Commitment and Oblivious Transfer in the Bounded Storage Model with Errors

Rafael Dowsley, Felipe Lacerda and Anderson C. A. Nascimento

***Abstract*—The bounded storage model restricts the memory of an adversary in a cryptographic protocol, rather than restricting its computational power, making information theoretically secure protocols feasible. We present the first protocols for commitment and oblivious transfer in the bounded storage model with errors, i.e., the model where the public random sources available to the two parties are not exactly the same, but instead are only required to have a small Hamming distance between themselves. Commitment and oblivious transfer protocols were known previously only for the error-free variant of the bounded storage model, which is harder to realize.**

***Index Terms*—Bounded storage model, commitment, oblivious transfer, unconditional security, error correction.**

I. Introduction

Commitment schemes are fundamental building blocks of modern cryptography. They are important in the construction of protocols such as identification protocols [2], contract signing [3], zero-knowledge proofs [4], coin flipping over the phone [5], and more generally in two- and multi-party computation protocols [6], [7]. A commitment scheme is a two-stage protocol between two parties, Alice and Bob. First they execute the *commit stage*, in which Alice chooses a value v as input and commits to it. Later, they execute the *open stage*, in which Alice reveals v to Bob. For the protocol to be secure, it must satisfy two conditions: the *hiding property*, which means that Bob cannot learn any information about v before the open stage, and the *binding property*, which means that after the commit phase, Alice cannot change v without that being detected by Bob.

Oblivious transfer (OT) is another essential primitive for two- and multi-party computation. It is a two-party protocol in which Alice inputs two strings s_0 and s_1, and Bob inputs a bit c. Bob's output is the string s_c. The protocol is secure if Alice never learns the choice bit c and Bob does not learn any information about s_{1-c}. OT is a fundamental building block for multi-party computation and can be used to realize *any* secure two-party computation [8], [9].

Unconditionally secure commitment and OT are impossible in the setting where the parties only communicate through noiseless channels (even if quantum channels are available [10]). However, both of them are possible in the context of computational security (in which the adversaries are restricted to be polynomial-time Turing machines), as long as computational hardness assumptions are made. Commitment can be obtained using generic assumptions such as the existence of pseudorandom generator [11] or (more efficiently) assuming the hardness of various specific computational problems [12], [5], [13]. OT can be obtained from dense trapdoor permutations [14] (which is conjectured to be stronger than pseudorandom generators) or assuming the hardness of many specific computational problems [15], [16], [17], [18], [19], [20], [21], [22].

Physical assumptions, such as the existence of noisy channels, enable one to obtain unconditional security for commitment and OT protocols. In this scenario the problem was studied from both the theoretical [23], [24], [25], [26], [27], [28], [29] and the efficient protocol designing [30], [31], [32], [33] points of view.

In this paper, we consider a different setting, the so called *bounded storage model* (BSM) [34], in which the adversary is assumed to have bounded memory.

A. The Bounded Storage Model

The bounded storage model assumes that both parties have access to a public random string, and that a dishonest party cannot store the whole string. This string can be obtained from a natural source, from a trusted third party, or, in some cases even generated by one of the parties.

A variety of cryptographic tasks can be implemented in the bounded storage model. Cachin and Maurer [35] proposed a key agreement protocol in the bounded storage model in which the parties have a small pre-shared key, and use it to select bits from a public random source of size n. Key agreement in this setting is always possible if the pre-shared key has size proportional to $\log n$, as long as the adversary has bounded memory. They also proposed a protocol for key agreement by public discussion (that is, without a pre-shared key) that requires $\sqrt{kn}$ (where k is the key length) samples from the random source and is thus less practical. Later, Dziembowski and Maurer [36] showed that this protocol is optimal, in the sense that one cannot have key agreement by public discussion in the bounded storage model with less than $O(\sqrt{n})$ samples.

The first OT protocol in the bounded storage model was introduced by Cachin et al. [37]. Ding [38] and Hong et al. [39] proposed improvements in a slightly different model. Ding et al. [40] obtained the first constant-round protocol.

Shikata and Yamanaka [41] and independently Alves [42] studied the problem of commitment in the bounded storage model and provided protocols for it.

A conference version of part of this work appeared at the proceedings of ISIT 2014 [1].

Rafael Dowsley and Felipe Lacerda are with the Department of Computer Science, Aarhus University, Åbogade 34, 8200, Aarhus N, Denmark. Emails: rafael@cs.au.dk, lacerda@cs.au.dk.

Anderson C. A. Nascimento is with the Institute of Technology, University of Washington Tacoma, 1900 Commerce Street, Tacoma, WA 98402-3100, USA. E-mail: andclay@uw.edu.

Unfortunately the bounded storage model assumes that there exists a random source that can be reliably broadcast to all parties, without errors in the transmission, and this is hard to realize in practice.

Consider a scenario where a satellite broadcasts a very large random string to be used in protocols in the bounded storage model. In his Ph.D. thesis, Ding [43] made an analysis of the practicality of this scenario, showing that an antenna with a surface area of $10m^2$ can be used to receive random bits from a geo-stationary satellite at rates up to 50 Gbps. Ding's analysis did not consider the fact that errors are introduced in the string and that an adversary might be able to jam signal received by a legitimate party. Our goal with this work is to study two-party protocols under more realistic assumptions.

B. *Our contribution*

In this work, we consider a more general variant of the BSM, in which errors are introduced in the public random source in arbitrary positions. This setting captures the situation in which the source is partially controlled by an adversary, and also the situation in which there are errors due to noise in the channel. It is only assumed that the fraction of errors, relative to the length of the public string, is not too large. Ding previously studied this model [44] in the context of secret key extension protocols. These protocols can be modified, at the cost of an efficiency loss, to handle the case of key agreement, when no pre-shared key exists. He defined a general paradigm for BSM randomness extraction schemes and also showed how to incorporate error correction in key agreement extension by using fuzzy extractors [45].

We give a brief introduction to the model and its notation in order to state our results. A transmission phase is executed prior to the realization of the protocols' main part. In this phase, Alice has access to a sample $x \in \{0,1\}^n$ from an αn-source X (a source with min-entropy at least αn), where $0 < \alpha < 1$, and the Bob to $\widetilde{x} \in \{0,1\}^n$ such that $\mathsf{HD}(x, \widetilde{x}) \leq \delta n$, where $\mathsf{HD}(\cdot)$ represents the hamming distance. We assume that an adversary (controlling either Alice or Bob) has complete control on where to insert the differences between the strings x and $\widetilde{x}$, thus capturing both the situation where the source is noisy and the situation where an adversary controls part of the source.

We propose the first protocols for bit commitment and OT in the BSM with errors, thus extending the results of [44] to the case of two-party secure protocols. We show that the techniques introduced by [44] originally in the context of key extension give us efficient protocols for implementing OT. Our OT protocol assumes a memory bounded Bob (i.e., he is able to store at most γn bits for $\gamma < \alpha$), but no limitation is put on Alice's memory. It works based on an efficient linear error correcting code proposed in [46] with rate β and achieving the Zyablov bound [47]. We show that as long as $\beta > 1 - \alpha - \gamma$ the protocol works for noise levels δ as severe as (approximately)

$$\max_{\beta < \widetilde{\beta} < 1} \frac{(1-\widetilde{\beta})y}{2},$$

where y is the unique value in $[0, 1/2]$ so that $h(y) = 1 - \beta/\widetilde{\beta}$ and $h(\cdot)$ is the binary entropy function. In case a random linear error correcting code is used an improved noise level can be tolerated

$$h(2\delta) < \alpha - \gamma.$$

This improvement in the resilience comes at the price of making the protocol inefficient from a computational complexity point of view, given the intractability of decoding random linear codes.

This OT protocol immediately gives us a commitment scheme. However, using OT for obtaining commitment is not a desirable solution. The communication, round and computational complexities of OT protocols are usually much higher than the ones for commitment schemes. Moreover, it could be the case that commitment protocols work for different ranges of noise δ.

We propose a direct construction of a commitment protocol that does not rely on the framework proposed by Ding [44], does not use error correcting codes at all, implements *string* commitment and has only one message from Bob to Alice. Again, we assume that Bob has limited memory. No limitations are imposed on Alice whatsoever. The protocol is very efficient and simple and works for

$$h(\delta) < \frac{\alpha - \gamma}{2}.$$

We then show that it is possible to obtain a protocol that works for a much larger range of noise

$$h(\delta) < \alpha - \gamma$$

at the cost of having one additional message in each direction and by using a family of $4k$-universal hash functions. Finally, we show that the use of families of $4k$-universal hash functions can be avoided by imposing a memory bound on Alice, instead of Bob. We note that being able to implement protocols in the memory bounded by bounding any of the parties is an important matter, particularly when one of the parties is much more powerful than the other. This protocol is based on the interactive hashing protocol of [40] and also works for

$$h(\delta) < \alpha - \gamma,$$

but has extra rounds of communication and implements *bit* rather than string commitment.

The techniques we use in our results are standard in the field: extractors, error-correcting codes, typicality tests, sampling, etc. However, to the best of our knowledge, this is the first time that these techniques are combined to obtain commitment and OT protocols in the memory bounded model with errors. Moreover, the study of how much adversarial noise can be tolerated in this model and its relation to round complexity is also original, as far as we know. Interestingly, the noise levels tolerated by our protocols are different for OT and commitment schemes. This contrasts sharply with the noiseless situation where either one has every possible secure two-party computation or nothing at all.

C. *Overview*

In Section II we present the main tools used in our protocols. Section III explains the security model. Our commitment protocols are introduced in Section IV and the oblivious transfer in Section V. A conference version of this work appeared at the proceedings of ISIT 2014 [1] and only covered the case of OT. In this full version a more detailed presentation of the case of OT is presented and the case of commitment is entirely new; the other sections are also extended accordingly.

II. Preliminaries

We use calligraphic letters for denoting domains of random variables and other sets, upper case letters for random variables and lower case letters for realizations of the random variables. We deal solely with discrete random variables. The probability mass function of a random variable X will be denoted by P_X. The set $\{1, \ldots, n\}$ will be written as $[n]$. If $x = (x_1, \ldots, x_n)$ is a sequence and $S = \{s_1, \ldots, s_t\} \subseteq [n]$, x^S denotes the sequence $(x_{s_1}, \ldots, x_{s_t})$. $u \overset{\$}{\leftarrow} U$ denotes that u is drawn from the uniform distribution over the set U and U_r is the uniformly-distributed r-bit random variable. $y \overset{\$}{\leftarrow} \mathcal{F}(x)$ denotes the act of running the probabilistic algorithm $\mathcal{F}$ with input x and obtaining the output y. $y \leftarrow \mathcal{F}(x)$ is similarly used for deterministic algorithms.

If x and y are strings, $\mathsf{HD}(x, y)$ denotes their Hamming distance (that is, the number of positions in which they differ) and $x \oplus y$ their bitwise exclusive or. Let $\log x$ denote the logarithm of x in base 2. The binary entropy function is denoted by h: for $0 \leq x \leq 1$, $h(x) = -x \log x - (1-x) \log(1-x)$. By convention, $0 \log 0 = 0$. $H(X)$ denotes the entropy of X and $I(X; Y)$ the mutual information between X and Y.

The *statistical distance* is a measure of the distance between two probability distributions. Here, we state its definition for the case of discrete probabilities.

Definition 2.1 (Statistical distance): The statistical distance $\|P_X - P_Y\|$ between two probability mass functions P_X, P_Y over an alphabet $\mathcal{X}$ is defined as

$$\|P_X - P_Y\| = \max_{A \subseteq \mathcal{X}} \left| \sum_{x \in A} P_X(x) - P_Y(x) \right|.$$

We say P_X and P_Y are ε-close if $\|P_X - P_Y\| \leq \varepsilon$.

The main entropy measure in this work is the *min-entropy*.

Definition 2.2 (Min-entropy): Let P_{XY} be a probability mass function over $\mathcal{X} \times \mathcal{Y}$. The min-entropy of X, denoted by $H_\infty(X)$, and the conditional min-entropy of X given Y, denoted by $H_\infty(X|Y)$, are respectively defined as

$$H_\infty(X) = \min_{x \in \mathcal{X}} (-\log P_X(x)) \text{ and}$$
$$H_\infty(X|Y) = \min_{y \in \mathcal{Y}} \min_{x \in \mathcal{X}} (-\log P_{X|Y=y}(x)).$$

X is called a k-source if $H_\infty(X) \geq k$.

The conditional min-entropy $H_\infty(X|Y)$ measures the extractable private randomness from the variable X, given the correlated random variable Y possessed by an adversary. The min-entropy has the problem of being sensitive to small changes in the probability mass function and due to this fact the notion of *smooth* min-entropy [48] will be used (please refer to the Appendix for a longer discussion about the advantages of the smooth variants for cryptographic applications).

Definition 2.3 (Smooth min-entropy): Let $\varepsilon > 0$ and P_{XY} be a probability mass functions. The ε-smooth min-entropy of X given Y is defined by

$$H_\infty^\varepsilon(X|Y) = \max_{X'Y' : \|P_{X'Y'} - P_{XY}\| \leq \varepsilon} H_\infty(X'|Y').$$

Intuitively, the smooth min-entropy is the maximum min-entropy in the neighborhood of the probability mass function. Similarly, we also define the max-entropy and its smooth version.

Definition 2.4 ((Smooth) Max-entropy): The max-entropy is defined as

$$H_0(X) = \log |\{x \in X | P_X(x) > 0\}|$$

and its conditional version is given by

$$H_0(X|Y) = \max_y H_0(X|Y = y).$$

The smooth variants are defined as

$$H_0^\varepsilon(X) = \min_{X' : \|P_{X'} - P_X\| \leq \varepsilon} H_0(X') \text{ and}$$
$$H_0^\varepsilon(X|Y) = \min_{X'Y' : \|P_{X'Y'} - P_{XY}\| \leq \varepsilon} H_0(X'|Y').$$

The following inequalities are smooth min-entropy analogues of the chain rule for conditional Shannon entropy.

Lemma 2.5 ([48]): Let $\varepsilon, \varepsilon', \varepsilon'' > 0$ and P_{XYZ} be a tripartite probability mass function. Then

$$H_\infty^{\varepsilon+\varepsilon'}(X, Y|Z) \geq H_\infty^\varepsilon(X|Y, Z) + H_\infty^{\varepsilon'}(Y|Z) \text{ and}$$
$$H_\infty^\varepsilon(X, Y|Z) < H_\infty^{\varepsilon+\varepsilon'+\varepsilon''}(X|Y, Z) + H_0^{\varepsilon''}(Y|Z) + \log(1/\varepsilon').$$

The notion of *min-entropy rate* and a few results regarding its preservation will be used subsequently.

Definition 2.6 (Min-entropy rate): Let X be a random variable with an alphabet $\mathcal{X}$, Y be an arbitrary random variable, and $\varepsilon \geq 0$. The min-entropy rate $R_\infty^\varepsilon(X|Y)$ is defined as

$$R_\infty^\varepsilon(X|Y) = \frac{H_\infty^\varepsilon(X|Y)}{\log |\mathcal{X}|}.$$

The following lemma follows by using Lemma 3.16 in [49] as a first step and some easy extra steps to obtain an inequality in terms of the min-entropy rate alone. It says that a source with high min-entropy also has high min-entropy when conditioned on a correlated short string. This lemma is what makes the bounded storage assumption useful: it implies that a memory bounded adversary has limited information about a string sampled from the public random string.

Lemma 2.7: Let $X \in \{0,1\}^n$ such that $R_\infty^\varepsilon(X) \geq \rho$ and Y be a random variable over $\{0,1\}^{\phi n}$. Fix $\varepsilon' > 0$. Then

$$R_\infty^{\varepsilon' + \sqrt{8\varepsilon}}(X|Y) \geq \rho - \phi - \frac{1 + \log(1/\varepsilon')}{n}.$$

Proof: Let $\psi = \rho - \phi - \frac{1+\log(1/\varepsilon')}{n}$. By lemma 3.16 in [49] we have that if $R_\infty^\varepsilon(X) \geq \rho$ then

$$\Pr_{y \overset{\$}{\leftarrow} Y} \left[R_\infty^{\sqrt{2\varepsilon}}(X|Y = y) \geq \psi \right] \geq 1 - \varepsilon' - \sqrt{2\varepsilon}.$$

To get the desired result, let $\mathcal{G} = \{y \in \mathcal{Y} | R_\infty^{\sqrt{2\varepsilon}}(X|Y = y) \geq \psi\}$ and P_{XY} be the joint probability distribution of X and Y. Let P'_{XY} be the distribution that is $\sqrt{2\varepsilon}$-close to P_{XY} and is such that $P'(X = x|Y = y) \leq 2^{-\psi n}$ for any $x \in \mathcal{X}, y \in \mathcal{G}$. Let P''_{XY} be obtained by letting $P''(X|Y = y) = P'(X|Y = y)$ for $y \in \mathcal{G}$ and defining $P''(X = x|Y = y) = 2^{-n}$ for any $x \in \mathcal{X}$, $y \notin \mathcal{G}$. As $\Pr[\mathcal{G}] \geq 1 - \varepsilon' - \sqrt{2\varepsilon}$, it holds that $\|P''_{XY} - P'_{XY}\| \leq \varepsilon' + \sqrt{2\varepsilon}$ and so $\|P''_{XY} - P_{XY}\| \leq \varepsilon' + 2\sqrt{2\varepsilon}$. Since $P''(X = x|Y = y) \leq 2^{-\psi n}$ for every $x \in \mathcal{X}, y \in \mathcal{Y}$, the lemma follows. ■

A. Averaging Samplers and Randomness Extractors

The sample-then-extract paradigm is usually employed in the bounded storage model - first some positions of the source are sampled and then an extractor is applied on these positions. Note that due to the assumption that it is infeasible to store the whole source string (the memory bound), it is not possible to apply an extractor to the complete string at once, the extractor needs to be locally computable [50]. In this context, *averaging samplers* [51], [52], [53] are a fundamental tool. Intuitively, averaging samplers produce samples such that the average value of any function applied to the sampled string is roughly the same as the average when taken over the original string.

Definition 2.8 (Averaging sampler): A function $\mathsf{Samp}\colon \{0,1\}^r \to [n]^t$ is an (μ, ν, ε)-averaging sampler if for every function $f\colon [n] \to [0,1]$ with average $\frac{\sum_{i=1}^n f(i)}{n} \geq \mu$ it holds that

$$\Pr_{\mathcal{S} \xleftarrow{\$} \mathsf{Samp}(U_r)} \left[\frac{1}{t} \sum_{i \in \mathcal{S}} f(i) \leq \mu - \nu \right] \leq \varepsilon.$$

Averaging samplers enjoy several useful properties. Particularly important to this work is the fact that averaging samplers roughly preserve the *min-entropy rate*.

Lemma 2.9 ([50]): Let $X \in \{0,1\}^n$ be such that $R_\infty(X|Y) \geq \rho$. Let τ be such that $1 \geq \rho \geq 3\tau > 0$ and $\mathsf{Samp}\colon \{0,1\}^r \to [n]^t$ be an (μ, ν, ε)-averaging sampler with distinct samples for $\mu = (\rho - 2\tau)/\log(1/\tau)$ and $\nu = \tau/\log(1/\tau)$. Then for $\mathcal{S} \xleftarrow{\$} \mathsf{Samp}(U_r)$

$$R_\infty^{\varepsilon'}(X^{\mathcal{S}}|\mathcal{S}, Y) \geq \rho - 3\tau,$$

where $\varepsilon' = \varepsilon + 2^{-\Omega(\tau n)}$.

For $t < n$, the uniform distribution over subsets of $[n]$ of size t is an averaging sampler, also called the (n,t)-*random subset sampler*.

Lemma 2.10: Let $0 < t < n$. For any $\mu, \nu > 0$, the (n,t)-random subset sampler is a $(\mu, \nu, e^{-t\nu^2/2})$-averaging sampler.

Proof: It is just a restatement of Lemma 5.5 in [54]. ■

A *randomness extractor* is a function that takes a string with high min-entropy as an input and outputs a string that is close (in the statistical distance sense) to a uniformly distributed string.

Definition 2.11 (Strong extractor): A function $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m$ is a (k, ε)-strong extractor if for every k-source X, we have

$$\|P_{\mathsf{Ext}(X,U_r),U_r} - P_{U_m,U_r}\| \leq \varepsilon.$$

The following lemma specifies the parameters of an explicit strong extractor construction [53].

Lemma 2.12 ([53]): Let $\rho, \psi > 0$ be arbitrary constants. For every $n \in \mathbb{N}$ and every $\varepsilon > e^{-n/2^{O(\log^* n)}}$, there is an explicit construction of a $(\rho n, \varepsilon)$-strong extractor $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m$ with $m = (1-\psi)\rho n$ and $r = O(\log n + \log(1/\varepsilon))$.

The OT protocol presented in this work uses a variant of a strong extractor called a *fuzzy extractor* [45]. Intuitively, fuzzy extractors are noise-resilient extractors, that is, extractors such that the extracted string can be reproduced by any party with a string that is close (in the Hamming distance sense) to the original source.

Definition 2.13 (Fuzzy extractor): A pair of functions $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m \times \{0,1\}^q$, $\mathsf{Rec}\colon \{0,1\}^n \times \{0,1\}^r \times \{0,1\}^q \to \{0,1\}^m$ is an $(k, \varepsilon, \delta, \beta)$-fuzzy extractor if:

- For every κ-source $X \in \{0,1\}^\ell$, $(Y, Q) \leftarrow \mathsf{Ext}(X, U_r)$. Then $\|P_{YU_rQ} - U_m \times P_{U_rQ}\| \leq \varepsilon$.
- For every $x, x' \in \{0,1\}^\ell$ such that $\mathsf{HD}(x, x') \leq \delta\ell$, let $r \xleftarrow{\$} U_r$, $(y, q) \leftarrow \mathsf{Ext}(x, r)$. Then it should hold that $\Pr[\mathsf{Rec}(x', r, q) = y] \geq 1 - \beta$.

Fuzzy extractors are a special case of *one-way key-agreement schemes* [55], [56]. Ultimately they are equivalent to performing information reconciliation followed by privacy amplification [57]. Since there is a restriction to close strings with respect to the Hamming distance, syndrome-based fuzzy extractors can be used, as summarized in the following lemma from Ding [44].

Lemma 2.14 ([44]): Let $1 \geq \rho, \psi > 0$ and $1/4 > \delta > 0$ be arbitrary constants. There is a constant β, depending on δ, such that for every sufficiently large $n \in \mathbb{N}$, and every $\varepsilon > e^{-n/2^{O(\log^* n)}}$, there exists an explicit construction of a $(\rho n, \varepsilon, \delta, 0)$-fuzzy extractor $(\mathsf{Ext}, \mathsf{Rec})$, where Ext is of the form $\mathsf{Ext}\colon \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m \times \{0,1\}^p$ with

$$\begin{aligned} m &= (1-\psi)\rho n, \\ r &= O\left(\log n + \log \frac{1}{\varepsilon}\right), \\ p &\leq \frac{1-\beta}{(1-\psi)\rho} m. \end{aligned}$$

Remark 2.15: The parameters β, δ refer to the error-correcting code used in the construction, specifically, a code of size n with rate β that can correct δn errors. It is known [58] that, for a given υ with $0 < \upsilon < 1/2$ and $0 \leq \mu \leq 1 - h(\upsilon)$, there exists a random linear code with minimum distance υn and $\beta \geq 1 - h(\upsilon) - \mu$ (i.e., it matches the Gilbert-Varshamov bound). However this construction has no known efficient decoding. We can instead use the concatenated solution in Theorem 4 of [46], which achieves the Zyablov bound [47]. The construction provides a code with linear-time encoding and decoding such that, for a given $0 < \beta < 1$ and $\mu > 0$, can correct δn errors, where

$$\delta \geq \max_{\beta < \widetilde{\beta} < 1} \frac{(1 - \widetilde{\beta} - \mu)y}{2}$$

and y is the unique number in $[0, 1/2]$ with $h(y) = 1 - \beta/\widetilde{\beta}$.

B. Interactive Hashing and Binary Encoding of Subsets

Interactive hashing was initially introduced in the context of computationally secure cryptography [59], but was later generalized to the information-theoretic setting, and is particularly useful in the context of designing oblivious transfer [37], [40], [60], [61], [28] and commitment protocols [41] with unconditional security. In this primitive Bob inputs a string $w \in \{0,1\}^m$ and both Alice and Bob receive as output two strings $w_0, w_1 \in \{0,1\}^m$ such that $w_0 \neq w_1$. The correctness requirement is that one of the two output strings, w_d, should be equal to w. The security guarantee for Alice is that one of the strings should be effectively beyond the control of (a malicious) Bob. On the other hand, the security guarantee for Bob states that (a malicious) Alice should not be able to learn d.

A variety of protocols for realizing interactive hashing have been proposed [37], [40], [62]. In this work interactive hashing is used as a black box since the security of our protocols does not depend on particular features of the interactive hashing protocol used, but only on its security properties.

Definition 2.16 (Interactive hashing): Interactive hashing is a protocol between Alice and Bob in which only Bob has an input $w \in \{0,1\}^m$, and both parties output $w_0, w_1 \in \{0,1\}^m$ such that $w_d = w$ for some $d \in \{0,1\}$. The protocol is called an η-uniform (t,θ)-secure interactive hashing protocol if:

1) If both parties are honest, then the random variable W_{1-d} is close to completely random, i.e., W_{1-d} is η-close to the uniform distribution on the $2^m - 1$ strings different from w_d.
2) Alice's view (possibly a malicious Alice) of the protocol is independent of d. Let $\mathsf{view}^{\mathsf{IH}}_{\mathsf{Alice}}(W)$ be Alice's view of the protocol when the input is the random variable W. Then
$$\left\{\mathsf{view}^{\mathsf{IH}}_{\mathsf{Alice}}(W)|W=W_0\right\} = \left\{\mathsf{view}^{\mathsf{IH}}_{\mathsf{Alice}}(W)|W=W_1\right\}.$$
3) For any $\mathcal{T} \subset \{0,1\}^m$ such that $|\mathcal{T}| \leq 2^t$, it should hold that after the protocol execution between an honest Alice and a possibly malicious Bob,
$$\Pr\left[W_0, W_1 \in \mathcal{T}\right] \leq \theta,$$
where the probability is over the parties' randomness.

By allowing W_{1-d} to be distributed only closely to uniform, this definition is weaker than the one usually given in the literature [61]. It is, however, enough to prove security of our oblivious transfer protocol. This more general definition allows for the possibility of using the constant-round protocol of Ding et al. [40] for interactive hashing.

Lemma 2.17 ([40]): Let t, m be positive integers such that $t \geq \log m + 2$. Then there exists a four-message (2^{-m})-uniform $(t, 2^{-(m-t)+O(\log m)})$-secure interactive hashing protocol.

The following lemma is a result by [62]. It is 0-uniform (that is, W_{1-d} is distributed uniformly), and achieves near-optimal security [61], but has the disadvantage of taking $m-1$ rounds to execute.

Lemma 2.18 ([62]): There exists a 0-uniform $(t, a\cdot 2^{-(m-t)})$-secure interactive hashing protocol for some constant $a > 0$.

A secure interactive hashing scheme guarantees that one of the outputs is random; however, in the oblivious transfer protocols, the two binary strings are not used directly, but as encodings of subsets of sequences. Thus for the protocol to succeed, both outputs need to be valid encodings of subsets of $\binom{[n]}{\ell}$. The original protocol of Cachin et al. [37] for oblivious transfer used an encoding scheme that has probability of success $1/2$, thus requiring that the protocol be repeated several times to guarantee correctness. Later, Ding et al. [40] proposed a "dense" encoding of subsets, ensuring that most m-bit strings are valid encodings. More precisely, they showed the following result.

Lemma 2.19 ([40]): Let $\ell \leq n$, $m \geq \lceil \log\binom{n}{\ell}\rceil$, $t_m = \lfloor 2^m/\binom{n}{\ell}\rfloor$. Then there exists an injective mapping $F\colon \binom{[n]}{\ell} \times [t_m] \to [2^m]$ with $|\operatorname{Im}(F)| > 2^m - \binom{n}{\ell}$.

C. Miscellaneous

Universal hash functions were introduced by Carter and Wegman [63] and are very useful in cryptography.

Definition 2.20 (t-universal hash functions): A family of functions $G = \{g\colon \mathcal{H} \to \mathcal{L}\}$ is called a *family of t-universal hash functions* if for $g \xleftarrow{\$} G$ and for any $x_1, \ldots, x_t \in \mathcal{H}$, the induced distribution on $(g(x_1), \ldots, g(x_t))$ is uniform over $\mathcal{L}^t$.

For any $\mathcal{H} = \{0,1\}^h$ and $\mathcal{L} = \{0,1\}^\ell$, there exists a t-universal family of hash functions for which the function description has size $poly(h,t)$ bits, and the sampling and computing times are in $poly(h,t)$.

The following is a basic fact that follows from simple counting.

Lemma 2.21: Let $0 \leq \delta < 1/2$ and let $x, y \in \{0,1\}^n$ such that $\mathsf{HD}(x,y) \leq \delta n$ and $H_\infty(X) \geq \alpha n$ where $0 < \alpha < 1$. Then $H_\infty(Y) \geq (\alpha - h(\delta))n$.

The next lemma shows that the restrictions of two tuples to random subsets of their positions have relative Hamming distances that are close to the one between the entire tuples.

Lemma 2.22: Let $x, y \in \{0,1\}^n$, $\mathcal{S}$ be a random subset of $[n]$ of size r and consider any $\nu \in [0,1]$. On one hand, if $\mathsf{HD}(x,y) \leq \delta n$, then $\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}}) < (\delta+\nu)r$ except with probability $e^{-r\nu^2/2}$. On the other hand, if $\mathsf{HD}(x,y) \geq \delta n$, then $\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}}) > (\delta-\nu)r$ except with probability $e^{-r\nu^2/2}$.

Proof: Lets begin with the first part of the Lemma. By Lemma 2.10, a random subset sampler is an $(\mu, \nu, e^{-r\nu^2/2})$-averaging sampler for any $\mu, \nu > 0$. Hence for any $f\colon [n] \to [0,1]$ with $\frac{1}{n}\sum_{i=1}^n f(i) \geq \mu$

$$\Pr\left[\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} f(i) \leq \mu - \nu\right] \leq e^{-r\nu^2/2}, \qquad (1)$$

Let

$$f(i) = \begin{cases} 0, & \text{if } x_i \neq y_i, \\ 1, & \text{otherwise.} \end{cases}$$

Fix $\mu = 1-\delta$. Note that $\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} f(i) = 1 - \frac{\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}})}{r}$ and

$\frac{1}{n}\sum_{i=1}^{n} f(i) = 1 - \frac{\mathsf{HD}(x,y)}{n} \geq \mu$. Thus by Equation (1)

$$\begin{aligned} e^{-r\nu^2/2} &\geq \Pr\left[\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} f(i) \leq \mu - \nu\right] \\ &= \Pr\left[1 - \frac{\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}})}{r} \leq 1 - \delta - \nu\right] \\ &= \Pr\left[\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}}) \geq (\delta + \nu)r\right], \end{aligned}$$

which proves the first part of the Lemma.

The second part of the Lemma uses the same idea, but now the function f is

$$f(i) = \begin{cases} 0, & \text{if } x_i = y_i, \\ 1, & \text{otherwise.} \end{cases}$$

Fixing $\mu = \delta$ it holds that $\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} f(i) = \frac{\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}})}{r}$ and $\frac{1}{n}\sum_{i=1}^{n} f(i) = \frac{\mathsf{HD}(x,y)}{n} \geq \mu$ and hence

$$\begin{aligned} e^{-r\nu^2/2} &\geq \Pr\left[\frac{1}{|\mathcal{S}|}\sum_{i\in\mathcal{S}} f(i) \leq \mu - \nu\right] \\ &= \Pr\left[\frac{\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}})}{r} \leq \delta - \nu\right] \\ &= \Pr\left[\mathsf{HD}(x^{\mathcal{S}}, y^{\mathcal{S}}) \leq (\delta - \nu)r\right], \end{aligned}$$

which finishes the proof of the lemma. ■

The following statement of the birthday paradox is standard.

Lemma 2.23: Let $\mathcal{A}, \mathcal{B} \subset [n]$, chosen independently at random, with $|\mathcal{A}| = |\mathcal{B}| = 2\sqrt{\ell n}$. Then

$$\Pr[|\mathcal{A} \cap \mathcal{B}| < \ell] < e^{-\ell/4}.$$

Proof: See corollary 3 in [38]. ■

The following useful lemma will also be needed in the subsequent sections.

Lemma 2.24 ([64]): Let $0 < \sigma < 1/2$. Then

$$\sum_{i=0}^{\sigma k} \binom{k}{i} \leq 2^{h(\sigma)k}.$$

The following lemma by Rompel will be also useful.

Lemma 2.25 ([65]): Suppose t is a positive even integer, $X_1, \cdots, X_u$ are t-wise independent random variables taking values in the range $[0,1]$, $X = \sum_{i=1}^{u} X_i$, $\mu = E[X]$, and $A > 0$. Then

$$\Pr\left[\,|X - \mu| > A\,\right] < O\left(\left(\frac{t\mu + t^2}{A^2}\right)^{t/2}\right).$$

III. Security Model

A. Bounded Storage Model

We work in a two-party scenario, where two players (Alice and Bob) engage in cryptographic protocols, more specifically commitment and oblivious transfer protocols. We assume that one of the players has an upper bound on the available memory. As usual in the cryptographic literature, we assume an adversary that can corrupt either party. We will call a corrupted party *dishonest*. Parties that have not been corrupted will be called *honest*.

Cryptographic protocols in the bounded storage model run a transmission phase prior to their main part. We briefly describe this phase here.

Transmission Phase: In this phase, the sender (Alice) has access to a sample $x \in \{0,1\}^n$ from an αn-source X, where $0 < \alpha < 1$, and the receiver (Bob) to $\widetilde{x} \in \{0,1\}^n$ such that $\mathsf{HD}(x, \widetilde{x}) \leq \delta n$. Note that this captures both the situation where the source is noisy and the situation where an adversary controls part of the source. In the bounded storage model normally a memory bound is imposed on both parties during this phase, but we are able to prove the security of our protocols while imposing a memory bound on only one of them (which one depends on the specific protocol). For a memory bounded Alice, she computes a randomized function $f(x)$ with output size at most γn for $\gamma < \alpha$, stores its output and discards x. Similarly, for a memory bounded Bob, he computes a randomized function $\widetilde{f}(\widetilde{x})$ with output size at most γn for $\gamma < \alpha$, stores its output and discards $\widetilde{x}$. We should mention that in all the proposed protocol the honest parties only have to store a bounded amount of information. It should also be highlighted that even if the memory bounded party gains infinite storage power after the transmission phase is over and the source is not available anymore, this does not affect the security of the protocol, i.e., it has everlasting security.

B. Secure Commitment

The main part of a commitment protocol has two phases: commitment and opening.

Commitment Phase: Alice has an input string $v \in \mathcal{V}$ (which is a realization of a random variable V) that she wants to commit to. The parties exchange messages, possibly in several rounds. Let $\mathsf{trans}^{\mathsf{CP}}(v)$ denote all the communication in this phase and $\mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(v)$ Bob's view at the end of this phase. These random variables are a function of v, the functions that the parties computed from the public random source and the parties' local randomness.

Opening Phase: Alice sends Bob the string $\widetilde{v}$ that she claims she committed to. The parties can then exchange messages in several rounds. Let $\mathsf{trans}^{\mathsf{OP}}(\widetilde{v})$ denote all the communication in this phase. In the end Bob performs a test

$$\mathsf{test}\left(\mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(v), \mathsf{trans}^{\mathsf{OP}}(\widetilde{v})\right)$$

that outputs 1 if Bob accepts Alice's commitment and 0 otherwise.

Security. A commitment protocol is called $(\lambda_{\mathsf{C}}, \lambda_{\mathsf{H}}, \lambda_{\mathsf{B}})$-secure if it satisfies the following properties:

1) λ_{C}-correct: if Alice and Bob are honest, then for every possible v, the probability that the protocol aborts is at most λ_{C}

$$\Pr\left[\text{no aborts and } \mathsf{test}\left(\mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(v), \mathsf{trans}^{\mathsf{OP}}(v)\right) = 1\right] \geq 1 - \lambda_{\mathsf{C}}.$$

2) λ_{H}-hiding: if Alice is honest then Bob's knowledge on her committed value is at most λ_{H},

$$I(V; \mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(V)|\widetilde{X}) \leq \lambda_{\mathsf{H}}.$$

3) λ_{B}-binding: if Bob is honest, then there are no v and $\widetilde{v} \neq \hat{v}$ that can be successfully open,

$$\Pr\left[\mathsf{test}\left(\mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(v), \mathsf{trans}^{\mathsf{OP}}(\widetilde{v})\right) = 1\right] \geq \lambda_{\mathsf{B}}$$

and

$$\Pr\left[\mathsf{test}\left(\mathsf{view}^{\mathsf{CP}}_{\mathsf{Bob}}(v), \mathsf{trans}^{\mathsf{OP}}(\hat{v})\right) = 1\right] \geq \lambda_{\mathsf{B}}.$$

C. Secure Oblivious Transfer

We use the definition of oblivious transfer presented in [40]. An oblivious transfer protocol is a protocol between two players, Alice and Bob, in which Alice inputs two strings $s_0, s_1 \in \mathcal{V}$ and outputs nothing, and Bob inputs $c \in \{0,1\}$ and outputs $s \in \{\perp, s_c\}$. Let $\mathsf{view}^{\mathsf{OT}}_{\mathsf{Alice}}(s_0, s_1; c)$ denote the view of an Alice that interacts with an honest Bob. Similarly, let $\mathsf{view}^{\mathsf{OT}}_{\mathsf{Bob}}(s_0, s_1; c)$ denote the view of a Bob that interacts with an honest Alice.

Intuitively, the protocol will be secure for Bob if the view of Alice does not depend on the choice bit c, and secure for Alice if Bob cannot obtain any information about s_{1-c}. However this is tricky to formalize, because a malicious Bob could choose to play with a different bit, depending on the public random source and the messages exchanged before any secret is used by Alice.

In order to have more generality, the main part of the oblivious transfer protocol is divided in two phase: the setup phase, which encompass all communication before Alice first uses her secrets, and the transfer phase, which happens from that point on. Two pairs of inputs $(s_0, s_1), (s'_0, s'_1)$ are called i-consistent if $s_i = s'_i$ for $i \in \{0,1\}$. By the end of the setup phase there should exist a random variable I, such that for any two I-consistent pairs of inputs, the resulting view of Bob is statistically close.

Security: A protocol is called $(\lambda_{\mathsf{C}}, \lambda_{\mathsf{B}}, \lambda_{\mathsf{A}})$-secure if it satisfies the following properties:

1) λ_{C}-correct: if Alice and Bob are honest, then

$$\Pr\left[\,\text{no aborts and } s = s_c\,\right] \geq 1 - \lambda_{\mathsf{C}}$$

2) λ_{B}-secure for Bob: for any strategy used by Alice,

$$\left\|\left\{\mathsf{view}^{\mathsf{OT}}_{\mathsf{Alice}}(s_0, s_1; 0)\right\} - \left\{\mathsf{view}^{\mathsf{OT}}_{\mathsf{Alice}}(s_0, s_1; 1)\right\}\right\| \leq \lambda_{\mathsf{B}}$$

3) λ_{A}-secure for Alice: for any strategy used by Bob with input c, there exists a random variable I, defined at the end of the setup stage, such that for every two I-consistent pairs $(s_0, s_1), (s'_0, s'_1)$, we have

$$\left\|\left\{\mathsf{view}^{\mathsf{OT}}_{\mathsf{Bob}}(s_0, s_1; c)\right\} - \left\{\mathsf{view}^{\mathsf{OT}}_{\mathsf{Bob}}(s'_0, s'_1; c)\right\}\right\| \leq \lambda_{\mathsf{A}}$$

IV. Commitment Protocols

In this section we present our three commitment protocols. There are tradeoffs between the range of noise that could be tolerated and the round complexity of the proposed protocols. In short, the first protocol only has one message from Bob to Alice, but tolerates less noise than the second protocol, which has more rounds. Both of these protocols impose a memory bound on Bob, the receiver of the commitment protocol, but nothing is assumed about the memory capacity of Alice. The third protocol then deals with the complementary case in which a memory bound is assumed on Alice, the committer, but no memory bound is assumed on Bob. Note that in many scenarios one of the parties is much more powerful than the other, so it is quite useful to have commitments protocols working in both directions and just assume a memory bound on the weaker party.

A. A Simple String Commitment Protocol

Next we present a quite simple string commitment protocol that only involves one message from Bob to Alice. A memory bound on Bob is assumed. The scheme works as follows. First, both parties sample a number of bits from the public source. Alice then extracts the randomness of her sample and uses it to conceal her commitment before sending it to Bob. This guarantees the hiding condition. She also computes a hash of her sample, where the hash function is chosen by Bob. Alice sends Bob the concealed commitment along with the hash value. In the open phase, Alice sends her committed value and her sampled string. Bob then performs a number of checks for consistency. These checks enforce binding. The details of the protocol are presented below.

The security parameter is ℓ and k is set as $k = 2\sqrt{\ell n}$ in order to satisfy the requirements of Lemma 2.23. Fix $\varepsilon' > 0$ and let $\rho = \alpha - \gamma - \frac{1+\log(1/\varepsilon')}{n}$ according to the requirements of Lemma 2.7. To satisfy the requirements of Lemma 2.9, fix τ such that $\frac{\rho}{3} \geq \tau > 0$, and also fix $\omega, \zeta > 0$ such that $\rho - 3\tau > \omega > 2h(\delta + \zeta)$ and $\delta + \zeta < 1/2$. Let $k_E = (\rho - 3\tau - \omega)k$ and for $\psi > 0$, $m = (1 - \psi)k_E$ be the parameters of the strong extractor (Lemma 2.12). The message space is $\mathcal{V} = \{0,1\}^m$. It is assumed that the following functionalities, which are possible due to the lemmas in Section II, are available to the parties:

- A family $\mathcal{G}$ of 2-universal hash functions $g\colon \{0,1\}^k \to \{0,1\}^{\omega k}$.
- A (k_E, ε_E)-strong extractor $\mathsf{Ext}\colon \{0,1\}^k \times \{0,1\}^r \to \{0,1\}^m$, for an arbitrary $\varepsilon_E > e^{-k/2^{O(\log^* k)}}$.

Remark 4.1: Note that it should hold that $2h(\delta) < \omega + 3\tau < \rho < \alpha - \gamma$, so the protocol is only possible if $2h(\delta) < \alpha - \gamma$.

Transmission phase:

1) Alice chooses uniformly k positions from X. Similarly, Bob samples k positions from $\widetilde{X}$. We call their sets of positions $\mathcal{A}$ and $\mathcal{B}$, respectively.

Commit phase:

1) Alice announces $\mathcal{A}$ to Bob.
2) Bob chooses $g \xleftarrow{\$} \mathcal{G}$ and sends its description to Alice.
3) Alice computes $p \leftarrow g(x^{\mathcal{A}})$, $u \xleftarrow{\$} \{0,1\}^r$, and $y \leftarrow \mathsf{Ext}(x^{\mathcal{A}}, u)$. She then computes $z = v \oplus y$ and sends (z, p, u) to Bob in order to commit to v.

Open phase:

1) Alice sends v' and w to Bob, which are defined as $v' = v$ and $w = x^{\mathcal{A}}$ in the case that she is honest.
2) Let $\mathcal{C} = \mathcal{A} \cap \mathcal{B}$, $c = |\mathcal{C}|$ and $w^{\mathcal{C}}$ be the restriction of w to the positions corresponding to the set $\mathcal{C}$. Bob verifies whether $c \geq \ell$, $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq (\delta + \zeta)c$, $p = g(w)$ and $v' = \mathsf{Ext}(w, u) \oplus z$. If any verification fails Bob outputs 0, otherwise he outputs 1.

Theorem 4.2: The protocol is $(\lambda_{\mathsf{C}}, \lambda_{\mathsf{H}}, \lambda_{\mathsf{B}})$-secure for $\lambda_{\mathsf{C}}, \lambda_{\mathsf{H}}$ and λ_{B} negligible in ℓ.

Proof: **Correctness:** It is clear that if both Alice and Bob are honest, the protocol will fail only in the case that $c < \ell$ or $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) > (\delta + \zeta)c$. By Lemma 2.23, $c \geq \ell$ except with probability at most $e^{-\ell/4}$. By Lemma 2.22, $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq (\delta + \zeta)c$ except with probability at most $e^{-c\zeta^2/2}$, which is negligible in ℓ if $c \geq \ell$.

Hiding: After the commit phase, (a possibly malicious) Bob possesses $(z, p, \mathcal{A}, u)$, g and the output of a function $\widetilde{f}(\cdot)$ of $\widetilde{x}$, where $|\widetilde{f}(\widetilde{x})| \leq \gamma n$ with $\gamma < \alpha$. The only random variable that can provide mutual information about V when conditioned on $\widetilde{X}$ is Z, but we prove below that Z is almost uniform from Bob's point of view, and so it works as an one-time pad and only negligible information can be leaked.

By Lemma 2.7,

$$R_{\infty}^{\varepsilon'}(X|\widetilde{f}(\widetilde{X})) \geq \alpha - \gamma - \frac{1 + \log(1/\varepsilon')}{n} = \rho.$$

Since Alice chooses $\mathcal{A}$ randomly and this is an $(\mu, \nu, e^{-k\nu^2/2})$-averaging sampler for any $\mu, \nu > 0$ according to Lemma 2.10. By setting $\mu = \frac{\rho - 2\tau}{\log(1/\tau)}$, $\nu = \frac{\tau}{\log(1/\tau)}$, we have by Lemma 2.9 that

$$R_{\infty}^{\varepsilon'' + \varepsilon'}(X^{\mathcal{A}}|\mathcal{A}, \widetilde{f}(\widetilde{X})) \geq \rho - 3\tau,$$

where ε'' is a negligible function of k.

It holds that

$$\begin{aligned} &H_{\infty}^{\varepsilon'' + \varepsilon'}(X^{\mathcal{A}}|G(X^{\mathcal{A}}), \mathcal{A}, U, G, \widetilde{f}(\widetilde{X})) \\ &\quad = H_{\infty}^{\varepsilon'' + \varepsilon'}(X^{\mathcal{A}}|G(X^{\mathcal{A}}), \mathcal{A}, \widetilde{f}(\widetilde{X})) \\ &\quad \geq H_{\infty}^{\varepsilon'' + \varepsilon'}(X^{\mathcal{A}}|\mathcal{A}, \widetilde{f}(\widetilde{X})) - H_0(G(X^{\mathcal{A}})) \\ &\quad \geq (\rho - 3\tau - \omega)k \\ &\quad = k_E. \end{aligned}$$

Therefore, setting ε' and ε_E to be negligible in ℓ, the use of the strong extractor to obtain y (and of y to xor the message) guarantees that only negligible information about the committed message can be leaked.

Binding: The protocol is binding if, after the commit phase, Alice cannot choose between two different values to successfully open. Let $\sigma = \delta + \zeta$. The only way Alice can cheat is if she can come up with two strings w, w' such that $g(w) = g(w')$, $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq \sigma c$ and $\mathsf{HD}(w'^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq \sigma c$ (with $c \geq \ell$). If this happens, it holds that either there are two strings w, w' such that $g(w) = g(w')$, $\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) \leq \sigma k$ and $\mathsf{HD}(w', \widetilde{x}^{\mathcal{A}}) \leq \sigma k$; or Alice can compute w (without knowing the set $\mathcal{B}$ that together with $\mathcal{A}$ determines $\mathcal{C}$) such that $\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) > \sigma k$ and $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq \sigma c$. It is proven below that the probability that Alice succeeds in cheating decreases exponentially with the security parameter ℓ (or, equivalently in k, c). First the probability that there exists two different strings w, w' both within Hamming distance σk from $\widetilde{x}^{\mathcal{A}}$ and such that $g(w) = g(w')$ is upper bounded by

$$\Pr\left[\exists w, w' \text{ s.t.} \begin{cases} w \neq w' \\ g(w) = g(w') \\ \mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) \leq \sigma k \\ \mathsf{HD}(w', \widetilde{x}^{\mathcal{A}}) \leq \sigma k \end{cases}\right] =$$

$$= \sum_{w\,:\,\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) \leq \sigma k} \left(\sum_{w' \neq w\,:\,\mathsf{HD}(w', \widetilde{x}^{\mathcal{A}}) \leq \sigma k} 2^{-\omega k} \right)$$
$$\leq 2^{-(\omega - 2h(\sigma))k}$$

where Lemma 2.24 was used to obtain the inequality. By design, it holds that $\omega > 2h(\sigma)$, therefore the probability that Alice successfully cheats by finding two strings that are at distance at most σk from $\widetilde{x}^{\mathcal{A}}$ and hash to the same value is negligible in k.

Now considering the second case, by assumption w has Hamming distance $(\sigma + \psi)k$ from $\widetilde{x}^{\mathcal{A}}$ for some $\psi > 0$. Since Bob is honest, $\mathcal{B}$ is chosen randomly. Hence Lemma 2.22 can be applied and thus the probability that $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq \sigma c$ is smaller than $e^{-c\psi^2/2}$. ■

B. Extending the Feasibility Region

While the previous protocol is simple, efficient and round optimal, it works for a rather limited range of noise: $h(\delta) < (\alpha - \gamma)/2$. We next present a more elaborate protocol that works for a much larger range of noise $h(\delta) < \alpha - \gamma$ at the cost of increasing the rounds of communication. The memory bound is still on Bob. The idea for guaranteeing the binding property is to use two rounds of hash challenge-responses in order to guarantee the binding condition. Consider the initial set of viable strings that Alice can possibly send to Bob during the commitment phase that would pass the Hamming distance test. The first hash challenge-response round binds Alice to one specific output of the hash function, and thus restrict the set of viable strings to be polynomial in the security parameter. The second hash challenge-response round then binds Alice to one specific value for the commitment. Our solution is based on families of $4k$-universal hash functions. This approach has been used before in a different context [23].

The security parameter is ℓ and k is set as $k = 2\sqrt{\ell n}$ in order to satisfy the requirements of Lemma 2.23. Fix $\varepsilon' > 0$ and let $\rho = \alpha - \gamma - \frac{1 + \log(1/\varepsilon')}{n}$ according to the requirements of Lemma 2.7. To satisfy the requirements of Lemma 2.9, fix τ such that $\frac{\rho}{3} \geq \tau > 0$, and also fix $\omega_1, \omega_2, \zeta > 0$ such that $\rho - 3\tau > \omega_1 + \omega_2$, $\omega_1 > h(\delta + \zeta)$, and $\delta + \zeta < 1/2$. Let $k_E = (\rho - 3\tau - \omega_1 - \omega_2)k$ and for $\psi > 0$, $m = (1 - \psi)k_E$ be the parameters of the strong extractor (Lemma 2.12). The message space is $\mathcal{V} = \{0,1\}^m$. It is assumed that the following functionalities, which are possible due to the lemmas in Section II, are available to the parties:

- A family $\mathcal{G}_1$ of $4k$-universal hash functions $g_1 \colon \{0,1\}^k \to \{0,1\}^{\omega_1 k}$.
- A family $\mathcal{G}_2$ of 2-universal hash functions $g_2 \colon \{0,1\}^k \to \{0,1\}^{\omega_2 k}$.

- A (k_E, ε_E)-strong extractor $\mathsf{Ext}\colon \{0,1\}^k \times \{0,1\}^r \to \{0,1\}^m$, for an arbitrary $\varepsilon_E > e^{-k/2^{O(\log^* k)}}$.

Remark 4.3: Note that it should hold that $h(\delta) < \omega_1 + 3\tau < \rho < \alpha - \gamma$, so the protocol is only possible if $h(\delta) < \alpha - \gamma$.

Transmission phase:

1) Alice chooses uniformly k positions from X. Similarly, Bob samples k positions from $\widetilde{X}$. We call their sets of positions $\mathcal{A}$ and $\mathcal{B}$, respectively.

Commit phase:

1) Alice announces $\mathcal{A}$ to Bob.
2) Bob chooses $g_1 \xleftarrow{\$} \mathcal{G}_1$ and sends its description to Alice.
3) Alice computes $p_1 \leftarrow g_1(x^{\mathcal{A}})$ and sends it to Bob.
4) Bob chooses $g_2 \xleftarrow{\$} \mathcal{G}_2$ and sends its description to Alice.
5) Alice computes $p_2 \leftarrow g_2(x^{\mathcal{A}})$, $u \xleftarrow{\$} \{0,1\}^r$, and $y \leftarrow \mathsf{Ext}(x^{\mathcal{A}}, u)$. She then computes $z = v \oplus y$ and sends (z, p_2, u) to Bob in order to commit to v.

Open phase:

1) Alice sends v' and w to Bob, which are defined as $v' = v$ and $w = x^{\mathcal{A}}$ in the case that she is honest.
2) Let $\mathcal{C} = \mathcal{A} \cap \mathcal{B}$, $c = |\mathcal{C}|$ and $w^{\mathcal{C}}$ be the restriction of w to the positions corresponding to the set $\mathcal{C}$. Bob verifies whether $c \geq \ell$, $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq (\delta + \zeta)c$, $p_1 = g_1(w)$, $p_2 = g_2(w)$ and $v' = \mathsf{Ext}(w, u) \oplus z$. If any verification fails Bob outputs 0, otherwise he outputs 1.

Theorem 4.4: The protocol is $(\lambda_{\mathsf{C}}, \lambda_{\mathsf{H}}, \lambda_{\mathsf{B}})$-secure for $\lambda_{\mathsf{C}}, \lambda_{\mathsf{H}}$ and λ_{B} negligible in ℓ.

Proof: **Correctness:** Same as in Theorem 4.2.

Hiding: Follows the same lines as in Theorem 4.2. The difference is that here $k_E = (\rho - 3\tau - \omega_1 - \omega_2)k$ in order to account for the entropy loss due to the output of both hash functions g_1 and g_2 (instead of $k_E = (\rho - 3\tau - \omega)$ in Theorem 4.2 that accounts for the output of a single hash function g).

Binding: The protocol is binding if, after the commit phase, Alice cannot choose between two different values to successfully open. Let $\sigma = \delta + \zeta$. The only way Alice can cheat is if she can come up with two different strings w, w' that pass all tests performed by Bob during the opening phase. Either $\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) \leq \sigma k$ and $\mathsf{HD}(w', \widetilde{x}^{\mathcal{A}}) \leq \sigma k$; or Alice can compute w (without knowing the set $\mathcal{B}$ that together with $\mathcal{A}$ determines $\mathcal{C}$) such that $\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) > \sigma k$ and $\mathsf{HD}(w^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq \sigma c$. The probability that Alice succeeds in cheating in the latter case can be upper bounded as in Theorem 4.2. Below we upper bound her cheating success probability in the former case and prove that it decreases exponentially with the security parameter ℓ (or, equivalently in k).

Let the viable set dynamically denote the strings that Alice can possibly send to Bob with non-negligible probability of successful opening. Before the first round of hash challenge-response, the viable set consists of all w such that $\mathsf{HD}(w, \widetilde{x}^{\mathcal{A}}) \leq \sigma k$. Now lets consider an arbitrary fixed value p_1 for the output of the first hash. Considering the j-th viable string before the first hash challenge-response round, define I_j as 1 if the j-th viable string is mapped by g_1 to p_1; otherwise $I_j = 0$. And define $I = \sum_j I_j$. Clearly $\mu = E[I] < 1$, as g_1 is chosen from a $4k$-universal family of hash functions with range of size $\{0,1\}^{\omega_1 k}$ for $\omega_1 > h(\delta + \zeta)$. Let p_1 be called bad if I is bigger than $8k + 1$. Using the fact that g_1 is $4k$-wise independent and applying Lemma 2.25 with $t = 4k$ and $A = 2t = 8k$, we get

$$\begin{aligned} \Pr\left[I > 8k+1 \right] &< O\left(\left(\frac{t\mu + t^2}{(2t)^2} \right)^{t/2} \right) \\ &< O\left(\left(\frac{1+t}{4t} \right)^{t/2} \right) \\ &< O\left(2^{-t/2} \right). \end{aligned}$$

Then the probability that any p_1 is bad is upper bounded by

$$O\left(2^{\omega_1 k} 2^{-t/2} \right) < O\left(2^{-k} \right).$$

If the viable set is reduced to at most $8k+1$ elements after the first hash challenge-response round, then the probability that some of those collide in the second hash challenge-response round is upper bounded by

$$(8k+1)^2 \, 2^{-\omega_2 k},$$

which is negligible in k. ■

C. Alternative Bit Commitment Protocol

Next we design a *bit* commitment protocol where the memory bound is imposed on Alice instead of Bob. The protocol works for $h(\delta) < \alpha - \gamma$ and uses (the cheaper) families of 2-universal hash functions, instead of $4k$-universal hash functions. The central idea is to use an interactive hashing execution to perform the bit commitment [41].

Before describing our solution, we remark that is important to obtain protocols that work for memory bounded Alice and protocols that work for memory bounded Bob. This is particularly interesting in the case of an asymmetry of power between the parties - when one of the parties is much more powerful than the other. It makes sense to impose the bound on the weak party, whenever it is the sender of the commitment (Alice) or the receiver of the commitment (Bob).

Alice has a bit v which she wants to commit to. The security parameter is ℓ and k is set as $k = 2\sqrt{\ell n}$ in order to satisfy the requirements of Lemma 2.23. Fix $\varepsilon' > 0$ and $\xi > 0$ such that $\delta + \xi < 1/2$, and let $\rho = \alpha - \gamma - \frac{1 + \log(1/\varepsilon')}{n}$ according to the requirements of Lemma 2.7. Fix $0 < \zeta < 1$ and τ such that $\frac{\rho}{3} \geq \tau > 0$ to satisfy the requirements of Lemma 2.9. Let $\mu = \frac{\rho - 2\tau}{\log(1/\tau)}$, $\nu = \frac{\tau}{\log(1/\tau)}$ and $\varepsilon'' = e^{-\ell\nu^2/2} - 2^{-\Omega(\tau n)}$, where the last term comes from Lemma 2.9. Fix $m \geq \ell(\log k + 1)$ and $m - O(\ell) \geq t \geq m - \zeta \log(1/(\varepsilon' + \varepsilon''))$ according to Lemma 2.17. It is assumed that the following functionality, which is possible due to the lemmas in Section II-B, is available to the parties:

- An 2^{-m}-uniform $(t, 2^{-(m-t)+O(\log m)})$-secure interactive hashing protocol with input domain $\mathcal{W} = \{0,1\}^m$ and an associated dense encoding of subsets F for tuples of size k and subsets of size ℓ.

The following bit commitment protocol is correct and secure if $h(\delta+\xi) < \rho - 3\tau$.

Transmission phase:

1) Alice chooses uniformly k positions from X. Similarly, Bob samples k positions from $\widetilde{X}$. We call their sets of positions $\mathcal{A}$ and $\mathcal{B}$, respectively.

Commit phase:

1) Bob announces $\mathcal{B}$ to Alice. Alice computes $\mathcal{D} = \mathcal{A} \cap \mathcal{B}$. If $|\mathcal{D}| < \ell$, Alice aborts. Otherwise, Alice picks a random subset $\mathcal{C}$ of $\mathcal{D}$ of size ℓ.
2) Alice computes the encoding w of $\mathcal{C}$ (as a subset of $\mathcal{B}$). Alice and Bob interactively hash w, producing two strings w_0, w_1. They compute the subsets $\mathcal{C}_0, \mathcal{C}_1 \subset \mathcal{B}$ that are respectively encoded in w_0, w_1. If either encoding is invalid, they abort.
3) Alice sends $p = v \oplus d$ to Bob, where $w_d = w$.

Open phase:

1) Alice sends v' and $x'^{\mathcal{C}'}$ to Bob, which are defined as $v' = v$ and $x'^{\mathcal{C}'} = x^{\mathcal{C}}$ in the case that she is honest.
2) Bob computes $d' = p \oplus v'$ and checks whether $\mathsf{HD}(x'^{\mathcal{C}'}, \widetilde{x}^{\mathcal{C}_{d'}}) \leq (\delta+\xi)\ell$. If the verification fails Bob outputs 0, otherwise he outputs 1.

Theorem 4.5: The protocol is $(\lambda_{\mathsf{C}}, 0, \lambda_{\mathsf{B}})$-secure for λ_{C} and λ_{B} negligible in ℓ.

Proof: **Correctness:** If both participants are honest, the protocol fails only in the following cases: (1) $|\mathcal{D}| < \ell$; (2) $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) > (\delta+\xi)\ell$ or (3) w_0 or w_1 is an invalid encoding of a subset. By Lemma 2.23, $|\mathcal{D}| \geq \ell$ except with probability at most $e^{-\ell/4}$. By Lemma 2.22, $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq (\delta+\xi)\ell$ except with probability at most $e^{-\ell\xi^2/2}$. Finally, since $w_d = w$ is the encoding of C, one of the two outputs of the interactive hashing protocol is always a valid encoding. The other output W_{1-d} is 2^{-m}-close to distributed uniformly over the $2^{-m}-1$ strings different from w_d. Since it is a dense encoding, Lemma 2.19 implies that the probability that it is not a valid encoding is thus less than or equal to

$$\begin{aligned} 2^{-m} + \frac{\binom{k}{\ell}}{2^m - 1} &\leq 2^{-m} + 2^{\ell\log k - m + 1} \\ &\leq 2^{-\ell\log k - \ell} + 2^{-\ell+1} \\ &\leq 2^{-\ell+2}, \end{aligned}$$

for $m \geq \ell(\log k + 1)$.

Putting everything together this proves the correctness.

Hiding: There are two possibilities: either the protocol does not abort; or it aborts due to $|\mathcal{D}| < \ell$ or an invalid encoding. If the protocol aborts, Alice still has not sent $p = v \oplus d$, so Bob's view is independent from V. On the other hand, if the protocol does not abort, then w_{1-d} is a valid encoding of some set $\mathcal{C}'$. Due to the properties of the interactive hashing protocol, Bob's view is then consistent with both

1) Alice committing to v and $\mathcal{C}$ being the subset for which she knows the positions of x, and
2) Alice committing to $1-v$ and $\mathcal{C}'$ being the subset for which she knows the positions of x.

Hence Bob's view is independent of V.

Binding: The strategy of the proof is to demonstrate that there is an i such $X^{\mathcal{C}_i}$ has high enough min-entropy from Alice's point of view so that she cannot guess (except with negligible probability) a string $X'^{\mathcal{C}_i}$ that is close enough to $\widetilde{X}^{\mathcal{C}_i}$. Hence she will not be able to successfully use this output of the interactive hashing during the opening phase and will thus be bounded to use the other output of the interactive hashing. By the bounded storage assumption, the bounded information $f(X)$ stored by Alice is such that $|f(X)| \leq \gamma n$ with $\gamma < \alpha$. Then, by Lemma 2.7,

$$R_\infty^{\varepsilon'}(X|f(X)) \geq \alpha - \gamma - \frac{1+\log(1/\varepsilon')}{n} = \rho.$$

Since Bob is honest, $\mathcal{B}$ is randomly chosen. Lets consider a random subset $\widetilde{\mathcal{C}}$ of $\mathcal{B}$ such that $|\widetilde{\mathcal{C}}| = \ell$. This is an $(\mu, \nu, e^{-\ell\nu^2/2})$-averaging sampler for any $\mu, \nu > 0$ according to Lemma 2.10. By setting $\mu = \frac{\rho-2\tau}{\log(1/\tau)}$, $\nu = \frac{\tau}{\log(1/\tau)}$, we have by Lemma 2.9 that

$$R_\infty^{\varepsilon'+\varepsilon''}(X^{\widetilde{\mathcal{C}}}|\mathcal{B}, \widetilde{\mathcal{C}}, f(X)) \geq \rho - 3\tau,$$

for $\varepsilon'' = e^{-\ell\nu^2/2} - 2^{-\Omega(\tau n)}$. For $\widetilde{\varepsilon} = (\varepsilon' + \varepsilon'')^{1-\zeta}$, let $\mathcal{BAD}$ be the set of $\widetilde{\mathcal{C}}$'s such that $R_\infty(X^{\widetilde{\mathcal{C}}}|\mathcal{B}, \widetilde{\mathcal{C}}, f(X))$ is not $\widetilde{\varepsilon}$-close to $(\rho - 3\tau)$-min entropy rate. Due to the above equation the density of $\mathcal{BAD}$ is at most $(\varepsilon' + \varepsilon'')^\zeta$. Then the size of the set $T \subset \{0,1\}^m$ of strings that maps (using the dense encoding scheme) to subsets in $\mathcal{BAD}$ is at most $(\varepsilon' + \varepsilon'')^\zeta 2^m \leq 2^t$. Hence the properties of the interactive hashing protocol guarantee that with overwhelming probability there will be an i such that

$$R_\infty^{\widetilde{\varepsilon}}(X^{\mathcal{C}_i}|\mathcal{B}, \mathcal{C}_i, f(X), M_{IH}) \geq \rho - 3\tau,$$

where M_{IH} are the messages exchanged during the interactive hashing protocol.

However, if $h(\delta+\xi) < \rho - 3\tau$ and the min-entropy rate is at least $\rho - 3\tau$, then fixing $0 < \hat{\varepsilon} < \rho - 3\tau - h(\delta+\xi)$, for large enough ℓ, the probability that Alice guesses one of the strings $X'^{\mathcal{C}_i}$ that would be accepted by Bob as being close enough to $\widetilde{X}^{\mathcal{C}_i}$ is upper bounded by

$$2^{(h(\delta+\xi)-\rho+3\tau-\hat{\varepsilon})\ell},$$

which is a negligible function of ℓ. ■

By fixing the parameters as small as possible we have that for large enough ℓ the protocol works for values α, γ, δ which satisfy $h(\delta) < \alpha - \gamma$.

V. Oblivious Transfer Protocol

Our OT protocol imposes a memory bound on Bob. We would like to point out that it is trivial to revert the direction of OT protocols [66]. We first present the intuition behind our protocol before a detailed description. Initially, both parties sample positions from the public random source. Then the parties use an interactive hashing protocol (with an associated dense encoding) to select two subsets of the positions initially sampled by Alice. Bob inputs into the interactive hashing protocol one subset for which he has also sampled the public random source in the same positions. The other subset is out of Bob's control due to the properties of the interactive hashing protocol. Finally the positions specified by the two subsets are used as input to a fuzzy extractor in order to obtain one-time pads. Bob sends one bit indicating which input string should be xored with which one-time pad. The security for Alice is guaranteed by the fact that one of the subsets is out of Bob's control and will have high min-entropy given his view, thus resulting in a good one-time pad. The security for Bob follows from the security of the interactive hashing. The correctness follows from the correctness of the fuzzy extractor.

The security parameter is ℓ and k is set as $k = 2\sqrt{\ell n}$ in order to satisfy the requirements of Lemma 2.23. Fix $\varepsilon', \hat{\varepsilon} > 0$ and $\xi > 0$ such that $1/4 > \delta + \xi > 0$ and let $\rho = \alpha - \gamma - \frac{1+\log(1/\varepsilon')}{n}$ according to the requirements of Lemma 2.7. Fix $0 < \zeta < 1$ and τ such that $\frac{\rho}{3} \geq \tau > 0$ to satisfy the requirements of Lemma 2.9. Let $\mu = \frac{\rho - 2\tau}{\log(1/\tau)}$, $\nu = \frac{\tau}{\log(1/\tau)}$ and $\varepsilon'' = e^{-\ell\nu^2/2} - 2^{-\Omega(\tau n)}$, where the last term comes from Lemma 2.9. Fix $m \geq \ell\,(\log k + 1)$ and $m - O(\ell) \geq t \geq m - \zeta \log(1/(\varepsilon' + \varepsilon''))$ according to Lemma 2.17. For β depending on $\delta + \xi$ (see comments about the code rate below), let k_F and m_F (the parameters of the fuzzy extractor) be such that $k_F = \rho + \beta - 3\tau - 2m_F - 1 - \frac{1+\log(1/\hat{\varepsilon})}{\ell}$ and $0 < m_F < k_F$. The message is $\mathcal{V} = \{0,1\}^{m_F \ell}$. We assume the following functionalities are available to the parties (see the lemmas in Sections II-A and II-B), :

- A pair of functions $\mathsf{Ext}\colon \{0,1\}^\ell \times \{0,1\}^r \to \{0,1\}^{m_F\ell} \times \{0,1\}^q$ and $\mathsf{Rec}\colon \{0,1\}^\ell \times \{0,1\}^r \times \{0,1\}^q \to \{0,1\}^{m_F\ell}$ that constitutes an $(k_F\ell, \varepsilon_F, \delta + \xi, 0)$-fuzzy extractor where $q = (1 - R)\ell$, ε_F is an arbitrary number with $\varepsilon_F > e^{-\ell/2^{O(\log^* \ell)}}$.
- An 2^{-m}-uniform $(t, 2^{-(m-t)+O(\log m)})$-secure interactive hashing protocol with input domain $\mathcal{W} = \{0,1\}^m$ and an associated dense encoding of subsets F for tuples of size k and subsets of size ℓ.

Recall (Remark 2.15) that there is a tradeoff between the fraction of errors $\delta + \xi$ that the fuzzy extractor can tolerate and the rate β of the code used in the construction. The construction given in Theorem 4 of [46] has linear-time encoding and decoding and achieves the Zyablov bound [47]: for given $1 > \beta > 0$ and $\mu > 0$, the code has rate β and

$$\delta + \xi \geq \max_{\beta < \widetilde{\beta} < 1} \frac{(1 - \widetilde{\beta} - \mu)y}{2}, \tag{2}$$

where y is the unique number in $[0, 1/2]$ with $h(y) = 1 - \beta/\widetilde{\beta}$ and $\delta + \xi$ the amount of errors that can be corrected by the code.

Note that in order for k_F to be positive, we need to have $\rho + \beta > 1$; since ρ approaches $\alpha - \gamma$ from below in the asymptotic limit, we can obtain an upper bound for δ by setting $\beta > 1 - \alpha + \gamma$ and $\mu = 0$ in Equation (2).

Random linear codes achieve a better bound, namely, the Gilbert-Varshamov bound: for a given relative distance υ and $\mu > 0$, a random code has (with high probability) rate $\beta \geq 1 - h(\upsilon) - \mu$. Applying again the constraint that $\rho + \beta > 1$ and that $\rho \to \alpha - \gamma$ in the asymptotic limit, and using the fact that a code that can correct δn errors has relative distance $\upsilon = 2\delta + 1/n \to 2\delta$, this gives an upper bound for δ: we must have $h(2\delta) < \alpha - \gamma$. However, as noted in Remark 2.15, the random linear code construction does not have efficient decoding. It is an open question whether an efficient construction can achieve better parameters than the one from [46].

Transmission phase:

- Alice chooses uniformly k positions from X. Similarly, Bob samples k positions from $\widetilde{X}$. We call their sets of positions $\mathcal{A}$ and $\mathcal{B}$, respectively.

Setup phase:

- Alice sends $\mathcal{A}$ to Bob. Bob computes $\mathcal{D} = \mathcal{A} \cap \mathcal{B}$. If $|\mathcal{D}| < \ell$, Bob aborts. Otherwise, Bob picks a random subset $\mathcal{C}$ of $\mathcal{D}$ of size ℓ.
- Bob computes the encoding w of $\mathcal{C}$ (as a subset of $\mathcal{A}$). Alice and Bob interactively hash w, producing two strings w_0, w_1. They compute the subsets $\mathcal{C}_0, \mathcal{C}_1 \subset A$ that are respectively encoded in w_0, w_1. If either encoding is invalid, they abort.

Transfer phase:

- Bob sends $p = c \oplus d$, where $w_d = w$.
- For $i \in \{0,1\}$, Alice picks $r_i \xleftarrow{\$} \{0,1\}^r$, computes $(y_i, q_i) \leftarrow \mathsf{Ext}(x^{\mathcal{C}_i}, r_i)$ and $z_i = s_{i \oplus p} \oplus y_i$, and sends (z_i, r_i, q_i) to Bob.
- Bob computes $y' \leftarrow \mathsf{Rec}(\widetilde{x}^{\mathcal{C}}, r_d, q_d)$ and outputs $s = y' \oplus z_d$.

Theorem 5.1: The protocol is $(\lambda_{\mathsf{C}}, 0, \lambda_{\mathsf{A}})$-secure for λ_{C} and λ_{A} negligible in ℓ.

Proof: **Correctness:** We first analyze the probability of an abort. The protocol aborts if either $|\mathcal{D}| < \ell$, or if one string obtained in the interactive hashing protocol is an invalid encoding of subsets of $\mathcal{A}$. By Lemma 2.23, $\Pr[|\mathcal{D}| < \ell] < e^{-\ell/4}$. Since $w_d = w$ is the encoding of $\mathcal{C}$, one of the two string is always a valid encoding. The other output W_{1-d} is 2^{-m}-close to distributed uniformly over the $2^{-m} - 1$ strings different from w_d. Since it is a dense encoding, Lemma 2.19 implies that the probability that it is not a valid encoding is thus less than or equal to

$$\begin{aligned} 2^{-m} + \frac{\binom{k}{\ell}}{2^m - 1} &\leq 2^{-m} + 2^{\ell \log k - m + 1} \\ &\leq 2^{-\ell \log k - \ell} + 2^{-\ell + 1} \\ &\leq 2^{-\ell + 2}, \end{aligned}$$

for $m \geq \ell\,(\log k + 1)$. If both parties are honest and there is no abort, then $s = s_c$ if and only if $\mathsf{Rec}(\widetilde{x}^{\mathcal{C}}, r_d, q_d) = y_d$. By the properties of the employed fuzzy extractor, this last event happens if $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) \leq (\delta + \xi)\ell$. By Lemma 2.22, $\mathsf{HD}(x^{\mathcal{C}}, \widetilde{x}^{\mathcal{C}}) > (\delta + \xi)\ell$ with probability at most $e^{-\xi^2\ell/2}$. Putting everything together this proves the correctness.

Security for Bob: There are two possibilities: either the protocol aborts or not. If the protocol aborts in the setup phase, Bob still has not sent $p = c \oplus d$, so Alice's view is independent from C. On the other hand, if the protocol does not abort, then w_{1-d} is a valid encoding of some set $\mathcal{C}'$. Due to the properties of the interactive hashing protocol, Alice's view is then consistent with both

1) Bob choosing c and $\mathcal{C}$, and
2) Bob choosing $1-c$ and $\mathcal{C}'$.

Hence Alice's view is independent of C.

Security for Alice: There should be an index i (determined at the setup stage) such that for any two pairs $(s_0, s_1), (s'_0, s'_1)$ with $s_i = s'_i$, Bob's view of the protocol executed with (s_0, s_1) is close to his view of the protocol executed with (s'_0, s'_1).

The proof's strategy is to show that for i, $X^{\mathcal{C}_{1-i}}$ has high enough min-entropy, given Bob's view of the protocol, in such a way that Y_{1-i} is indistinguishable from an uniform distribution. Indistinguishability of Bob's views will then follow.

By the bounded storage assumption, $|\widetilde{f}(\widetilde{X})| \leq \gamma n$ with $\gamma < \alpha$. Then, by Lemma 2.7,

$$R_\infty^{\varepsilon'}(X|\widetilde{f}(\widetilde{X})) \geq \alpha - \gamma - \frac{1+\log(1/\varepsilon')}{n} = \rho.$$

Since Alice is honest, $\mathcal{A}$ is randomly chosen. Lets consider a random subset $\widetilde{\mathcal{C}}$ of $\mathcal{A}$ such that $|\widetilde{\mathcal{C}}| = \ell$. This is an $(\mu, \nu, e^{-\ell\nu^2/2})$-averaging sampler for any $\mu, \nu > 0$ according to Lemma 2.10. By setting $\mu = \frac{\rho - 2\tau}{\log(1/\tau)}$, $\nu = \frac{\tau}{\log(1/\tau)}$, we have by Lemma 2.9 that

$$R_\infty^{\varepsilon'+\varepsilon''}(X^{\widetilde{\mathcal{C}}}|\mathcal{A}, \widetilde{\mathcal{C}}, \widetilde{f}(\widetilde{X})) \geq \rho - 3\tau,$$

for $\varepsilon'' = e^{-\ell\nu^2/2} - 2^{-\Omega(\tau n)}$. For $\widetilde{\varepsilon} = (\varepsilon' + \varepsilon'')^{1-\zeta}$, let $\mathcal{BAD}$ be the set of $\widetilde{\mathcal{C}}$'s such that $R_\infty(X^{\widetilde{\mathcal{C}}}|\mathcal{A}, \widetilde{\mathcal{C}}, \widetilde{f}(\widetilde{X}))$ is not $\widetilde{\varepsilon}$-close to $(\rho - 3\tau)$-min entropy rate. Due to the above equation the density of $\mathcal{BAD}$ is at most $(\varepsilon' + \varepsilon'')^\zeta$. Then the size of the set $T \subset \{0,1\}^m$ of strings that maps (using the dense encoding scheme) to subsets in $\mathcal{BAD}$ is at most $(\varepsilon' + \varepsilon'')^\zeta 2^m \leq 2^t$. Hence the properties of the interactive hashing protocol guarantee that with overwhelming probability there will be an i such that

$$R_\infty^{\widetilde{\varepsilon}}(X^{\mathcal{C}_{1-i}}|\mathcal{A}, \mathcal{C}_{1-i}, \widetilde{f}(\widetilde{X}), M_{IH}) \geq \rho - 3\tau,$$

where M_{IH} are the messages exchanged during the interactive hashing protocol. We now show that $X^{\mathcal{C}_{1-i}}$ has high min-entropy even when given Z_i, Y_i, Q_i. We can see (Z_i, Y_i, Q_i) as a random variable over $\{0,1\}^{(2m_F+1-\beta)\ell}$. Then, by Lemma 2.7,

$$\begin{aligned} R_\infty^{\hat{\varepsilon}+\sqrt{8\widetilde{\varepsilon}}}(X^{\mathcal{C}_{1-i}}|\mathcal{A}, \mathcal{C}_{1-i}, \widetilde{f}(\widetilde{X}), M_{IH}, Z_i, Y_i, Q_i) \geq \\ \geq \rho + \beta - 3\tau - 2m_F - 1 - \frac{1+\log(1/\hat{\varepsilon})}{\ell} = k_F. \end{aligned}$$

Thus setting ε' and $\hat{\varepsilon}$ to be negligible in ℓ, the use of the $(k_F\ell, \varepsilon_F, \delta + \xi, 0)$-fuzzy extractor to obtain y_i that is used as an one-time pad guarantees that only negligible information about $s_{i \oplus e}$ can be leaked and that the protocol is λ_{A}-secure for Alice for negligible λ_{A}. ■

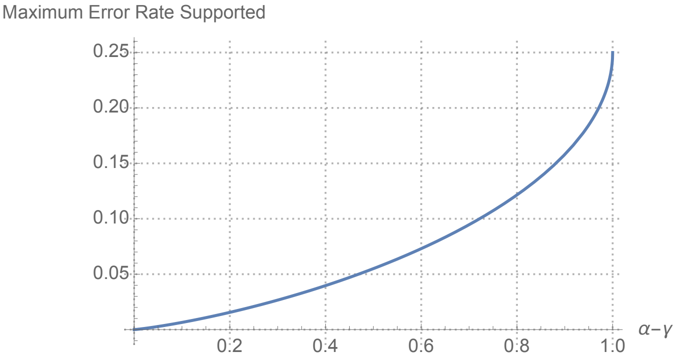

Fig. 1. Acceptable levels of noise as a function of $\alpha - \gamma$ for the oblivious transfer protocol

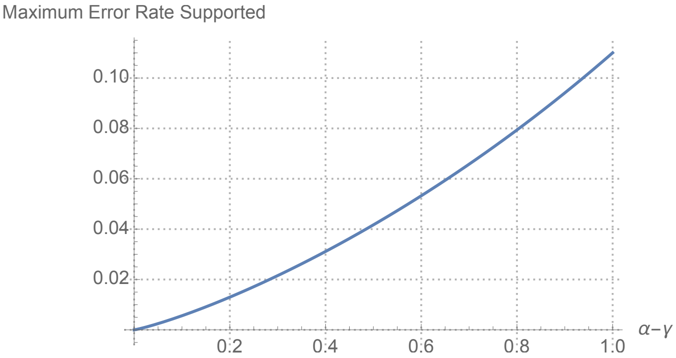

Fig. 2. Acceptable levels of noise as a function of $\alpha - \gamma$ for the non-interactive commitment protocol

VI. Discussion

In this section we briefly discuss the protocols we obtained in terms of their robustness against noise.

For the case of oblivious transfer, our best protocol works for levels of noise such that $h(2\delta) < \alpha - \gamma$. Putting $\alpha = 1$ and $\gamma = 0.5$ (that means, a public string that is perfectly random and the bound on the memory equal to half the length of the publicly available string), we have that oblivious transfer is possible if $\delta < 0.055$. Figure 1 presents the maximum supported values of noise for $\alpha - \gamma$ ranging from 0 to 1.

Our non-interactive commitment protocol works for $h(\delta) < (\alpha - \gamma)/2$. For $\alpha = 1$ and $\gamma = 0.5$ we have that non-interactive commitments are possible in the noisy memory bounded model if $\delta < 0.041$. Figure 2 presents the maximum supported values of noise for $\alpha - \gamma$ ranging from 0 to 1.

Finally, interactive commitments are possible if $h(\delta) < \alpha - \gamma$. For the same settings ($\alpha = 1$ and $\gamma = 0.5$), this gives us a maximum noise rate of $\delta < 0.11$. Figure 3 presents the maximum supported values of noise for $\alpha - \gamma$ ranging from 0 to 1.

VII. Conclusion

In this work we presented the first protocols for commitment and oblivious transfer in the bounded storage model with errors, thus extending the previous results existing in the literature for key agreement [44]. As expected, our protocols work for a limited range of values of the noise parameter δ.

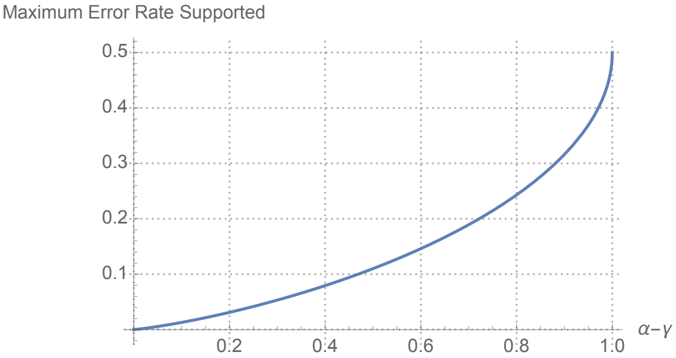

Fig. 3. Acceptable levels of noise as a function of $\alpha - \gamma$ for the interactive commitment protocols

The allowed range for our commitment schemes is different than the one for the OT protocol. For the case of commitment schemes, the range of noise that could be tolerated depended on the round complexity of the proposed protocols: extra rounds helped tolerating a more severe noise.

There are many open questions that follow our results here:

- To prove the impossibility of commitment protocols when $h(\delta) \geq \alpha - \gamma$.
- To obtain efficient OT protocols that work for the range of noise achieved by our protocols based on random linear codes.
- What is the best range of noise that can be achieved by non-interactive commitment protocols?
- Is there an intrinsic difference in the level of noise tolerated by bit commitment and OT protocols?

We do conjecture that there exists an intrinsic difference between OT and commitment schemes in the sense that there exist levels of noise so that one of them is possible but not the other. If this conjecture is proven, this would sharply contrast with the noise-free bounded memory model, where there is an all-or-nothing situation: either one has OT and bit commitment or one has nothing. Our main argument in support of this conjecture is the need for error correction in the case of OT protocols in the bounded storage model. In the case of commitment protocols error correction is not needed, the main tool used to prevent Alice from cheating is a typicality test.

References

[1] R. Dowsley, F. Lacerda, and A. C. A. Nascimento, "Oblivious transfer in the bounded storage model with errors," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 1623–1627.

[2] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *Advances in Cryptology – CRYPTO'86*, ser. Lecture Notes in Computer Science, A. M. Odlyzko, Ed., vol. 263. Springer, Aug. 1987, pp. 186–194.

[3] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Commun. ACM*, vol. 28, no. 6, pp. 637–647, Jun. 1985.

[4] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems," *Journal of the ACM*, vol. 38, no. 3, pp. 691–729, 1991.

[5] M. Blum, "Coin flipping by telephone a protocol for solving impossible problems," *SIGACT News*, vol. 15, no. 1, pp. 23–27, Jan. 1983.

[6] O. Goldreich, S. Micali, and A. Wigderson, "How to play any mental game or A completeness theorem for protocols with honest majority," in *19th Annual ACM Symposium on Theory of Computing*, A. Aho, Ed. ACM Press, May 1987, pp. 218–229.

[7] D. Chaum, I. Damgård, and J. van de Graaf, "Multiparty computations ensuring privacy of each party's input and correctness of the result," in *Advances in Cryptology – CRYPTO'87*, ser. Lecture Notes in Computer Science, C. Pomerance, Ed., vol. 293. Springer, Aug. 1988, pp. 87–119.

[8] J. Kilian, "Founding cryptography on oblivious transfer," in *20th Annual ACM Symposium on Theory of Computing*. ACM Press, May 1988, pp. 20–31.

[9] Y. Ishai, M. Prabhakaran, and A. Sahai, "Founding cryptography on oblivious transfer - efficiently," in *Advances in Cryptology – CRYPTO 2008*, ser. Lecture Notes in Computer Science, D. Wagner, Ed., vol. 5157. Springer, Aug. 2008, pp. 572–591.

[10] D. Mayers, "Unconditionally secure quantum bit commitment is impossible," *Physical review letters*, vol. 78, no. 17, pp. 3414–3417, 1997.

[11] M. Naor, "Bit commitment using pseudorandomness," *Journal of Cryptology*, vol. 4, no. 2, pp. 151–158, 1991.

[12] S. Even, "Protocol for signing contracts," in *Advances in Cryptology – CRYPTO'81*, A. Gersho, Ed., vol. ECE Report 82-04. U.C. Santa Barbara, Dept. of Elec. and Computer Eng., 1981, pp. 148–153.

[13] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Advances in Cryptology – CRYPTO'91*, ser. Lecture Notes in Computer Science, J. Feigenbaum, Ed., vol. 576. Springer, Aug. 1992, pp. 129–140.

[14] I. Haitner, "Implementing oblivious transfer using collection of dense trapdoor permutations," in *TCC 2004: 1st Theory of Cryptography Conference*, ser. Lecture Notes in Computer Science, M. Naor, Ed., vol. 2951. Springer, Feb. 2004, pp. 394–409.

[15] M. O. Rabin, "How to exchange secrets by oblivious transfer," Technical Report TR-81, Harvard Aiken Computation Laboratory, Tech. Rep., 1981.

[16] M. Bellare and S. Micali, "Non-interactive oblivious transfer and spplications," in *Advances in Cryptology – CRYPTO'89*, ser. Lecture Notes in Computer Science, G. Brassard, Ed., vol. 435. Springer, Aug. 1990, pp. 547–557.

[17] Y. T. Kalai, "Smooth projective hashing and two-message oblivious transfer," in *Advances in Cryptology – EUROCRYPT 2005*, ser. Lecture Notes in Computer Science, R. Cramer, Ed., vol. 3494. Springer, May 2005, pp. 78–95.

[18] C. Peikert, V. Vaikuntanathan, and B. Waters, "A framework for efficient and composable oblivious transfer," in *Advances in Cryptology – CRYPTO 2008*, ser. Lecture Notes in Computer Science, D. Wagner, Ed., vol. 5157. Springer, Aug. 2008, pp. 554–571.

[19] R. Dowsley, J. van de Graaf, J. Müller-Quade, and A. C. A. Nascimento, "Oblivious transfer based on the McEliece assumptions," in *ICITS 08: 3rd International Conference on Information Theoretic Security*, ser. Lecture Notes in Computer Science, R. Safavi-Naini, Ed., vol. 5155. Springer, Aug. 2008, pp. 107–117.

[20] R. Dowsley, J. van de Graaf, J. Müller-Quade, and A. C. A. Nascimento, "Oblivious transfer based on the McEliece assumptions," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E95-A, no. 2, pp. 567–575, 2012.

[21] B. David, R. Dowsley, and A. C. A. Nascimento, "Universally composable oblivious transfer based on a variant of LPN," in *CANS 14: 13th International Conference on Cryptology and Network Security*, ser. Lecture Notes in Computer Science, D. Gritzalis, A. Kiayias, and I. G. Askoxylakis, Eds., vol. 8813. Springer, Oct. 2014, pp. 143–158.

[22] P. S. L. M. Barreto, B. David, R. Dowsley, K. Morozov, and A. C. A. Nascimento, "A framework for efficient adaptively secure composable oblivious transfer in the rom," Cryptology ePrint Archive, Report 2017/993, 2017, http://eprint.iacr.org/2017/993.

[23] I. Damgård, J. Kilian, and L. Salvail, "On the (im)possibility of basing oblivious transfer and bit commitment on weakened security assumptions," in *Advances in Cryptology – EUROCRYPT'99*, ser. Lecture Notes in Computer Science, J. Stern, Ed., vol. 1592. Springer, May 1999, pp. 56–73.

[24] A. Winter, A. C. A. Nascimento, and H. Imai, "Commitment capacity of discrete memoryless channels," in *IMA Int. Conf.*, ser. Lecture Notes in Computer Science, K. G. Paterson, Ed., vol. 2898. Springer, 2003, pp. 35–51.

[25] H. Imai, K. Morozov, and A. C. A. Nascimento, "On the oblivious transfer capacity of the erasure channel," in *Information Theory (ISIT), 2006 IEEE International Symposium on*, Jul. 2006, pp. 1428–1431.

[26] R. Ahlswede and I. Csiszár, "On oblivious transfer capacity," in *Information Theory (ISIT), 2007 IEEE International Symposium on*, Jun. 2007, pp. 2061–2064.

[27] A. C. A. Nascimento and A. Winter, "On the oblivious-transfer capacity of noisy resources," *Information Theory, IEEE Transactions on*, vol. 54, no. 6, pp. 2572–2581, 2008.

[28] A. C. B. Pinto, R. Dowsley, K. Morozov, and A. C. A. Nascimento, "Achieving oblivious transfer capacity of generalized erasure channels in the malicious model," *Information Theory, IEEE Transactions on*, vol. 57, no. 8, pp. 5566–5571, 2011.

[29] R. Dowsley and A. C. A. Nascimento, "On the oblivious transfer capacity of generalized erasure channels against malicious adversaries: The case of low erasure probability," *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6819–6826, Oct 2017.

[30] C. Crépeau and J. Kilian, "Achieving oblivious transfer using weakened security assumptions," in *Foundations of Computer Science, 1988., 29th Annual Symposium on*, 1988, pp. 42–52.

[31] C. Crépeau, "Efficient cryptographic protocols based on noisy channels," in *Advances in Cryptology – EUROCRYPT'97*, ser. Lecture Notes in Computer Science, W. Fumy, Ed., vol. 1233. Springer, May 1997, pp. 306–317.

[32] D. Stebila and S. Wolf, "Efficient oblivious transfer from any non-trivial binary-symmetric channel," in *Information Theory (ISIT), 2002 IEEE International Symposium on*, Jun. 2002, p. 293.

[33] C. Crépeau, K. Morozov, and S. Wolf, "Efficient unconditional oblivious transfer from almost any noisy channel," in *SCN 04: 4th International Conference on Security in Communication Networks*, ser. Lecture Notes in Computer Science, C. Blundo and S. Cimato, Eds., vol. 3352. Springer, Sep. 2005, pp. 47–59.

[34] U. M. Maurer, "Conditionally-perfect secrecy and a provably-secure randomized cipher," *Journal of Cryptology*, vol. 5, no. 1, pp. 53–66, 1992.

[35] C. Cachin and U. M. Maurer, "Unconditional security against memory-bounded adversaries," in *Advances in Cryptology – CRYPTO'97*, ser. Lecture Notes in Computer Science, B. S. Kaliski Jr., Ed., vol. 1294. Springer, Aug. 1997, pp. 292–306.

[36] S. Dziembowski and U. M. Maurer, "The bare bounded-storage model: The tight bound on the storage requirement for key agreement," *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2790–2792, 2008.

[37] C. Cachin, C. Crépeau, and J. Marcil, "Oblivious transfer with a memory-bounded receiver," in *39th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, Nov. 1998, pp. 493–502.

[38] Y. Z. Ding, "Oblivious transfer in the bounded storage model," in *Advances in Cryptology – CRYPTO 2001*, ser. Lecture Notes in Computer Science, J. Kilian, Ed., vol. 2139. Springer, Aug. 2001, pp. 155–170.

[39] D. Hong, K.-Y. Chang, and H. Ryu, "Efficient oblivious transfer in the bounded-storage model," in *Advances in Cryptology – ASIACRYPT 2002*, ser. Lecture Notes in Computer Science, Y. Zheng, Ed., vol. 2501. Springer, Dec. 2002, pp. 143–159.

[40] Y. Z. Ding, D. Harnik, A. Rosen, and R. Shaltiel, "Constant-round oblivious transfer in the bounded storage model," in *TCC 2004: 1st Theory of Cryptography Conference*, ser. Lecture Notes in Computer Science, M. Naor, Ed., vol. 2951. Springer, Feb. 2004, pp. 446–472.

[41] J. Shikata and D. Yamanaka, "Bit commitment in the bounded storage model: Tight bound and simple optimal construction," in *13th IMA International Conference on Cryptography and Coding*, ser. Lecture Notes in Computer Science, L. Chen, Ed., vol. 7089. Springer, Dec. 2011, pp. 112–131.

[42] V. M. Alves, "Protocolo de comprometimento de bit eficiente com segurança sequencial baseado no modelo de memória limitada," Master's thesis, Universidade de Brasília, 2010.

[43] Y. Z. Ding, "Provable everlasting security in the bounded storage model," Ph.D. dissertation, Harvard University, Cambridge, MA, USA, 2001, aAI3011357.

[44] ——, "Error correction in the bounded storage model," in *TCC 2005: 2nd Theory of Cryptography Conference*, ser. Lecture Notes in Computer Science, J. Kilian, Ed., vol. 3378. Springer, Feb. 2005, pp. 578–599.

[45] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in Cryptology – EUROCRYPT 2004*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds., vol. 3027. Springer, May 2004, pp. 523–540.

[46] V. Guruswami and P. Indyk, "Near-optimal linear-time codes for unique decoding and new list-decodable codes over smaller alphabets," in *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. ACM, 2002, pp. 812–821.

[47] V. V. Zyablov, "An estimate of the complexity of constructing binary linear cascade codes," *Problemy Peredachi Informatsii*, vol. 7, no. 1, pp. 5–13, 1971.

[48] R. Renner and S. Wolf, "Simple and tight bounds for information reconciliation and privacy amplification," in *Advances in Cryptology – ASIACRYPT 2005*, ser. Lecture Notes in Computer Science, B. K. Roy, Ed., vol. 3788. Springer, Dec. 2005, pp. 199–216.

[49] Y. Z. Ding, D. Harnik, A. Rosen, and R. Shaltiel, "Constant-round oblivious transfer in the bounded storage model," *Journal of Cryptology*, vol. 20, no. 2, pp. 165–202, Apr. 2007.

[50] S. P. Vadhan, "Constructing locally computable extractors and cryptosystems in the bounded-storage model," *Journal of Cryptology*, vol. 17, no. 1, pp. 43–77, Jan. 2004.

[51] M. Bellare and J. Rompel, "Randomness-efficient oblivious sampling," in *35th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, Nov. 1994, pp. 276–287.

[52] R. Canetti, G. Even, and O. Goldreich, "Lower bounds for sampling algorithms for estimating the average," *Information Processing Letters*, vol. 53, no. 1, pp. 17–25, 13 Jan. 1995.

[53] D. Zuckerman, "Randomness-optimal oblivious sampling," *Random Structures & Algorithms*, vol. 11, no. 4, pp. 345–367, 1997.

[54] L. Babai and T. P. Hayes, "Near-independence of permutations and an almost sure polynomial bound on the diameter of the symmetric group," in *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, ser. SODA '05. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2005, pp. 1057–1066.

[55] T. Holenstein and R. Renner, "One-way secret-key agreement and applications to circuit polarization and immunization of public-key encryption," in *Advances in Cryptology – CRYPTO 2005*, ser. Lecture Notes in Computer Science, V. Shoup, Ed., vol. 3621. Springer, Aug. 2005, pp. 478–493.

[56] B. Kanukurthi and L. Reyzin, "Key agreement from close secrets over unsecured channels," in *Advances in Cryptology – EUROCRYPT 2009*, ser. Lecture Notes in Computer Science, A. Joux, Ed., vol. 5479. Springer, Apr. 2009, pp. 206–223.

[57] R. Renner and S. Wolf, "The exact price for unconditionally secure asymmetric cryptography," in *Advances in Cryptology – EUROCRYPT 2004*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds., vol. 3027. Springer, May 2004, pp. 109–125.

[58] R. Varshamov, "Estimate of the number of signals in error correcting codes," *Dokl. Akad. Nauk SSSR*, vol. 117, no. 5, pp. 739–741, 1957.

[59] R. Ostrovsky, R. Venkatesan, and M. Yung, "Fair games against an all-powerful adversary," in *Sequences II*, R. Capocelli, A. Santis, and U. Vaccaro, Eds. Springer New York, 1993, pp. 418–429.

[60] C. Crépeau and G. Savvides, "Optimal reductions between oblivious transfers using interactive hashing," in *Advances in Cryptology – EUROCRYPT 2006*, ser. Lecture Notes in Computer Science, S. Vaudenay, Ed., vol. 4004. Springer, May / Jun. 2006, pp. 201–221.

[61] G. Savvides, "Interactive Hashing and reductions between Oblivious Transfer variants," Ph.D. dissertation, McGill University, 2007.

[62] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung, "Perfect zero-knowledge arguments for NP using any one-way permutation," *Journal of Cryptology*, vol. 11, no. 2, pp. 87–108, 1998.

[63] J. L. Carter and M. N. Wegman, "Universal classes of hash functions," *Journal of Computer and System Sciences*, vol. 18, no. 2, pp. 143 – 154, 1979.

[64] N. Alon and J. H. Spencer, *The probabilistic method*. John Wiley & Sons, 2004.

[65] J. T. Rompel, "Techniques for computing with low-independence randomness," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 1990.

[66] S. Wolf and J. Wullschleger, "Oblivious transfer is symmetric," in *Advances in Cryptology – EUROCRYPT 2006*, ser. Lecture Notes in Computer Science, S. Vaudenay, Ed., vol. 4004. Springer, May / Jun. 2006, pp. 222–232.

Appendix

A. Smooth Entropies and their Applications in Cryptography

We recall that the conditional min-entropy is defined as

$$H_\infty(X|Y) = \min_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} (-\log P_{X|Y=y}(x))$$

for a probability mass function P_{XY} over $\mathcal{X} \times \mathcal{Y}$.

One faces some difficulties when trying to characterize the amount of randomness one can extract from X given Y based on this classical definition of conditional min-entropy. The first problem is that min-entropy can be highly sensitive (can change drastically) due to events that occur with a very small probability. For example, assume that ε is a negligible parameter. Consider random variables X over $\mathcal{X} = \{1, \ldots, 2^{\ell}\}$ and Y over $\mathcal{Y} = \{0, 1\}$. Intuitively Y can express the occurrence of some bad event. Let $P_{X|Y}(x|y = 0) = 1/|\mathcal{X}|$ for all $x \in \mathcal{X}$, and $P_{X|Y}(x = 1|y = 1) = 1$. On the one hand, if $P_Y(y = 1) = 0$, then $H_\infty(X|Y) = \ell$. On the other hand, if $P_Y(y = 1) = \varepsilon$ is non-zero (even if it is very small value), then $H_\infty(X|Y) = 0$.

A second problem is that the usual chain rule for entropy, which holds for the conditional Shannon entropy ($H(X|Y) = H(XY) - H(X)$), does not hold anymore for the conditional min-entropy as here defined. This makes the use of the traditional conditional min-entropy quite limited in the context of randomness extraction and its applications to cryptography. The same problems are also found with the similarly defined max-entropy $H_0(X|Y)$.

Smooth entropies solve these problems by "smoothing out" events that have a small probability. We recall the definition of conditional smooth min-entropy. For $\varepsilon > 0$, the ε-smooth min-entropy of X given Y is

$$H_\infty^{\varepsilon}(X|Y) = \max_{X'Y' : \|P_{X'Y'} - P_{XY}\| \leq \varepsilon} H_\infty(X'|Y').$$

for probability mass functions P_{XY} and $P_{X'Y'}$. Thus, $H_\infty^{\varepsilon}(X|Y)$ is maximized over all probability distributions that are ε distant from the original distribution P_{XY}. Going back to our original example where $P_{X|Y}(x|y = 0) = 1/|\mathcal{X}|$ and $P_Y(y = 1) = \varepsilon > 0$, we have that $H_\infty^{\varepsilon}(X|Y) = \ell$. $H_\infty^{\varepsilon}(X|Y)$ is "robust" to any event happening with probability at most ε.

The smooth min-entropy also has the nice property of having approximate chain rules (that depend on ε).

Lemma A.1 ([48]): Let $\varepsilon, \varepsilon', \varepsilon'' > 0$ and P_{XYZ} be a tripartite probability mass function. Then

$$\begin{aligned} H_\infty^{\varepsilon+\varepsilon'}(X, Y|Z) &\geq H_\infty^{\varepsilon}(X|Y, Z) + H_\infty^{\varepsilon'}(Y|Z) \text{ and} \\ H_\infty^{\varepsilon}(X, Y|Z) &< H_\infty^{\varepsilon+\varepsilon'+\varepsilon''}(X|Y, Z) + H_0^{\varepsilon''}(Y|Z) \\ &\quad + \log(1/\varepsilon'). \end{aligned}$$

It is shown in [48] that the conditional smooth min-entropy $H_\infty^{\varepsilon}(X|Y)$ optimally characterizes the amount of randomness that one can extract from X when Y is given as side information.

The ε-smooth variants directly allow to capture bad events that happens with probability at most ε. Another possible alternative would be to use "average min-entropy", in which the min-entropy is averaged over the random variable Y [45]. In this way, small probability events are naturally taken care of in the average computation. Moreover, one can also show that average min-entropy also satisfies an approximate chain rule [45]. Smooth min-entropy and average min-entropy are asymptotically identical when applied to independent and identically distributed distributions.