Resilient Cryptography, which aims to construct secure cryptographic protocols even if side information about the honest parties' secrets leak to an adversary. For instance, the security of a leakage-resilient stream cipher based on a weak pseudorandom function (weak PRF) was proved using the classical Leakage Simulation Lemma [Pie09, JP14], and the security of the construction based on a pseudorandom generator (PRG) was proved by the classical Leakage Chain Rule [DP08].

Here, we use our Quantum Leakage Simulation Lemma to obtain a stream cipher that is secure against quantum adversary that can get quantum leakage as well as classical leakage, provided that the adversary has bounded quantum storage. (The classical storage of the adversary is unbounded.) The construction is the same as in [DP08] but instantiated with a PRG secure against quantum adversaries with quantum advice. Our proof follows the framework in [JP14], but with certain necessary modifications to make the proof go through in the quantum setting.

5.6.1 Quantum leakage-resilient stream-cipher

In this section, we generalize the leakage-resilient stream cipher defined in [DP08] to capture quantum leakage in the bounded-quantum-storage model. A *stream cipher* is given by a function $\mathsf{SC} : \{0,1\}^m \to \{0,1\}^m \times \{0,1\}^n$. Initially, the internal $s^{(0)} \xleftarrow{\mathrm{r}} \{0,1\}^m$. In the i-th round, $(s^{(i)}, x^{(i)}) = \mathsf{SC}(s^{(i-1)})$ is computed. When we iteratively apply the function SC, the internal state evolves and generates the output $X^{(1)}, X^{(2)}, \ldots$. In a *quantum leakage-resilient stream cipher*, we consider adversaries with quantum power that also learn some bounded-length quantum leakage $\lambda^{(i)}$ about the internal state $s^{(i-1)}$ that was used for generating $x^{(i)}$.

More precisely, we assume the leakage $\lambda^{(i)}$ only depends on the part that was used for evaluating $\mathsf{SC}(s^{(i-1)})$. That is, if we write $s^{(i-1)}$ as $(s^{(i-1)}_{\text{active}}, s^{(i-1)}_{\text{inactive}})$, such that $\mathsf{SC}(s^{(i-1)})$ is independent to $s^{(i-1)}_{\text{inactive}}$, then $\lambda^{(i)} = \lambda^{(i)}(s_{\text{active}})$ (following the "*only computation leaks*" model [MR04]). Informally, a quantum leakage-resilient stream cipher SC is secure within q rounds, if for all $i \in [q]$, $X^{(i)}$ is pseudorandom conditioned on the previous output $X^{(1)}, \ldots, X^{(i-1)}$, and the leakage states $\lambda^{(i)}(S^{(i-1)})$.

In this paper, we study the quantum leakage-resilient stream cipher in the *bounded-quantum-storage model* [DFSS08, KT08, WW08], where an adversary's *quantum memory size* is bounded. In this model, an adversary is more restricted. For example, as the number of rounds increases, it cannot store all quantum leakages in the memory. Instead, it has to convert some of them to classical bits by measurement in order to get more quantum leakage. For simplicity, in the below formal security definition, we assume the leakage occupies the whole quantum memory.

Security definition. Let $\mathsf{SC} : \{0,1\}^m \to \{0,1\}^m \times \{0,1\}^n$ be a quantum leakage-resilient stream cipher, and A be an adversary whose memory is a cq-state $\rho_{YZ} = \sum_y p_y |y\rangle\langle y| \otimes \rho_Z^y \in \mathrm{Dens}(\mathcal{Y} \otimes \mathcal{Z})$ with $\dim(\mathcal{Y}) = 2^{m_{\mathsf{A}}}$ and $\dim(\mathcal{Z}) = 2^{\ell}$. The adversary A is defined by a quantum leakage circuit $\lambda \leftarrow \{0,1\}^{n'+m_{\mathsf{A}}} \to \mathrm{Dens}(\mathbb{C}^{2^{\ell}})$, and a quantum circuit $\mathsf{C}_{\mathsf{A}} : \mathrm{Dens}(\mathcal{X} \otimes \mathcal{Y} \otimes \mathcal{Z}) \to \{0,1\}^{m_{\mathsf{A}}}$ of size t where $\dim(\mathcal{X}) = 2^n$.

We define a security game G_0^q in the bounded-quantum-storage model as described in Game 5.6.1. We also define the game $\widetilde{G}_0^q$, which is identical to the game G_0^q, except that in Step 2(c) of q-th round, $x^{(q)}$ is resampled from U_m instead of produced by SC.

We use $\mathsf{A}(G)$ to denote the output of the adversary A in a game G, where G depends on SC and A depends on L_{A} and C_{A} implicitly. The security of a quantum leakage-resilient stream cipher is defined as follows.

Definition 5.6.1. *A quantum leakage-resilient stream cipher* $\mathsf{SC} : \{0,1\}^m \to \{0,1\}^m \times \{0,1\}^n$ *is* $(t, \varepsilon, q, \ell)$-secure in the bounded-quantum-storage model *if for every quantum adversary* A *of size t with an ℓ-qubit memory and every $q' \in [q]$, G_0^q and $\widetilde{G}_0^q$ are ε-indistinguishable by* A.

Game 5.6.1: G_0^q

1. Initially, $s^{(0)} \xleftarrow{\mathrm{r}} U_m$, and the memory state of the adversary A is $\rho^{(0)}_{YZ} = \sum_y p^{(0)}_y |y\rangle\langle y| \otimes \rho_Z^{y\,(0)}$.
2. For $i = 1, \dots, q$, in the i-th round,
   (a) The stream cipher computes $(s^{(i)}, x^{(i)}) = \mathsf{SC}(s^{(i-1)}) \in \{0,1\}^n \times \{0,1\}^m$.
   (b) If $i < q$, the adversary learns the leakage of $s^{(i-1)}_{\text{active}}$ via L_{A}. That is, the memory state becomes
   $$\rho_{YZ}{}^{(i-1/2)} = \sum_y p_y^{(i-1)} |y\rangle\langle y| \otimes \lambda\Big(s^{(i-1)}_{\text{active}}, y\Big)\,,$$
   where $p_y^{(i-1)} = \Pr[Y^{(i-1)} = y]$
   (c) The adversary sees $x^{(i)}$. Its classical memory state becomes $\rho^{(i)}_{YZ} = \mathsf{C}_{\mathsf{A}}\big(\big|x^{(i)}\big\rangle\big\langle x^{(i)}\big| \otimes \rho_{YZ}{}^{(i-1/2)}\big)$.
3. The adversary outputs the first bit of $y^{(q)}$.

Namely

$$\Big|\Pr[\mathsf{A}(\widetilde{G}_0^q) = 1] - \Pr[\mathsf{A}(G_0^q) = 1]\Big| \leq \varepsilon\,.$$

5.6.2 Construction

The construction follows the one in [DP08], but here we require the extractors and the pseudorandom generators in the construction to be secure against quantum adversaries. Concretely, first, we define a function $\mathsf{f} : \{0,1\}^{k+n} \to \{0,1\}^{k+n}$, which serves as a building block of the construction:

$$\mathsf{f}(k, x) = \mathsf{Prg}(\mathsf{Ext}(k, x), x)\,,$$

where $\mathsf{Ext} : \{0,1\}^{k+n} \to \{0,1\}^m$ is a quantum-proof strong randomness extractor (e.g., Trevisan's extractor [Tre99, DPVR12]) and $\mathsf{Prg} : \{0,1\}^m \to \{0,1\}^{k+n}$ is a pseudorandom generator secure against quantum adversary with quantum advice. The existence of quantum-secure PRGs is known to follow from the quantum-hardness of lattice problems (e.g., learning with rounding [BPR12]). Formally,

Definition 5.6.2 (Quantum-proof strong randomness extractor)**.** *We say* $\mathsf{Ext} : \{0,1\}^{k+n} \to \{0,1\}^m$ *is a* $(k_{\mathsf{Ext}}, \varepsilon_{\mathsf{Ext}})$-quantum-proof extractor *if for all cq-states* $\rho_{KZ} = \sum_k p_k |k\rangle\langle k| \otimes \rho_Z^k$ *in the state space* $\mathcal{X} \otimes \mathcal{Z}$ *with* $\mathrm{H}_{\mathsf{min}}(K|Z)_\rho \geq k_{\mathsf{Ext}}$ *where* $\dim(\mathcal{K}) = 2^n$*, then we have that the trace distance between two ccq-state*

$$\sum_{k,x} p_k \cdot 2^{-n} \Big|\mathsf{Ext}(k,x)\Big\rangle\Big\langle\ \mathsf{Ext}(k,x)\Big| \otimes |x\rangle\langle x| \otimes \rho_Z^k \qquad \textit{and} \qquad \rho_{\mathcal{Y}}^{\mathsf{mm}} \otimes \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \rho_Z$$

is at most $\varepsilon_{\mathsf{Ext}}$.

Theorem 5.6.3 ([DPVR12])**.** *There exists a* $(k_{\mathsf{Ext}}, \varepsilon_{\mathsf{Ext}})$*-quantum-proof extractor* $\mathsf{Ext} : \{0,1\}^{k+n} \to \{0,1\}^m$ *with complexity* $\mathrm{poly}(k)$ *where* $m = k_{\mathsf{Ext}} - 4\log(1/\varepsilon_{\mathsf{Ext}}) - O(1)$.

Definition 5.6.4. *We say* $\mathsf{Prg} : \{0,1\}^m \to \{0,1\}^n$ *is an* $(t_{\mathsf{Prg}}, \varepsilon_{\mathsf{Prg}})$-quantum$^+$-secure pseudorandom generator *if for all quantum distinguishers* D *of size* t_{Prg} *with quantum advice,*

$$\Big|\Pr[\mathsf{D}(\mathsf{Prg}(U_m)) = 1] - \Pr[\mathsf{D}(U_n) = 1]\Big| \leq \varepsilon_{\mathsf{Prg}}.$$

Combining the extractor and the pseudorandom generator, we have the following claim for our building block f.

Claim 5.6.5. *Let* $\mathsf{Ext} : \{0,1\}^{k+n} \to \{0,1\}^m$ *be a* $(k_{\mathsf{Ext}}, \varepsilon_{\mathsf{Ext}})$*-quantum-proof extractor, and* $\mathsf{Prg} : \{0,1\}^{m+n} \to \{0,1\}^{k+n}$ *be an* $(t_{\mathsf{Prg}}, \varepsilon_{\mathsf{Prg}})$*-quantum+-secure pseudorandom generator. The the function* $\mathsf{f} : \{0,1\}^{k+n} \to \{0,1\}^{k+n}$ *defined as* $\mathsf{f}(K,X) \overset{\text{def}}{=} \mathsf{Prg}(\mathsf{Ext}(K,X), X)$ *satisfies the follows. If a cq-state* $\rho_{KZ} \in \mathcal{K} \otimes \mathcal{Z}$ *satisfies* $\mathrm{H}_{\mathsf{min}}(K|Z)_\rho \geq k_{\mathsf{Ext}}$*, then*

$$\sum_{k,x} p_k \cdot 2^{-n} \Big|\mathsf{f}(k,x)\Big\rangle\Big\langle\ \mathsf{f}(k,x)\Big| \otimes \rho_Z^k \quad \textit{and} \quad \rho_{\mathcal{K}}^{\mathsf{mm}} \otimes \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \rho_Z$$

are $(t_{\mathsf{Prg}}, \varepsilon_{\mathsf{Ext}} + \varepsilon_{\mathsf{Prg}})$*-quantum*$^+$*-indistinguishable.*

Proof. For every quantum distinguisher D (with quantum advice) of size at most t_{Prg},

$$\left|\Pr\left[\mathsf{D}\Big(\sum_{k,x} p_k \cdot 2^{-n} \Big|\mathsf{f}(k,x)\Big\rangle\Big\langle\ \mathsf{f}(k,x)\Big| \otimes \rho_Z^k\Big) = 1\right] - \Pr\left[\mathsf{D}\Big(\rho_{\mathcal{K}}^{\mathsf{mm}} \otimes \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \rho_Z\Big) = 1\right]\right|$$

$$= \left|\Pr\left[\mathsf{D}\Big(\sum_{k,x} p_k \cdot 2^{-n} \Big|\mathsf{Prg}(\mathsf{Ext}(k,x),x)\Big\rangle\Big\langle\ \mathsf{Prg}(\mathsf{Ext}(k,x),x)\Big| \otimes \rho_Z^k\Big) = 1\right]\right.$$

$$-\Pr\left[\mathsf{D}\left(\rho_{\mathcal{K}}^{\mathsf{mm}} \otimes \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \rho_Z\right) = 1\right]\Bigg|$$

$$\leq \left|\Pr\left[\mathsf{D}\left(\sum_{y,x} 2^{-m} \cdot 2^{-n} \Big|\mathsf{Prg}(y,x)\Big\rangle\Big\langle \mathsf{Prg}(y,x)\Big| \otimes \rho_Z^k\right) = 1\right]\right.$$

$$\left. -\Pr\left[\mathsf{D}\left(\rho_{\mathcal{K}}^{\mathsf{mm}} \otimes \rho_{\mathcal{X}}^{\mathsf{mm}} \otimes \rho_Z\right) = 1\right]\right| + \varepsilon_{\mathsf{Ext}}$$

$$\leq \varepsilon_{\mathsf{Prg}} + \varepsilon_{\mathsf{Ext}}.$$

The first inequality is because the trace distance between $\sum_{k,x} p_k \cdot 2^{-n}\Big|\mathsf{Ext}(k,x)\Big\rangle\Big\langle \mathsf{Ext}(k,x)\Big|$ and $\rho_{\mathcal{Y}}^{\mathsf{mm}}$ is at most $\varepsilon_{\mathsf{Ext}}$ and Proposition 5.2.1. The second inequality is due to the property of the quantum-secure pseudorandom generator in Definition 5.6.4. The system Z part can be seen as a quantum advice. □

Based on f, we define the stream cipher SC as follows. Suppose the internal state right before the i-th round be $s^{(i-1)} = \left(k_{\mathsf{L}}^{(i-1)}, x^{(i-1)}, k_{\mathsf{R}}^{(i-1)}\right)$. In the i-th round, the state becomes $s^{(i)} = (k_{\mathsf{L}}^{(i)}, x^{(i)}, k_{\mathsf{R}}^{(i)})$, where

$$\begin{cases} \left(k_{\mathsf{L}}^{(i)}, x^{(i)}\right) = \mathsf{f}\left(k_{\mathsf{L}}^{(i-1)}, x^{(i-1)}\right)\ ,\ k_{\mathsf{R}}^{(i)} = k_{\mathsf{R}}^{(i-1)} & \text{if } i \text{ is odd;} \\ \left(k_{\mathsf{R}}^{(i)}, x^{(i)}\right) = \mathsf{f}\left(k_{\mathsf{R}}^{(i-1)}, x^{(i-1)}\right)\ ,\ k_{\mathsf{L}}^{(i)} = k_{\mathsf{L}}^{(i-1)} & \text{if } i \text{ is even.} \end{cases}$$

We repeat $x^{(i)}$ in the internal state $s^{(i)}$ to make the definition consistent with the definition of stream cipher previously. Note only one of $k_{\mathsf{R}}^{(i)}$ and $k_{\mathsf{L}}^{(i)}$ is used when calculating $(s^{(i)}, x^{(i)})$ from $s^{(i-1)}$, so

$$s_{\text{active}}^{(i)} = \begin{cases} \left(k_{\mathsf{R}}^{(i)}, x^{(i)}\right) & \text{if } i \text{ is odd} \\ \left(k_{\mathsf{L}}^{(i)}, x^{(i)}\right) & \text{if } i \text{ is even} \end{cases}.$$

5.6.3 Security

The security game for the above construction in Section 5.6.2 is described in Security Game 5.6.2. We also include the steps (marked with tildes) that will be used by hybrid games to prove the security in the description. The whole system can be described by a ccccq-state ρ in the state space $\mathcal{K}_{\mathsf{L}} \otimes \mathcal{X} \otimes \mathcal{K}_{\mathsf{R}} \otimes \mathcal{Y} \otimes \mathcal{Z}$ where $\mathcal{Y} \otimes \mathcal{Z}$ is the memory of an adversary.

Games 5.6.2

1. Initially, let $k_{\mathsf{L}}^{(0)} \xleftarrow{\text{r}} U_k$, $x^{(0)} \xleftarrow{\text{r}} U_n$, and $k_{\mathsf{R}}^{(0)} \xleftarrow{\text{r}} U_k$. The adversary initialize its memory as $\rho_{YZ}^{(0)} = \sum_y p_y^{(0)} |y\rangle\langle y| \otimes \rho_Z^{y\,(0)}$. Thus, $\rho^{(0)} = \rho^{\mathsf{mm}}_{K_{\mathsf{L}}XK_{\mathsf{R}}} \otimes \rho_{YZ}^{(0)}$
2. For $i = 1 \to q-1$,

   (Below we state for the case i being odd. Swap all L's and R's if i is even.)

   (a) SC computes $k_{\mathsf{L}}^{(i)}, x^{(i)}$ and the adversary learns the leakage via L_{A}:

   $$\rho^{(i-1/2)} = \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}},y} p^{(i-1)}_{k_{\mathsf{L}}xk_{\mathsf{R}}y} \Big| \mathsf{f}(k_{\mathsf{L}},x), k_{\mathsf{R}}, y \Big\rangle \Big\langle\, \mathsf{f}(k_{\mathsf{L}},x), k_{\mathsf{R}}, y \Big| \otimes \lambda(k_{\mathsf{L}}, y)\,.$$

   ($\tilde{\text{a}}$) SC computes $k_{\mathsf{L}}^{(i)}, x^{(i)}$ and adversary's quantum memory is simulated by $\mathsf{C} : \{0,1\}^{n+k+m_{\mathsf{A}}} \to \mathrm{Dens}(\mathcal{Z})$ of size $t_{\mathsf{C}} = \mathrm{poly}(t, t_{\mathsf{SC}}, \varepsilon^{-1}, q, 2^{\ell})$:

   $$\rho^{(i-1/2)} = \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}},y} p^{(i-1)}_{k_{\mathsf{L}}xk_{\mathsf{R}}y} \Big| \mathsf{f}(k_{\mathsf{L}},x), k_{\mathsf{R}}, y \Big\rangle \Big\langle\, \mathsf{f}(k_{\mathsf{L}},x), k_{\mathsf{R}}, y \Big| \otimes \mathsf{C}\left(\mathsf{f}(k_{\mathsf{L}},x), y\right),$$

   s.t. $\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ}$ and "$\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ}$ in Step (a)" are $(t + t_{\mathsf{SC}}, \varepsilon/4q)$-quantum-indistinguishable. (by Theorem 5.5.1)

   ($\tilde{\tilde{\text{a}}}$) f in Step ($\tilde{\text{a}}$) is replaced by resampling from uniformly random strings:

   $$\rho^{(i-1/2)} = \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}},y} 2^{-(k+n)} \cdot p_y^{(i-1)} \Big| k_{\mathsf{L}}, x, k_{\mathsf{R}}, y \Big\rangle \Big\langle k_{\mathsf{L}}, x, k_{\mathsf{R}}, y \Big| \otimes \mathsf{C}\left(k_{\mathsf{L}}, x, y\right),$$

   (b) $x^{(i)}$ is revealed, the adversary's memory is updated by C:

   $$\rho^{(i)} = \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}}} p^{(i-1/2)}_{k_{\mathsf{L}},x,k_{\mathsf{R}}} |k_{\mathsf{L}}, x, k_{\mathsf{R}}\rangle\langle k_{\mathsf{L}}, x, k_{\mathsf{R}}| \otimes \mathsf{C}_{\mathsf{A}}\big(|x\rangle\langle x| \otimes \rho_{YZ}^{k_{\mathsf{L}}xk_{\mathsf{R}}\,(i-1/2)}\big).$$

3 $\big(k_{\mathsf{L}}^{(q)}, x^{(q)}\big) = \mathsf{f}\big(k_{\mathsf{L}}^{(q-1)}, x^{(q-1)}\big)$ if q is odd. (replace L's by R's if q is even)

$\widetilde{3}$ Resample $x^{(q)} \xleftarrow{\text{r}} \{0,1\}^n$

4 The adversary outputs the first bit of $\mathsf{C}_{\mathsf{A}}\big(|x\rangle\langle x| \otimes \rho_{YZ}^{(q-1)}\big)$.

Theorem 5.6.6. *Let $\varepsilon_{\mathsf{Ext}} = \varepsilon_{\mathsf{Prg}} = \varepsilon/4q$. There exists $t_{\mathsf{Prg}} = \mathrm{poly}(t, 2^{\ell}, 1/\varepsilon, q, n, k)$ such that if $\mathsf{Prg} : \{0,1\}^m \to \{0,1\}^{k+n}$ is an $(t_{\mathsf{Prg}}, \varepsilon_{\mathsf{Prg}})$-quantum$^+$-pseudorandom generator and $\mathsf{Ext} : \{0,1\}^{k+n} \to \{0,1\}^m$ is an $(\varepsilon_{\mathsf{Ext}}, k-\ell)$-quantum-proof extractor, then the above construction for SC is a $(t, \varepsilon, q, \ell)$-secure quantum leakage-resilient stream cipher.*

Proof. Let $t_{\mathsf{Prg}} = t + t_{\mathsf{SC}} + t_{\mathsf{C}}$ where t_{SC} is the circuit size of the q-round stream cipher and $t_{\mathsf{C}} = \mathrm{poly}(t, t_{\mathsf{SC}}, n, k, 2^{\ell}, q, 1/\varepsilon)$ is the circuit size of a "leakage simulator", which will be defined later.

We define following hybrid games (refer Game 5.6.2).

- For $i \in \{0\} \cup [q-1]$, Game G_i^q executes Step 2($\widetilde{\widetilde{\text{a}}}$) in the first i rounds, Step 2(a) afterward.

- For $i \in [q-1]$, Game $G_{i-1/2}^q$ executes Step 2($\widetilde{\widetilde{\text{a}}}$) in the first $i-1$ rounds, the Step 2($\widetilde{\text{a}}$) in the i-th round, and Step 2 afterward.

- For $i \in \{0\} \cup [q-1]$, Game $\widetilde{G}_i^q$ is identical to G_i^q, except it executes Step $\widetilde{3}$ instead.

- For $i \in [q-1]$, Game $\widetilde{G}_{i-1/2}^q$ is identical to $G_{i-1/2}^q$, except in the q round, it runs Step $(\widetilde{3})$ instead.

The goal is to show that for all adversary A of size t, $\left|\mathbb{E}[\mathsf{A}(G_0^q)] - \mathbb{E}[\mathsf{A}(\widetilde{G}_0^q)]\right| \leq \varepsilon$. We consider a sequence of games $G_0^q, G_{1/2}^q, \dots, G_{q-1}^q, \widetilde{G}_{q-1}^q, \widetilde{G}_{q-1-1/2}^q, \dots, \widetilde{G}_0^q$ and argue that every neighboring games are indistinguishable.

Intuitively, the game $G_{i-1/2}^q$ and G_i^q (similarly for $\widetilde{G}_{i-1/2}^q$ and $\widetilde{G}_i^q$) are indistinguishable since the output f is pseudorandom if the states k_{L} (or k_{R}) has high HILL min-entropy from the adversary's view. For the game G_{i-1}^q and $G_{i-1/2}^q$ (similarly for $\widetilde{G}_{i-1}^q$ and $\widetilde{G}_{i-1/2}^q$), they are indistinguishable due to the Quantum Leakage Simulation Lemma. The argument is formalized by the following two claims.

Claim 5.6.7. *For every $i \in [q-1]$ and every adversary A of size t, we have*

$$\left|\mathbb{E}[\mathsf{A}(G_{i-1}^q)] - \mathbb{E}[\mathsf{A}(G_{i-1/2}^q)]\right| \leq \frac{\varepsilon}{4q} \quad \text{and} \quad \left|\mathbb{E}[\mathsf{A}(\widetilde{G}_{i-1}^q)] - \mathbb{E}[\mathsf{A}(\widetilde{G}_{i-1/2}^q)]\right| \leq \frac{\varepsilon}{4q}\,.$$

Claim 5.6.8. *For every $i \in [q-1]$ and every adversary A of size t, we have*

$$\left|\mathbb{E}[\mathsf{A}(G_{i-1/2}^q)] - \mathbb{E}[\mathsf{A}(G_i^q)]\right| \leq \frac{\varepsilon}{4q} \quad \text{and} \quad \left|\mathbb{E}[\mathsf{A}(\widetilde{G}_{i-1/2}^q)] - \mathbb{E}[\mathsf{A}(\widetilde{G}_i^q)]\right| \leq \frac{\varepsilon}{4q}\,.$$

Also,

$$\left|\mathbb{E}[\mathsf{A}(G_{q-1}^q)] - \mathbb{E}[\mathsf{A}(\widetilde{G}_{q-1}^q)]\right| \leq \frac{\varepsilon}{4q}$$

By the claims above and a triangle inequality with absolute value, we conclude the theorem. □

Proof of Claim 5.6.7. Assume for contradiction, there exists $i \in [q-1]$ and an adversary adv of size t such that

$$\left|\mathbb{E}[\mathsf{A}(G^q_{i-1/2})] - \mathbb{E}[\mathsf{A}(G^q_i)]\right| > \frac{\varepsilon}{4q}\,.$$

We also assume i to be odd as the case of even i is symmetric. Consider the following quantum distinguisher D of size $t + 2t_{\mathsf{SC}}$ for $G^q_{i-1}(\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ})$ and $G^q_{i-1/2}(\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ})$ where $G^q(\rho)$ denotes the state ρ in the game G^q. For a given input $G^q_j(\rho^{(i-1/2)}_{XK_{\mathsf{L}}YZ})$ with $j = i-1$ or $i-1/2$, the distinguisher D simulates the game G^q_j starting from Step 2(b) in the i-th round by randomly sampling $k^{(i)}_{\mathsf{R}} \xleftarrow{r} \{0,1\}^k$ to form the state $\rho^{(i-1/2)}$ in G^q_j. Therefore, we have

$$\left|\mathsf{D}(G^q_{i-1/2}(\rho_{XK_{\mathsf{L}}YZ})) - \mathsf{D}(G^q_i(\rho_{XK_{\mathsf{L}}YZ}))\right| = \left|\mathbb{E}[\mathsf{A}(G^q_{i-1/2})] - \mathbb{E}[\mathsf{A}(G^q_i)]\right| > \frac{\varepsilon}{4q}\,,$$

which contradict the property we got from Step 2($\tilde{\text{b}}$). The proof for games $\widetilde{G}^q$'s is similar. □

Proof of Claim 5.6.8. Assume for contradiction there exists i and an adversary A of size t such that

$$\left|\mathbb{E}[A(G^q_{i-1/2})] - \mathbb{E}[A(G^q_i)]\right| > \frac{\varepsilon}{4q}\,.$$

We also assume i to be odd as the case of even i is completely symmetric. Consider the following quantum distinguisher D of size $t + t_{\mathsf{SC}} + t_{\mathsf{C}}$ for the states after updating k_{L}, x but before simulating the leakage in the i-th round of Step 2($\tilde{\text{a}}$) in $G^q_{i-1/2}$ and Step 2($\tilde{\tilde{\text{a}}}$) in G^q_i. Namely,

$$G^q_{i-1/2}(\rho^{(i-3/4)}) \stackrel{\text{def}}{=} \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}},y} p^{(i-1)}_{k_{\mathsf{L}}xk_{\mathsf{R}}y}\Big|\mathsf{f}(k_{\mathsf{L}},x),k_{\mathsf{R}},y\Big\rangle\Big\langle\,\mathsf{f}(k_{\mathsf{L}},x),k_{\mathsf{R}},y\Big| \otimes \rho_Z^{k_{\mathsf{L}}xk_{\mathsf{R}}y\,(i-1)} \quad \text{in} \quad G^q_{i-1/2} \text{ and}$$

$$G^q_i(\rho^{(i-3/4)}) \stackrel{\text{def}}{=} \sum_{k_{\mathsf{L}},x,k_{\mathsf{R}},y} 2^{-(k+n)} \cdot p^{(i-1)}_y\Big|k_{\mathsf{L}},x,k_{\mathsf{R}},y\Big\rangle\Big\langle k_{\mathsf{L}},x,k_{\mathsf{R}},y\Big| \otimes \rho_Z^{k_{\mathsf{L}}xk_{\mathsf{R}}y\,(i-1)} \quad \text{in} \quad G^q_i$$

$$G^q_{i-1/2}\Big(\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ}\Big) \quad \text{and} \quad G^q_i\Big(\rho^{(i-1/2)}_{K_{\mathsf{L}}XYZ}\Big)\,.$$

For a given input $G^q_j(\rho^{(i-3/4)})$ where $j = i-1/2$ or i, We simulate the game G^q_j starting from Step 2($\tilde{\text{a}}$) (or 2($\tilde{\tilde{\text{a}}}$)) of the i-th round, and output the result. To finish simulating the game, we need the simulating circuit C to get the state $G^q_j(\rho^{(i-1/2)})$. Therefore, the distinguisher D

is of size $t + t_{\mathsf{SC}} + t_{\mathsf{C}}$ and we have Then we have

$$\begin{aligned} &\left|\mathbb{E}[\mathsf{D}\left(G^q_{i-1/2}(\rho^{(i-3/4)})\right)] - \mathbb{E}[\mathsf{D}\left(G^q_i(\rho^{(i-3/4)})\right)]\right| \\ &\qquad\qquad\qquad \geq \left|\mathbb{E}[\mathsf{A}(G^q_{i-1/2})] - \mathbb{E}[\mathsf{A}(G^q_i)]\right| > \frac{\varepsilon}{4q} \,. \end{aligned}$$

Note that $G^q_{i-1}(\rho^{(i-3/4)}_{XK_{\mathsf{L}}})$ is in fact uniformly random. We will argue that $G^q_{i-1/2}(\rho^{(i-3/4)}_{XK_{\mathsf{L}}})$ is pseudorandom given $\rho^{(i-1)}_{YZ}$ (same in both Game $G^q_{i-1/2}$ and G^q_i) and yields a contradiction. In order to get the independence condition, we will actually prove that $G^q_{i-1}(\rho^{(i)}_{XK_{\mathsf{L}}})$ is pseudorandom given $\rho^{(j)}_{YZ}$ for all $j \in [i-1]$. By Claim 5.6.5, it suffices to show that

1. $K^{(i-1)}_{\mathsf{L}}$ and $X^{(i-1)}$ are independent given $\rho^{(j)}_{YZ}$ for all $j \in [i-1]$.

2. $\mathrm{H}_{\mathsf{min}}(K_{\mathsf{L}}|YZ)_{\rho^{(i-1)}} \geq k - \ell$.

The first condition can be obtained by observing that the only "dependence path" between $K^{(i-1)}_{\mathsf{L}}$ and $X^{(i-1)}$ is

$$K^{(i-1)}_{\mathsf{L}} \to \rho^{(i-2)}_{YZ} \to \rho^{(i-1)}_{YZ} \leftarrow X^{(i-1)} \,.$$

Note that only conditioning on $\rho^{(i-1)}_{YZ}$ is not sufficient for arguing the independence condition. The second condition is directly by the fact that $K^{(i-1)}_{\mathsf{L}}$ is uniform and the leakage chain rule for min-entropy (Theorem 5.5.12). □

5.7 Appendix

5.7.1 Pseudorandom states

Theorem 5.7.1. *For every $t, d \in \mathbb{N}, \varepsilon > 0$ such that $d \geq (t/\varepsilon)^4$, if we sample $|\psi\rangle$ uniformly at random from $\left\{\sum_{i=1}^d \alpha_i |i\rangle : \alpha_i \in \{\pm 1/\sqrt{d}\}\right\}$, then with all but $2^{-\Omega(\sqrt{d})}$ probability, $|\psi\rangle\langle\psi| \in$* Dens$(\mathbb{C}^d)$ *is an (t, ε)-quantum-pseudorandom pure state.*

Proof. Let $\mathsf{A} : \mathrm{Dens}(\mathbb{C}^d) \to \{0,1\}$ be any fixed quantum distinguisher and Π_A be the corresponding measurement operator. Then

$$\Pr\left[A(\rho^{\mathsf{mm}}_d) = 1\right] = \frac{1}{d}\left\langle \Pi_A \,,\, \mathbb{1}_d \right\rangle = \frac{1}{d}\sum_{i=1}^d \left\langle i \middle| \Pi_A \middle| i \right\rangle .$$

For a pure state $|\psi\rangle = \sum_{i=1}^{d} \alpha_i |i\rangle$, we have

$$\begin{aligned}\Pr[A(|\psi\rangle\langle\psi|) = 1] = \langle \Pi_A\,,\, |\psi\rangle\langle\psi|\rangle = \langle\psi|\Pi_A|\psi\rangle = \sum_{i,j\in[2^m]} \alpha_i^* \alpha_j \langle i|\Pi_A|j\rangle \\ = \sum_i |\alpha_i|^2 \langle i|\Pi_A|i\rangle + \sum_{i\neq j} \alpha_i^*\alpha_j\langle i|\Pi_A|j\rangle,\end{aligned}$$

Taking expectation over $|\psi\rangle \xleftarrow{r} \left\{\sum_{i=1}^{d} \alpha_i|i\rangle : \alpha_i \in \{\pm 1/\sqrt{d}\}\right\}$, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{|\psi\rangle}[\Pr[A(|\psi\rangle\langle\psi|) = 1]] =& \mathop{\mathbb{E}}_{|\psi\rangle}[\sum_i |\alpha_i|^2\langle i|\Pi_A|i\rangle] + \mathop{\mathbb{E}}_{|\psi\rangle}[\sum_{i\neq j}\alpha_i^*\alpha_j\langle i|\Pi_A|j\rangle] \\ =& \frac{1}{2^m}\sum_i \langle i|\Pi_A|i\rangle = \Pr[A(\rho_d^{\mathsf{mm}}) = 1].\end{aligned}$$

For a fixed distinguisher A, define the function $f : \left\{\sum_{i=1}^{d} \alpha_i|i\rangle : \alpha_i \in \{\pm 1/\sqrt{d}\}\right\} \to [0,1]$ as

$$f(|\psi\rangle) = \Pr[A(|\psi\rangle\langle\psi|) = 1] = \langle\psi|\Pi_A|\psi\rangle.$$

Now we are going to show the concentration using Talagrand's inequality. To that end, we first find the *Lipschitz constant* η of the function f. For all $|\psi\rangle, |\phi\rangle \in \mathrm{Ball}(\mathbb{C}^d)$, we have

$$\begin{aligned}|f(|\psi\rangle) - f(|\phi\rangle)| =& |\langle\psi|\Pi_A|\psi\rangle - \langle\phi|\Pi_A|\phi\rangle| \\ \leq& |\langle\psi|\Pi_A|\psi\rangle - \langle\psi|\Pi_A|\phi\rangle| + |\langle\phi|\Pi_A|\psi\rangle - \langle\phi|\Pi_A|\phi\rangle| \\ \leq& \|\langle\psi|\Pi_A\|_2 \cdot \||\psi\rangle - |\phi\rangle\|_2 + \|\langle\psi| - \langle\phi|\|_2 \cdot \|\Pi_A|\phi\rangle\|_2 \\ \leq& 2\cdot\|\Pi_A\|_{\mathsf{op}}\||\psi\rangle - |\phi\rangle\|_2 \quad \leq 2\cdot\||\psi\rangle - |\phi\rangle\|_2,\end{aligned}$$

where the second inequality follows from the Cauchy-Schwartz inequality, and the last inequality follows because $0 \leq \Pi_A \leq \mathbb{1}_d$ Therefore the Lipschitz constant η of the function f is at most 2. Also, f is a convex function:

$$\begin{aligned}f\left(\frac{|\psi\rangle + |\phi\rangle}{2}\right) \leq& f\left(\frac{|\psi\rangle + |\phi\rangle}{2}\right) + f\left(\frac{|\psi\rangle - |\phi\rangle}{2}\right) \\ =& \frac{1}{2}(\langle\psi|\Pi|\psi\rangle + \langle\phi|\Pi|\phi\rangle) = \frac{1}{2}(f(|\psi\rangle) + f(|\phi\rangle))\,.\end{aligned}$$

Now we are ready to apply the Talagrand's concentration inequality [Tal95]: If f is an η-Lipschitz convex function on the hypercube $H = \{\pm K\}^d$, D is a product distribution on H, and $\mu = \mathbb{E}_{|\psi\rangle \xleftarrow{\mathrm{r}} D}[f(|\psi\rangle)]$, then for all $t > 0$, we have

$$\Pr_{|\psi\rangle \xleftarrow{\mathrm{r}} D}\Big[|f(|\psi\rangle) - \mu| \geq Kt\Big] \leq 2^{-\Omega(t^2/\eta^2)} .$$

Taking $K = 1/\sqrt{d}$, $t = \sqrt{d}\varepsilon$ and $\eta = 2$, and D to be uniform on $\{\pm K\}^d$, we have

$$\Pr_{|\psi\rangle}\Big[\Big|\Pr[A(|\psi\rangle\langle\psi|) = 1] - \Pr[A(\rho_d^{\mathsf{mm}}) = 1]\Big| \geq \varepsilon\Big] = \Pr_{|\psi\rangle}\Big[|f(|\psi\rangle) - \mu| \geq \varepsilon\Big] \leq 2^{-\Omega(d\varepsilon^2)}.$$

There are only $s^{O(s)} = 2^{O(s \log s)}$ many different quantum circuits of size s. By a union bound,

$$\Pr_{|\psi\rangle}[\exists A \text{ of size } s, \Big|\Pr[A(|\psi\rangle\langle\psi|) = 1] - \Pr[A(\rho_d^{\mathsf{mm}}) = 1]\Big| \geq \varepsilon] \leq 2^{O(s \log s)} \cdot 2^{-\Omega(d\varepsilon^2)} \leq 2^{-\Omega(\sqrt{d})}$$

provided that $d \geq (s/\varepsilon)^4$. □

An interesting follow-up question is that whether we can *explicitly* generate pseudorandom pure states, say as the output of a small quantum circuit (with no measurements) on input $|0^n\rangle$ — which we could think of as a "seedless" pseudorandom generator. If the generator is of polynomial size, then its output cannot be pseudorandom against all polynomial-sized distinguishers, because (measurement-free) quantum computation is reversible. But if we allow the generator circuit to be larger than the distinguishers then it is conceivable to have a pseudorandom pure state as output. As aforementioned, in [BaHH16a, BaHH16b], they use probabilistic method to show the existence of a generator circuit of size n^{11k+9} that can fool all n^k-size quantum distinguishers. It would be interesting to construct such generators explicitly under plausible (quantum) complexity assumptions.

Pseudorandom state against quantum circuit with quantum advice

In the classical setting, a well known result in pseudorandomness is that if we randomly choose $2^{\omega(\log n)}$ elements from $\{0,1\}^n$ to form a set S where n is the security parameter. Then with high probability, the set S is a pseudorandom set. Now we show that this phenomenon can be extended to quantum distinguishers with quantum advice.

Theorem 5.7.2. *There exists a set $S \subseteq \{0,1\}^n$ with $|S| = O\big((t \log t + \log d')/\varepsilon^2\big)$ such that U_S is (t, ε)-quantum$^+$-indistinguishable where U_S is the uniform distribution over the set S.*

The canonical proof in the classical case is that for a fixed circuit, the random S is pseudorandom with overwhelming high probability by Chernoff bound. Then by union bound over all bounded size circuits, with high probability, the random S is pseudorandom against all bounded size circuits at the same time. However, in the quantum case, if we include the quantum advice there are infinitely many bounded size quantum circuits. Therefore, we cannot union bound over all of them. Here we will prove it again using the Rademacher complexity.

Proof. Let $2^n = d$. We consider a fixed quantum distinguisher $C : \mathrm{Dens}(\mathbb{C}^{d \times d'}) \to \{0,1\}$ with a d'-dimensional advice input. For every $x \in \{0,1\}^n$, we define the POVM Π_x to be the corresponding measurement operator of $C(x;\cdot)$. Then we have that $\Pr[C(x;\tau)] = \langle \Pi_x \,, \tau \rangle$. Define the class of function

$$\mathcal{F} \overset{\text{def}}{=} \left\{ C(\cdot;\tau) = \langle \,, \tau \rangle : \tau \in \mathrm{Dens}(\mathbb{C}^{d'}) \right\} .$$

By Theorem 5.5.2, we know that the Rademacher complexity $\mathfrak{R}_T(\mathcal{F})$ is $O\big(\sqrt{\log d' T}\big)$. Therefore, by the generalization bound (Theorem 5.3.10), we know that for every $\delta > 0$, if we sample $x_1, \ldots, x_T$ uniformly at random from $\{0,1\}^n$, then with $1-\delta$ probability,

$$\forall \tau \,, \left| \mathop{\mathbb{E}}_{x \overset{\text{r}}{\leftarrow} \{0,1\}^n} [C(x;\tau)] - \frac{1}{T} \sum_{i=1}^{T} C(x_i;\tau) \right| \le O\big(\sqrt{\log d'/T}\big) + O\big(\sqrt{\log(1/\delta)/T}\big). \tag{5.17}$$

Choose $\delta = 1/t^{O(t)}$ such that $1/\delta$ is more than the number of circuits of size at most t, and $T = O\big(\log(1/\delta)/\varepsilon^2\big)$. Then we have that if we randomly choose a set $S \subseteq \{0,1\}^n$ of size T, then with $1-\delta$ probability,

$$\forall \tau \,, \left| \mathop{\mathbb{E}}_{x \overset{\text{r}}{\leftarrow} \{0,1\}^n} [C(x;\tau)] - \mathop{\mathbb{E}}_{x \overset{\text{r}}{\leftarrow} U_S} [C(x;\tau)] \right| \le \delta \,.$$

By union bound over all circuits of size at most t, we conclude the theorem. □

5.7.2 Barrier for gap amplification

One of the main challenges in extending classical proofs to quantum cases is the celebrated Wootters-Zurek no-cloning theorem [WZ82]. Here we exhibit another barrier — the *gap* *amplification problem* defined as follows. Given a quantum distinguisher A (whose input is a quantum state ρ), where the acceptance probability is greater than p for YES instances and less than q for NO instances, can we have another quantum distinguisher A' where the gap $p' - q'$ is larger than that in A? If we were able to clone an arbitrary quantum state, then the gap amplification would be easy (as discussed below). Thus, we can view the gap amplification problem as a special case of the no-cloning theorem. Moreover, we will show that the impossibility of amplifying the gap implies that imperfect cloning of a single qubit to within a constant in trace distance is impossible.

In the classical case, gap amplification demonstrate the robustness of the definition of the complexity class **BPP** in the that no matter what the constant we use for acceptance probabilities for YES and NO instances, the definitions for **BPP** are equivalent. Similarly, in the quantum setting, the gap amplification problem is connected to the amplification of the acceptance probability of quantum proofs in **QMA**. The gap amplification problem is trivial in the classical case, as there is no cloning restriction in the classical world. For a given input, we can make copies of the input, run the original algorithm multiple times, and then use a majority or threshold rule to reduce the error probability, as follows from a concentration bound (e.g., a Chernoff bound). However, in the quantum case, due to the no-cloning theorem, we cannot follow this strategy directly. Note that the no-cloning theorem does not directly imply the impossibility of amplification,

First, we define the gap amplification problem as follows.

Definition 5.7.3 (Gap-Amplification Problem)**.** *Let* $\mathsf{D} : \mathrm{Dens}(\mathbb{C}^M) \to \{0,1\}$ *be q quantum distinguisher,* $0 < q < p < 1$*. We say that a quantum distinguisher* $\mathsf{D}' : \mathrm{Dens}(\mathbb{C}^M) \to \{0,1\}$ *is a* (p,q)-amplified version *of* D *if for every input* $|\psi\rangle \in \mathrm{Ball}(\mathbb{C}^M)$*,*

$$\begin{cases} \Pr\Big[\mathsf{D}(|\psi\rangle\langle\psi|) = 1\Big] \geq p & \Rightarrow \quad \Pr\Big[\mathsf{D}'(|\psi\rangle\langle\psi|) = 1\Big] > p \\ \Pr\Big[\mathsf{D}(|\psi\rangle\langle\psi|) = 1\Big] \leq p & \Rightarrow \quad \Pr\Big[\mathsf{D}'(|\psi\rangle\langle\psi|) = 1\Big] < p\,. \end{cases}$$

We show that such amplification is impossible in general.

Theorem 5.7.4. *For every real numbers* $0 < q < p < 1$*, there exists a quantum distinguisher*

$\mathsf{D} : \mathrm{Dens}(\mathbb{C}^2) \to \{0,1\}$ *such that no* (p,q)*-amplified version of* D *exists.*

Proof. Let A be a single-qubit measurement in the computational basis $\{|0\rangle, |1\rangle\}$. Consider the pure states $|\psi\rangle = (\cos\alpha)|0\rangle + (\sin\alpha)|1\rangle$ and $|\phi\rangle = (\cos\beta)|0\rangle + (\sin\beta)|1\rangle$, where $\alpha = \sin^{-1}(\sqrt{p})$ and $\beta = \sin^{-1}(\sqrt{q})$. Thus $\Pr[A(|\psi\rangle) = 1] = p$ and $\Pr[A(|\phi\rangle) = 1] = q$.

Let the BPOVM of A' be $\Pi = \begin{bmatrix} a & -b+ci \\ -b-ci & d \end{bmatrix}$ for $0 \le a, d \le 1$ and appropriate real numbers b and c such that that $\Pi \ge 0$. Assume that A' is a (p,q)-amplified version of A such that $\langle \Pi, |\psi\rangle\langle\psi| \rangle > \sin^2\alpha$ and $\langle \Pi, |\phi\rangle\langle\phi| \rangle < \sin^2\beta$. That is,

$$a\cos^2\alpha - 2b\sin\alpha\cos\alpha + d\sin^2\alpha > \sin^2\alpha,$$
$$a\cos^2\beta - 2b\sin\beta\cos\beta + d\sin^2\beta < \sin^2\beta.$$

After dividing the two inequalities by $\sin^2\alpha$ and $\sin^2\beta$, respectively, we obtain

$$a\cot^2\alpha + d > 1 + 2b\cot\alpha, \tag{5.18}$$
$$a\cot^2\beta + d < 1 + 2b\cot\beta. \tag{5.19}$$

Since $d \le 1$, we have $a > \frac{2b}{\cot\alpha}$ by Equation (5.18). On the other hand, subtracting Equation (5.18) from Equation (5.19) and dividing it by $(\cot\beta - \cot\alpha)$, which is positive since $\sin\alpha = \sqrt{p} > \sqrt{q} > \sin\beta$, we get $a < \frac{2b}{\cot\beta + \cot\alpha} < \frac{2b}{\cot\alpha}$. That gives a contradiction.

□

Chapter 6

Conclusion

In this thesis, we have seen that computational entropies are exceptionally useful in the constructions of basic cryptographic primitives, including pseudorandom generators, universal one-way hash functions, and statistically hiding commitment schemes, from one-way functions. In fact, some definitions of computational entropies are motivated by seeking to make such constructions more efficient. For constructing pseudorandom generators, even though a series of works has made a great improvement in efficiency, there is still a gap between the upper and lower bounds. Our lower bound for flattening entropies can be viewed as a step towards closing the gap and may help in understanding the construction of those cryptographic primitives. On the other hand, the construction of universal one-way hash functions is still relatively inefficient. Our new notion for exploring the computational hardness inside one-way functions provides a more modular way to obtain an inaccessible entropy, which may be useful for further simplifying and improving the construction of universal one-way hash functions.

We have also initiated the study of computational entropies in the quantum setting. Most of the positive results can be proved via the Non-uniform Quantum Min-max Theorems, which we developed using the generalization bound for Rademacher complexity. A notable one is the Quantum Leakage Simulation Lemma, which has applications in leakage-resilient cryptography. On the other hand, we also show the natural quantum extensions of some classical theorems about computational entropy do not hold. Interestingly, one of them is due to the existence of pseudorandom pure states. Roughly speaking, when quantum states show up as side information, as in post-quantum cryptography, there are more chances that results can be generalized. Otherwise, most of the barriers for proving desirable theorems are due to the no-cloning theorem or more precisely, the hardness of gap amplification. We expect that computational notions of quantum entropy will find other natural applications in quantum cryptography. Also, studying their fundamental properties may provide new insights in quantum complexity theory.

References

[Aar07] Aaronson Scott. The learnability of quantum states. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 463(2088):3089–3114, December 2007.

[ACHV19] Rohit Agrawal, Yi-Hsiu Chen, Thibaut Horel, and Salil Vadhan. Unifying computational entropies via Kullback-Leibler divergence. Technical Report 264, 2019.

[AHK12] Sanjeev Arora, Elad Hazan, and Satyen Kale. The Multiplicative Weights Update Method: A Meta-Algorithm and Applications. *Theory of Computing*, 8(1):121–164, May 2012.

[AK07] Scott Aaronson and Greg Kuperberg. Quantum Versus Classical Proofs and Advice. *Theory of Computing*, 3(1):129–157, September 2007.

[AS14] Sergei Artemenko and Ronen Shaltiel. Lower Bounds on the Query Complexity of Non-uniform and Adaptive Reductions Showing Hardness Amplification. *computational complexity*, 23(1):43–83, March 2014.

[BaHH16a] Fernando G. S. L. Brandão, Aram W. Harrow, and Michał Horodecki. Efficient Quantum Pseudorandomness. *Physical Review Letters*, 116(17):170502, April 2016.

[BaHH16b] Fernando G. S. L. Brandão, Aram W. Harrow, and Michał Horodecki. Local Random Quantum Circuits are Approximate Polynomial-Designs. *Communications in Mathematical Physics*, 346(2):397–434, September 2016.

[BB14] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, December 2014.

[BCC88] Gilles Brassard, David Chaum, and Claude Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, 37(2):156–189, October 1988.

[BDF$^+$11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random Oracles in a Quantum World. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, Lecture Notes in Computer Science, pages 41–69. Springer Berlin Heidelberg, 2011.

[BDJR97] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*, pages 394–403, October 1997.

[BM82] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo random bits. In *23rd Annual Symposium on Foundations of Computer Science (Sfcs 1982)*, pages 112–117, November 1982.

[BM02] Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian Complexities: Risk Bounds and Structural Results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

[BMW09] Michael J. Bremner, Caterina Mora, and Andreas Winter. Are Random Pure States Useful for Quantum Computation? *Physical Review Letters*, 102(19):190502, May 2009.

[BPR12] Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom Functions and Lattices. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, Lecture Notes in Computer Science, pages 719–737. Springer Berlin Heidelberg, 2012.

[BSW03] Boaz Barak, Ronen Shaltiel, and Avi Wigderson. Computational Analogues of Entropy. In Sanjeev Arora, Klaus Jansen, José D. P. Rolim, and Amit Sahai, editors, *Approximation, Randomization, and Combinatorial Optimization.. Algorithms and Techniques*, Lecture Notes in Computer Science, pages 200–215. Springer Berlin Heidelberg, 2003.

[CA97] N. J. Cerf and C. Adami. Negative Entropy and Information in Quantum Mechanics. *Physical Review Letters*, 79(26):5194–5197, December 1997.

[CCL$^+$17] Yi-Hsiu Chen, Kai-Min Chung, Ching-Yi Lai, Salil P. Vadhan, and Xiaodi Wu. Computational Notions of Quantum Min-Entropy. *arXiv:1704.07309 [quant-ph]*, April 2017.

[CCL18] Yi-Hsiu Chen, Kai-Min Chung, and Jyun-Jie Liao. On the Complexity of Simulating Auxiliary Input. In Jesper Buus Nielsen and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2018*, Lecture Notes in Computer Science, pages 371–390. Springer International Publishing, 2018.

[CGVZ18] Yi-Hsiu Chen, Mika Göös, Salil P. Vadhan, and Jiapeng Zhang. A Tight Lower Bound for Entropy Flattening. In Rocco A. Servedio, editor, *33rd Computational Complexity Conference (CCC 2018)*, volume 102 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 23:1–23:28, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[CHY15] Hao-Chung Cheng, Min-Hsiu Hsieh, and Ping-Cheng Yeh. The Learnability of Unknown Quantum Measurements. *arXiv:1501.00559 [quant-ph, stat]*, January 2015.

[CKLR11] Kai-Min Chung, Yael Tauman Kalai, Feng-Hao Liu, and Ran Raz. Memory Delegation. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, Lecture Notes in Computer Science, pages 151–168. Springer Berlin Heidelberg, 2011.

[CLP15] Kai-Min Chung, Edward Lui, and Rafael Pass. From Weak to Strong Zero-Knowledge and Applications. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 66–92. Springer Berlin Heidelberg, 2015.

[CSW14] Kai-Min Chung, Yaoyun Shi, and Xiaodi Wu. Physical Randomness Extractors: Generating Random Numbers with Minimal Assumptions. *arXiv:1402.4797 [quant-ph]*, February 2014.

[DFSS08] I. DamgÅrd, S. Fehr, L. Salvail, and C. Schaffner. Cryptography in the Bounded-Quantum-Storage Model. *SIAM Journal on Computing*, 37(6):1865–1890, January 2008.

[DH76] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, November 1976.

[DHRS07] Yan Zong Ding, Danny Harnik, Alon Rosen, and Ronen Shaltiel. Constant-Round Oblivious Transfer in the Bounded Storage Model. *Journal of Cryptology*, 20(2):165–202, April 2007.

[DKM$^+$06] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, Lecture Notes in Computer Science, pages 486–503. Springer Berlin Heidelberg, 2006.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 265–284. Springer Berlin Heidelberg, 2006.

[DORS08] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. *SIAM Journal on Computing*, 38(1):97–139, January 2008.

[DP08] S. Dziembowski and K. Pietrzak. Leakage-Resilient Cryptography. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 293–302, October 2008.

[DPVR12] A. De, C. Portmann, T. Vidick, and R. Renner. Trevisan's Extractor in the Presence of Quantum Side Information. *SIAM Journal on Computing*, 41(4):915–940, January 2012.

[FK99] Alan Frieze and Ravi Kannan. Quick Approximation to Matrices and Applications. *Combinatorica*, 19(2):175–220, February 1999.

[FOR15] Benjamin Fuller, Adam O'Neill, and Leonid Reyzin. A Unified Approach to Deterministic Encryption: New Constructions and a Connection to Computational Entropy. *Journal of Cryptology*, 28(3):671–717, July 2015.

[FR12] Benjamin Fuller and Leonid Reyzin. Computational Entropy and Information Leakage. Technical Report 466, 2012.

[Fre95] Y. Freund. Boosting a Weak Learning Algorithm by Majority. *Information and Computation*, 121(2):256–285, September 1995.

[GFE09] D. Gross, S. T. Flammia, and J. Eisert. Most Quantum States Are Too Entangled To Be Useful As Computational Resources. *Physical Review Letters*, 102(19):190501, May 2009.

[GGKT05] R. Gennaro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the Efficiency of Generic Cryptographic Constructions. *SIAM Journal on Computing*, 35(1):217–246, January 2005.

[GGM84] O. Goldreich, S. Goldwasser, and S. Micali. How To Construct Random Functions. In *25th Annual Symposium onFoundations of Computer Science, 1984.*, pages 464–479, October 1984.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to Construct Random Functions. *J. ACM*, 33(4):792–807, August 1986.

[GL89] O. Goldreich and L. A. Levin. A Hard-core Predicate for All One-way Functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 25–32, New York, NY, USA, 1989. ACM.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, April 1984.

[GMW86] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In *27th Annual Symposium on Foundations of Computer Science (Sfcs 1986)*, pages 174–187, October 1986.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson. How to Play ANY Mental Game. In *Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing*, STOC '87, pages 218–229, New York, NY, USA, 1987. ACM.

[GMW91] O. Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):690–728, July 1991.

[GNW11] Oded Goldreich, Noam Nisan, and Avi Wigderson. On Yao's XOR-Lemma. In Oded Goldreich, editor, *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation: In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan,* *Salil Vadhan, Avi Wigderson, David Zuckerman*, Lecture Notes in Computer Science, pages 273–301. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[GSV99a] Oded Goldreich, Amit Sahai, and Salil Vadhan. Can Statistical Zero Knowledge Be Made Non-interactive? or On the Relationship of SZK and NISZK. In Michael Wiener, editor, *Advances in Cryptology — CRYPTO' 99*, Lecture Notes in Computer Science, pages 467–484. Springer Berlin Heidelberg, 1999.

[GSV99b] Oded Goldreich, Amit Sahai, and Salil Vadhan. Can Statistical Zero Knowledge be made Non-Interactive? or On the Relationship of SZK and NISZK. Technical Report 013, 1999.

[GT08] Ben Green and Terence Tao. The Primes Contain Arbitrarily Long Arithmetic Progressions. *Annals of Mathematics*, 167(2):481–547, 2008.

[GW11] Craig Gentry and Daniel Wichs. Separating Succinct Non-interactive Arguments from All Falsifiable Assumptions. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, pages 99–108, New York, NY, USA, 2011. ACM.

[Has90] J. Hastad. Pseudo-random Generators Under Uniform Assumptions. In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing*, STOC '90, pages 395–404, New York, NY, USA, 1990. ACM.

[HHJ$^+$16] Jeongwan Haah, Aram W. Harrow, Zhengfeng Ji, Xiaodi Wu, and Nengkun Yu. Sample-optimal Tomography of Quantum States. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16, pages 913–925, New York, NY, USA, 2016. ACM.

[HHR06] Iftach Haitner, Danny Harnik, and Omer Reingold. Efficient Pseudorandom Generators from Exponentially Hard One-Way Functions. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, Lecture Notes in Computer Science, pages 228–239. Springer Berlin Heidelberg, 2006.

[HHR$^+$10] Iftach Haitner, Thomas Holenstein, Omer Reingold, Salil Vadhan, and Hoeteck Wee. Universal One-Way Hash Functions via Inaccessible Entropy. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, Lecture Notes in Computer Science, pages 616–637. Springer Berlin Heidelberg, 2010.

[HILL99] J. Håstad, R. Impagliazzo, L. Levin, and M. Luby. A Pseudorandom Generator from any One-way Function. *SIAM Journal on Computing*, 28(4):1364–1396, January 1999.

[HJ12] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, October 2012.

[HLR07] Chun-Yuan Hsiao, Chi-Jen Lu, and Leonid Reyzin. Conditional Computational Entropy, or Toward Separating Pseudoentropy from Compressibility. In Moni

Naor, editor, *Advances in Cryptology - EUROCRYPT 2007*, Lecture Notes in Computer Science, pages 169–186. Springer Berlin Heidelberg, 2007.

[HNO$^+$09] I. Haitner, M. Nguyen, S. Ong, O. Reingold, and S. Vadhan. Statistically Hiding Commitments and Statistical Zero-Knowledge Arguments from Any One-Way Function. *SIAM Journal on Computing*, 39(3):1153–1218, January 2009.

[Hol05] Thomas Holenstein. Key Agreement from Weak Bit Agreement. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 664–673, New York, NY, USA, 2005. ACM.

[Hol06] Thomas Holenstein. Pseudorandom Generators from One-Way Functions: A Simple Construction for Any Hardness. In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 443–461. Springer Berlin Heidelberg, 2006.

[HR11] T. Holenstein and R. Renner. On the Randomness of Independent Experiments. *IEEE Transactions on Information Theory*, 57(4):1865–1871, April 2011.

[HRV10] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency Improvements in Constructing Pseudorandom Generators from One-way Functions. In *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, STOC '10, pages 437–446, New York, NY, USA, 2010. ACM.

[HRV13] I. Haitner, O. Reingold, and S. Vadhan. Efficiency Improvements in Constructing Pseudorandom Generators from One-Way Functions. *SIAM Journal on Computing*, 42(3):1405–1430, January 2013.

[HRVW09] Iftach Haitner, Omer Reingold, Salil Vadhan, and Hoeteck Wee. Inaccessible Entropy. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC '09, pages 611–620, New York, NY, USA, 2009. ACM.

[HRVW19] Iftach Haitner, Omer Reingold, Salil Vadhan, and Hoeteck Wee. Inaccessible Entropy I: Inaccessible Entropy Generators and Statistically Hiding Commitments from One-Way Functions. page 57, 2019.

[HS12] T. Holenstein and M. Sinha. Constructing a Pseudorandom Generator Requires an Almost Linear Number of Calls. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 698–707, October 2012.

[HV17] Iftach Haitner and Salil Vadhan. The Many Entropies in One-Way Functions. In Yehuda Lindell, editor, *Tutorials on the Foundations of Cryptography: Dedicated to Oded Goldreich*, Information Security and Cryptography, pages 159–217. Springer International Publishing, Cham, 2017.

[IL89] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *30th Annual Symposium on Foundations of Computer Science*, pages 230–235, October 1989.

[ILL89] R. Impagliazzo, L. A. Levin, and M. Luby. Pseudo-random Generation from One-way Functions. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 12–24, New York, NY, USA, 1989. ACM.

[Imp95] R. Impagliazzo. Hard-core distributions for somewhat hard problems. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 538–545, October 1995.

[JP14] Dimitar Jetchev and Krzysztof Pietrzak. How to Fake Auxiliary Input. In Yehuda Lindell, editor, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 566–590. Springer Berlin Heidelberg, 2014.

[KK05] Jonathan Katz and Chiu-Yuen Koo. On Constructing Universal One-Way Hash Functions from Arbitrary One-Way Functions. Technical Report 328, 2005.

[KPWW16] Stephan Krenn, Krzysztof Pietrzak, Akshay Wadia, and Daniel Wichs. A counterexample to the chain rule for conditional HILL entropy. *computational complexity*, 25(3):567–605, September 2016.

[KR19] Yael Tauman Kalai and Leonid Reyzin. A Survey of Leakage-Resilient Cryptography. Technical Report 302, 2019.

[KRS09] R. Konig, R. Renner, and C. Schaffner. The Operational Meaning of Min- and Max-Entropy. *IEEE Transactions on Information Theory*, 55(9):4337–4347, September 2009.

[KT08] R. T. Konig and B. M. Terhal. The Bounded-Storage Model in the Presence of a Quantum Adversary. *IEEE Transactions on Information Theory*, 54(2):749–762, February 2008.

[LFC$^+$09] J. S. Lundeen, A. Feito, H. Coldenstrodt-Ronge, K. L. Pregnell, Ch Silberhorn, T. C. Ralph, J. Eisert, M. B. Plenio, and I. A. Walmsley. Tomography of quantum detectors. *Nature Physics*, 5(1):27–30, January 2009.

[LTW11] Chi-Jen Lu, Shi-Chun Tsai, and Hsin-Lung Wu. Complexity of Hard-Core Set Proofs. *computational complexity*, 20(1):145–171, March 2011.

[Lub94] Michael George Luby. *Pseudorandomness and Cryptographic Applications*. Princeton University Press, Princeton, NJ, USA, 1994.

[LZ17] Shachar Lovett and Jiapeng Zhang. On the Impossibility of Entropy Reversal, and Its Application to Zero-Knowledge Proofs. In Yael Kalai and Leonid Reyzin, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 31–55. Springer International Publishing, 2017.

[Mon15] Montanaro Ashley. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2181):20150301, September 2015.

[MPRV09] Ilya Mironov, Omkant Pandey, Omer Reingold, and Salil Vadhan. Computational Differential Privacy. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009*, Lecture Notes in Computer Science, pages 126–142. Springer Berlin Heidelberg, 2009.

[MR04] Silvio Micali and Leonid Reyzin. Physically Observable Cryptography. In Moni Naor, editor, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 278–296. Springer Berlin Heidelberg, 2004.

[MS16] Carl A. Miller and Yaoyun Shi. Robust Protocols for Securely Expanding Randomness and Distributing Keys Using Untrusted Quantum Devices. *J. ACM*, 63(4):33:1–33:63, October 2016.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, January 1991.

[NC02] Michael A. Nielsen and Isaac Chuang. Quantum Computation and Quantum Information. *American Journal of Physics*, 70(5):558–559, April 2002.

[NOVY98] Moni Naor, Rafail Ostrovsky, Ramarathnam Venkatesan, and Moti Yung. Perfect Zero-Knowledge Arguments for NP Using Any One-Way Permutation. *Journal of Cryptology*, 11(2):87–108, March 1998.

[NV06] Minh-Huyen Nguyen and Salil Vadhan. Zero Knowledge with Efficient Provers. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '06, pages 287–295, New York, NY, USA, 2006. ACM.

[NY89] M. Naor and M. Yung. Universal One-way Hash Functions and Their Cryptographic Applications. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, pages 33–43, New York, NY, USA, 1989. ACM.

[NY04] Harumichi Nishimura and Tomoyuki Yamakami. Polynomial time quantum computation with advice. *Information Processing Letters*, 90(4):195–204, May 2004.

[NZ96] Noam Nisan and David Zuckerman. Randomness is Linear in Space. *Journal of Computer and System Sciences*, 52(1):43–52, February 1996.

[Oka00] Tatsuaki Okamoto. On Relationships between Statistical Zero-Knowledge Proofs. *Journal of Computer and System Sciences*, 60(1):47–108, February 2000.

[OV08] Shien Jin Ong and Salil Vadhan. An Equivalence Between Zero Knowledge and Commitments. In Ran Canetti, editor, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 482–500. Springer Berlin Heidelberg, 2008.

[Pie09] Krzysztof Pietrzak. A Leakage-Resilient Mode of Operation. In Antoine Joux, editor, *Advances in Cryptology - EUROCRYPT 2009*, Lecture Notes in Computer Science, pages 462–482. Springer Berlin Heidelberg, 2009.

[PS16] Krzysztof Pietrzak and Maciej Skórski. Pseudoentropy: Lower-Bounds for Chain Rules and Transformations. In Martin Hirt and Adam Smith, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 183–203. Springer Berlin Heidelberg, 2016.

[Ren08] Renato Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 06(01):1–127, February 2008.

[Rey11] Leonid Reyzin. Some Notions of Entropy for Cryptography. In Serge Fehr, editor, *Information Theoretic Security*, Lecture Notes in Computer Science, pages 138–142. Springer Berlin Heidelberg, 2011.

[Rom90] J. Rompel. One-way Functions Are Necessary and Sufficient for Secure Signatures. In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing*, STOC '90, pages 387–394, New York, NY, USA, 1990. ACM.

[RTTV08] O. Reingold, L. Trevisan, M. Tulsiani, and S. Vadhan. Dense Subsets of Pseudorandom Sets. In *2008 49th Annual IEEE Symposium on Foundations of Computer Science*, pages 76–85, October 2008.

[RW04] R. Renner and S. Wolf. Smooth Renyi entropy and applications. In *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, pages 233–, June 2004.

[RW05] Renato Renner and Stefan Wolf. Simple and Tight Bounds for Information Reconciliation and Privacy Amplification. In Bimal Roy, editor, *Advances in Cryptology - ASIACRYPT 2005*, Lecture Notes in Computer Science, pages 199–216. Springer Berlin Heidelberg, 2005.

[RW16] Tim Roughgarden and Joshua R. Wang. Minimizing Regret with Multiple Reserves. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 601–616, New York, NY, USA, 2016. ACM.

[SBM05] Vivek V. Shende, Stephen S. Bullock, and Igor L. Markov. Synthesis of Quantum Logic Circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, ASP-DAC '05, pages 272–275, New York, NY, USA, 2005. ACM.

[Sha49] C. E. Shannon. Communication Theory of Secrecy Systems*. *Bell System Technical Journal*, 28(4):656–715, 1949.

[Sko16a] Maciej Skorski. Simulating Auxiliary Inputs, Revisited. In Martin Hirt and Adam Smith, editors, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 159–179. Springer Berlin Heidelberg, 2016.

[Sko16b] Maciej Skorski. A Subgradient Algorithm For Computational Distances and Applications to Cryptography. Technical Report 158, 2016.

[Sko17] Maciej Skorski. *Approximating Min-Max Strategies by Statistical Learning.* February 2017.

[Son14] Fang Song. A Note on Quantum Security for Post-Quantum Cryptography. In Michele Mosca, editor, *Post-Quantum Cryptography*, Lecture Notes in Computer Science, pages 246–265. Springer International Publishing, 2014.

[SV97] A. Sahai and S. P. Vadhan. A complete promise problem for statistical zero-knowledge. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*, pages 448–457, October 1997.

[SV10] R. Shaltiel and E. Viola. Hardness Amplification Proofs Require Majority. *SIAM Journal on Computing*, 39(7):3122–3154, January 2010.

[Tal95] Michel Talagrand. Concentration of measure and isoperimetric inequalities in product spaces. *Publications Mathématiques de l'Institut des Hautes Études Scientifiques*, 81(1):73–205, December 1995.

[Tre99] Luca Trevisan. Extractors and Pseudorandom Generators. *Journal of the ACM*, 48:2001, 1999.

[Tre11] Luca Trevisan. Dense Model Theorems and Their Applications. In Yuval Ishai, editor, *Theory of Cryptography*, Lecture Notes in Computer Science, pages 55–57. Springer Berlin Heidelberg, 2011.

[TTV09] L. Trevisan, M. Tulsiani, and S. Vadhan. Regularity, Boosting, and Efficiently Simulating Every High-Entropy Distribution. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 126–136, July 2009.

[TZ08] Terence Tao and Tamar Ziegler. The primes contain arbitrarily long polynomial progressions. *Acta Mathematica*, 201(2):213–305, December 2008.

[Unr12] Dominique Unruh. Quantum Proofs of Knowledge. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, Lecture Notes in Computer Science, pages 135–152. Springer Berlin Heidelberg, 2012.

[Vad99] Salil Pravin Vadhan. *A Study of Statistical Zero-Knowledge Proofs*. PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1999.

[Val76] Leslie G. Valiant. Universal Circuits (Preliminary Report). In *Proceedings of the Eighth Annual ACM Symposium on Theory of Computing*, STOC '76, pages 196–203, New York, NY, USA, 1976. ACM.

[VV12] Umesh Vazirani and Thomas Vidick. Certifiable Quantum Dice: Or, True Random Number Generation Secure Against Quantum Adversaries. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 61–76, New York, NY, USA, 2012. ACM.

[VV19] Umesh Vazirani and Thomas Vidick. Fully Device Independent Quantum Key Distribution. *Commun. ACM*, 62(4):133–133, March 2019.

[VZ12] Salil Vadhan and Colin Jia Zheng. Characterizing Pseudoentropy and Simplifying Pseudorandom Generator Constructions. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 817–836, New York, NY, USA, 2012. ACM.

[VZ13a] Salil Vadhan and Colin Jia Zheng. A Uniform Min-Max Theorem with Applications in Cryptography. Technical Report 101, 2013.

[VZ13b] Salil Vadhan and Colin Jia Zheng. A Uniform Min-Max Theorem with Applications in Cryptography. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013*, Lecture Notes in Computer Science, pages 93–110. Springer Berlin Heidelberg, 2013.

[Wat09] J. Watrous. Zero-Knowledge against Quantum Attacks. *SIAM Journal on Computing*, 39(1):25–58, January 2009.

[WTHR11] Severin Winkler, Marco Tomamichel, Stefan Hengl, and Renato Renner. Impossibility of Growing Quantum Bit Commitments. *Physical Review Letters*, 107(9):090502, August 2011.

[WW08] Stephanie Wehner and Jürg Wullschleger. Composable Security in the Bounded-Quantum-Storage Model. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming*, Lecture Notes in Computer Science, pages 604–615. Springer Berlin Heidelberg, 2008.

[WZ82] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802, October 1982.

[Yao82] A. C. Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (Sfcs 1982)*, pages 80–91, November 1982.

[Zha11] Jiapeng Zhang. On the query complexity for Showing Dense Model. Technical Report 038, 2011.

[Zha12] M. Zhandry. How to Construct Quantum Random Functions. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 679–687, October 2012.

[Zhe13] Jia Zheng. A Uniform Min-Max Theorem and Characterizations of Computational Randomness. page 206, 2013.