Secure Computation from Leaky Correlated Randomness

Divya Gupta* Hemanta K. Maji† Amit Sahai‡ Yuval Ishai§

Abstract

Viewed through the lens of information-theoretic cryptography, almost all nontrivial two-party secure computation requires a-priori correlated randomness to be given to parties. Furthermore, the high level of efficiency of information-theoretic protocols has motivated a paradigm of starting with correlated randomness, specifically random oblivious transfer (OT) correlations, that is set up through an offline phase. But what if some information about the correlated randomness is leaked to an adversary? Can we still recover "fresh" correlated randomness after such leakage has happened?

This question is a direct analog of the question of privacy amplification in the context of a *shared* random secret key, to the setting of *correlated* random secrets. Remarkably, despite decades of study of OT-based secure computation, very little is known about this question. In particular, the critical question of how much leakage is tolerable for preserving OT correlations has remained open. In our work, we resolve this question.

Prior to our work, the work of Ishai, Kushilevitz, Ostrovsky, and Sahai (FOCS 2009) obtained an initial feasibility result, tolerating only a tiny constant leakage rate. Since then, no progress has been made on this question. In our work, we show that starting with n random bit OT correlations, where each party holds $2n$ bits, up to $(1-\epsilon)\frac{n}{2}$ bits of leakage are tolerable. This result is optimal, by known negative results on OT combiners.

We then ask the same question for other correlations: is there a correlation that is more leakage-resilient than OT correlations, and also supports secure computation? We answer in the affirmative, by showing that there exists a correlation (that we call the inner product correlation) where each party receives $2n$ bits, and up to $(1-\epsilon)n$ bits of leakage are tolerable.

Keywords: Correlation Extractor, Random Oblivious Transfer Correlation, Information-theoretic Security, Secure Multi-party Computation, Optimal Leakage Resilience.

*University of California, Los Angeles. divyag@cs.ucla.edu.
†University of California, Los Angeles. hemanta.maji@gmail.com.
‡University of California, Los Angeles. sahai@cs.ucla.edu.
§Technion, Haifa. yuvali@cs.technion.ac.il.

Contents

1 Introduction

Secure two-party computation allows two mutually distrusting parties to perform secure computation using their private inputs without revealing any extra information to adversarial parties. It is known that even against semi-honest adversaries, i.e. adversaries who follow the prescribed protocol but are curious to find additional information, achieving information theoretic security in the plain model is impossible for most tasks [Kil88, Kus89, Bea89, MPR09]. For example, even for the seemingly trivial task of privately computing the AND of two boolean inputs is not possible. On the other hand, if suitable correlated randomness is provided as setup to parties, then two-party and multi-party computation can be performed securely [Kil00, CLOS02, IPS08]. One especially notable example of correlated randomness is the random oblivious transfer (OT) correlation, where the sender receives two random bits (s_0, s_1) and the receiver receives (c, s_c), where c is a random bit.

Due to the high efficiency of information-theoretic techniques for secure computation using OT correlations, protocols such as TinyOT [NNOB12] have popularized the approach in practice of starting with random bit OT correlations. Random OT correlations can be pre-computed in an offline phase and later used online to perform a desired secure computation. But what if some information about the correlated randomness is leaked to an adversary? Can we still recover "fresh" correlated randomness after such leakage has happened?

This question is a direct analog of the question of privacy amplification [BBR88, BBCM95] that arose in the context of secure communication. Privacy amplification asks the following question: given shared randomness which has been partially leaked to an eavesdropper, can parties agree upon a common key which remains hidden from the eavesdropper? In our setting, we ask the same question for correlated randomness, which is critical to secure computation. Note, however, that participants in a privacy amplification protocol protect their secret only from an outsider. Instead, in our setting, parties must protect their secrets against the other party. For example, a fresh oblivious transfer correlation ensures that the bit c is hidden from the sender and the bit s_{1-c} is hidden from the receiver.

In contrast to the setting of privacy amplification, remarkably, despite decades of study of OT-based secure computation, very little is known about our question. In particular, the critical question of how much leakage is tolerable for preserving OT correlations has remained open. In our work, we resolve this question.

Prior to our work, Ishai, Kushilevitz, Ostrovsky and Sahai [IKOS09] studied this question, introducing the notion of correlations extractors. They consider the setting where n copies of random OT correlations are shared among two parties. However, in their work, parties can leak only an extremely small constant fraction of the bits to the other party. Indeed, the fractional leakage resilience their protocol is approximately 10^{-7}. So, at best, this serves as a proof of concept result.

Since their work in 2009, there has not been any progress on this problem. In our work, we show that given n OT correlations as setup, we can tolerate $(1 - \varepsilon)n/2$ bits of leakage, for arbitrary constant $\varepsilon \in (0, 1)$. Further, the maximal leakage tolerance exhibited by our protocol is near-optimal [IMSW14]. Finally, our protocol is conceptually simpler. It completely avoids the use of Algebraic-Geometric codes [Gop81, GS96] needed in [IKOS09]. Instead, it uses a simple structured binary linear code.

Having resolved the question of leakage-resilience for OT correlations, we then step back, and

consider the question more broadly. While OT correlations are extremely useful and have a long history of applicability, perhaps there are other correlations that are better with respect to leakage-resilience, and still allow for secure computation. More precisely, we ask if there are correlations (X, Y) such that both parties receive $2n$ bits but where even after greater than $n/2$ bits of leakage, it is still possible to produce fresh secure OT correlations. It turns out that the answer is affirmative. The inner product correlation, where parties receive a random vector each and an additive share of their inner product, can tolerate significantly higher fractional leakage. We show that, in fact, they can tolerate $(1-\varepsilon)n$ bits of leakage, where $\varepsilon \in (0,1)$ is a constant. This opens up a new set of questions to explore in future work. In particular, we conjecture that this is optimal and that greater than n bits of leakage tolerance is impossible.

1.1 Our Contribution

In this section we highlight the main results of our paper.

1.2 Oblivious Transfer Correlation Extractor

We present our results in the terminology of "random oblivious transfer extractors." A random oblivious transfer (ROT) is a two party privative where client S receiver random bits (s_0, s_1); and the client R receives random bit c and s_c. Random oblivious transfer correlations suffice to perform general multiparty computation.

We work in the ROT^n-hybrid, that is, there are n copies of ROT correlation provided to the two parties. A semi-honest client S can leak t_S bits from the correlation and a semi-honest client R can leak t_R bits from the correlation. Oblivious transfer is a two party primitive where client S has inputs (s_0, s_1) and client R has input c; and client R obtains output s_c. An $(n, t_S, t_R, \varepsilon)$ OT extractor is a two-party protocol between client S and client R such that it produces a secure copy of oblivious transfer despite prior leakage obtained by the clients.

Our first result shows the following feasibility result:

Theorem 1 (OT Extractor)**.** *For any $n, t_S, t_R \in \mathbb{N}$, there exists an $(n, t_S, t_R, \varepsilon)$ OT Extractor which produces a secure OT, such that $\varepsilon \leqslant 2^{-(g/4+1)}$ and $g := n - (t_S + t_R)$.*

Note that our result shows that if there is sufficient *gap* between n and the total leakage $(t_S + t_R)$, then we can securely extract one oblivious transfer. Further, the simulation error decreases exponentially in the gap. For example, $t_S = t_R = 0.49n$ leakage tolerant extractors exist by our result. Contrast this to the result of [IKOS09] who can tolerate leakage up cn bits of leakage where c is a minuscule small constant. Thus, ours is the first feasibility result in the regime of high leakage tolerance; and our leakage resilience is (near) optimal due to the negative result of [IMSW14]. Our protocol also improves upon the round complexity of [IKOS09].

We show that if the gap $g = n - (t_S + t_R)$ is at least cn, for some constant $c \in (0,1)$, then we can trade off simulation error and increase the production rate of our extractor. That is, in the leaky ROT^n hybrid, we can produce large number of secure independent copies of oblivious transfer. Our result is summarized in the following theorem:

Theorem 2 (High Production)**.** *For every $m, t_S, t_R \in \mathbb{N}$, such that $g = n - (t_S + t_R) = \Theta(n)$, and $\rho = \omega(\log n)$, there exists an (n, t_S, t_R) OT Extractor with production rate $p = n/\rho$ and $\varepsilon \leqslant \mathsf{negl}(n)$.*

Intuitively, this theorem states that if the gap is linear in n then we can obtain slightly sub-linear number of secure oblivious transfers while incurring negligible security error. Although our production rate is not linear, we show that it is possible to extract large number of secure oblivious transfers even if parties are permitted to perform $t_S = t_R = 0.49n$ bits of leakage. Contrasting this with the result of [IKOS09], for practical and typical n the number of oblivious transfers produced in our scheme surpasses the number of oblivious transfers produced in their protocol. Because their production rate, although linear, is a very small constant; even a generous estimate of the rate of production puts it below $1.2 \cdot 10^{-7}$. The constants in our production rate is small, say upper bounded by 10^{-1}. So, our rate of production is $\sim (g/n)/10 \log^2 n$, which is higher than the rate of [IKOS09] in all practical settings (we use $\rho = \log^2 n$ to derive this bound). An obvious open problem is to explore whether our approach can be extended to achieve the ideal goal of producing a linear number of secure oblivious transfers even if the gap is arbitrarily small linear function of n.

Overall, our construction significantly simplifies the prior construction of [IKOS09] at a conceptual level by forgoing usage of Algebraic Geometric [Gop81, GS96] codes and instead relying on binary linear codes generated by generator matrices whose parity check matrices are random Toeplitz matrices.

Unlike [IKOS09], we do not achieve constant (multiplicative) communication overhead per instance of oblivious transfer produced. Our communication complexity overhead per oblivious transfer produced is linear in n. We also do not consider the problem of error tolerance, another important area of exploration in future work.

Restriction to Combiners. Combiners are special types of extractors where parties's leakage functions are restricted. Parties are allowed to only indicate $T \subseteq [n]$ as their leakage function. The client S can send $|T| \leqslant t_S$ and client R can send $|T| \leqslant t_R$. The leakage provided in (s_0, s_1, c, s_c) of all ROT correlations indexed by T. Note that the actual information learned by the clients is one-bit per index (because each client already knows 3 of those entries). We show that our construction yields slightly better simulation error than the general analysis of Theorem 1.

Theorem 3 (OT Combiner)**.** *For any $n, t_S, t_R \in \mathbb{N}$, there exists an $(n, t_S, t_R, \varepsilon)$ OT-Extractor which produces one secure OT, such that $\varepsilon \leqslant 2^{-g/2}$ and $g := n - (t_S + t_R)$.*

Note that the construction presented in [IMSW14] achieves similar bounds but the communication complexity in their construction is quadratic in n; while ours in linear in n. We emphasize that the higher production result of Theorem 2 also can be proven for the setting of combiner with quadratic improvement in simulation error. But Theorem 2 is a qualitative result and not a quantitative one, so we forgo this version of the result for combiners.

1.3 Larger Correlations

We show that there are correlations (X, Y) over $\{0,1\}^{4n}$ such that even though parties leak t_S and t_R bits from the correlation, such that the gap $g = n - (t_S + t_R)$ is a linear function, we can obtain asymptotically higher number of secure oblivious transfer copies. Note that the correlations (X, Y) have the same size as n copies of ROT correlation, i.e. a total of $4n$ bits of correlation provided in the hybrid.

Theorem 4 (High Production from Bigger Correlations)**.** *For every n, t_S, t_R such that $n-(t_S+t_R) = \Theta(n)$, there exists correlations (X, Y) over $\{0,1\}^{4n}$ such that, even after t_S bits of leakage to client S and t_R bits of leakage to client R, one can extract $\Theta(n/\sqrt{\log n})$ independent copies of random oblivious transfer correlations with $\mathsf{negl}(n)$ simulation error.*

Finally, we explore what is the maximum tolerable leakage parameter $\min\{t_S/s, t_R/s\}$, where the correlation (X, y) is over $\{0,1\}^s$. That is maximize the minimal fractional leakage tolerance of correlations when each party receives s bits of correlations. For example, our result Theorem 1 shows that we can achieve $t_S/s = t_R/s = 1/4 - \varepsilon$, for any constant $\varepsilon \in (0, 1/4)$.

We show that, in fact, there are correlations which can tolerate higher fractional leakage.

Theorem 5 (High Tolerance)**.** *For any $s, t \in \mathbb{N}$, there exists a correlation (X, Y) over sample space $\{0,1\}^{2s+2}$ and non-trivial correlation Z, such that, even after any party leaks t bits on the correlation (X, Y), they can securely compute one copy of Z with simulation error $\varepsilon \leqslant 2^{-(g/2+1)}$, where $g := s/2 - t$.*

A non-trivial correlation is one which suffices to securely realize oblivious transfer (by using it multiple times). The set of all non-trivial correlations against semi-honest adversaries were characterized by [Kil00]. In fact, the Z securely realized in Theorem 5 by our construction is ROT correlation itself.

It is interesting that the correlation (X, Y) demonstrated by us cannot produce multiple OT correlations, i.e. it can only produce one. Intuitively, it sacrifices maximal achievable production rate for higher resilience. We conjecture that the threshold $t_S/s \geqslant 1/2$ and $t_R/s \geqslant 1/2$ is impossible in the information theoretic world; thus, the maximum fractional tolerance demonstrated in Theorem 5 is optimal.

1.4 Prior Related Works

A closely related concept is the notion of OT combiners, which are a restricted variant of OT extractors. In this setting, parties are restricted to leaking information about individual OT correlations; and not any global leakage. The study of this field was initiated by Harnik et al. [HKN+05]. Since then, there has been work on several variant and generalizations of combiners [HIKN08, IPS08, MP06, MPW07, PW08]. Recently, [IMSW14] constructed combiners with optimal leakage parameters.

The most relevant work to our paper is the paper of Ishai, Kushilevitz, Ostrovsky, and Sahai [IKOS09], where the notion of correlation extractors was proposed. They showed that if the parties are allowed to leak a small linear amount of leakage, then a small linear number of correlations can be extracted. Both the leakage and production rates are a minuscule fraction of the initial number of correlations.

1.5 Technical Overview

We provide a short overview of our construction which proves Theorem 1. Our construction is inspired by the Massey secret sharing scheme [Mas95]. Our construction is closely related to the constructions of [IKOS09, IMSW14]. The central novelty in our construction approach is that we

choose a different class of matrices (thus, reducing communication complexity of our algorithm), but the primary technical contribution of our work is our new analysis in the context of leakage. We consider general leakage (unlike the setting of [IMSW14]) and, hence, lose a small quadratic factor in simulation error. But the same construction when used in the setting of combiners yields identical simulation error as [IMSW14].

For $i \in [n]$, suppose the client S receives random pair of bits (a_i, b_i) and client R receives (x_i, z_i), such that x_i is a random bit and $z_i = a_i x_i \oplus b_i$, from the setup. Client S picks a random codeword $(u_0, u_1, \ldots, u_n)$ in a binary linear code $\mathcal{C}$ of length $(n+1)$. Client R picks a random codeword $(r_0, r_1, \ldots, r_n)$ in the binary linear code $\mathcal{C}^\perp$ of length $(n+1)$. Note that the set of all component-wise product of such codewords has non-trivial distance. Hence, they can correct one erasure. For example, $u_0 r_0 = \sum_{i \in [n]} u_i r_i$. Hence, the clients need not explicitly compute $u_0 r_0$; but, instead, it suffices to compute $u_i r_i$ for all $i \in [n]$ and recovering one erasure thereafter.

For this section, we shall only consider privacy of client R against a semi-honest client S. Consider the following protocol: For each $i \in [n]$,

1. Client R sends $m_i = x_i \oplus r_i$.
2. Client S sends $\alpha_i = a_i \oplus u_i$. Client S sends $\beta_i = a_i m_i \oplus b_i$.

Note that client R can compute $\beta_i \oplus \alpha_i r_i \oplus z_i = u_i r_i$. To argue the privacy of client R, we need to show that r_0 remains hidden from the view of client S. Let H be the generator matrix of $\mathcal{C}^\perp$ and H is interpreted as $[H_0|H']$, where H_0 is the first column of H and H' is the remaining n columns. Note that the ability of client S to predict x_0 can be abstracted out as follows: For λ uniform random vector, given $(\lambda H', H)$, client S needs to predict λH_0.

Note that since client S is permitted to perform t_S bits of leakage on $x_{[n]}$, we have the guarantee that $x_{[n]}$ has high min-entropy on average. Now, the experiment is reminiscent of min-entropy extraction from high min-entropy sources via masking with small bias distributions. But, the uniform distribution over binary linear code $\mathcal{C}^\perp$ is not a small-bias source (projection on every dual codewords has full bias). So, we consider a set of codes $(\mathcal{C}_I, \mathcal{C}_I^\perp)$, where I is the index, such that on average these codewords have small bias. Such a distribution suffices in our setting, because leakage is performed in an offline phase and the index is chosen only in the online phase. The class of matrices chosen are binary matrices in systematic form whose parity check matrices are uniformly chosen Toeplitz matrices. This, intuitively, is the basic argument which all our proofs reduce to.

Theorem 2 is obtained by sampling $\{S_1, \ldots, S_m\}$ such that they are all distinct and each S_i indexes a set of servers. One OT is extracted by applying Theorem 1 on each index set S_i.

Theorem 4 is obtained by extracting one large correlation of the following form. Client S receives random $(a, b) \in \mathbb{F}^2$ and client R receives $(x, z) \in \mathbb{F}^2$ such that x is random and $z = ax + b$. This extraction uses ideas mentioned above. The field is suitably chose so that $\mathbb{F}$ is a product of $\mathbb{Z}_{p_i}$, for primes p_i. Now, due to Chinese Remainder Theorem, we have component-wise (a_i, b_i) with client S and (x_i, r_i) with client R. Finally, we extract $\sqrt{t}$ distinct OT from each such component, where p_i is t-bit long.

2 Preliminaries

Symbol Notations. We represent random variables by capital letters, for example X, and the values they take by small letters, for example $\mathrm{P}[X = x]$. The set $\{1, \ldots, n\}$ is represented by $[n]$, for $n \in \mathbb{N}$. Given a vector $v = (v_1, \ldots, v_n)$ and $T \subseteq [n]$, we represent $\left(v_{i_1}, \ldots, v_{i_{|T|}}\right)$ by v_T where $T = \{i_1, \ldots, i_{|T|}\}$. Similarly, given a $k \times n$ matrix G, we represent by G_T the sub-matrix of G formed by columns indexed by T. For brevity, we use G_i instead of $G_{\{i\}}$, where $i \in [n]$.

Probability Basics. A probability distribution X over a universe U is a flat source if there exists a constant $c \in (0, 1]$ such that $\mathrm{P}[X = x] \in \{0, c\}$, for all $x \in U$. Further, we say that X is a flat-source of size $1/c$. The support of X, represented as $\mathsf{Supp}(X)$ is the set of elements in the sample space which are assigned non-zero probability by the distribution X. A uniform distribution over a set S is represented by $\mathbf{U}_S$.

For a probability distribution X over a sample space U, we define $H_X(x) := -\lg \mathrm{P}[X = x]$, for every $x \in U$. The entropy of X, represented by $\mathbf{H}(X)$, is defined to be $\mathbb{E}[H_X(x)]$. The min-entropy of X, represented by $\mathbf{H}_\infty(X)$, is defined to be $\min_{x \in \mathsf{Supp}X} H_X(x)$. If $\mathbf{H}_\infty(X) \geqslant n$, then X can be written as convex linear combination of distributions, each of which are flat sources of size $\geqslant 2^n$.

Given a joint distribution (X, Y) over sample space $U \times V$, the marginal distribution Y is a distribution over sample space V such that, for any $y \in V$, the probability assigned to y is $\sum_{x \in U} \mathrm{P}[X = x, Y = y]$. The conditional distribution $(X|y)$ represents the distribution over sample space U such that the probability of $x \in U$ is $\mathrm{P}[X = x | Y = y]$. The average min-entropy [DORS08], represented by $\tilde{\mathbf{H}}_\infty(X|Y)$, is defined to be $-\lg \mathbb{E}_{y \sim Y}\left[2^{-\mathbf{H}_\infty(X|y)}\right]$.

Following lemma will be useful:

Lemma 1 ([DORS08])**.** *If $\mathbf{H}_\infty(X) \geqslant n$ and L be arbitrary ℓ-bit leakage on X, then $\tilde{\mathbf{H}}_\infty(X|L) \geqslant n - \ell$.*

The statistical distance between two distributions X and Y over a sample space U is defined to be: $\frac{1}{2}\sum_{u \in U} |\mathrm{P}[X = u] - \mathrm{P}[Y = u]|$.

Lemma 2 (Left-over Hash Lemma for Average-min-entropy [DORS08])**.** *Let $(\mathbb{F}, +, \cdot)$ be an arbitrary field. Let (X, L) be a joint distribution such that the marginal distribution X is over $\mathbb{F}^n$. Let $H : \mathbb{F}^n \to \mathbb{F}^m$ be a family of universal hash functions. Then we have:*

$$\mathrm{SD}\left((H(X), H, L), (\mathbf{U}_{\mathbb{F}^m}, H, L)\right) \leqslant \frac{1}{2}\sqrt{2^{-\tilde{\mathbf{H}}_\infty(X|L)+mf}},$$

where $f = \lg |\mathbb{F}|$.

2.1 Functionalities

We introduce some useful functionalities in this section.

Oblivious Transfer. A *2-choose-1 bit Oblivious Transfer* (referred to as OT) is a two party functionality which takes input $(s_0, s_1) \in \{0, 1\}^2$ from the *sender* and input $c \in \{0, 1\}$ from the *receiver* and outputs s_c to the receiver.

Random Oblivious Transfer. A *random 2-choose-1 bit Oblivious Transfer* (referred to as ROT) is an input-less two party functionality which samples uniformly random bits s_0, s_1, c and outputs (s_0, s_1) to the sender and (c, s_c) to the receiver. The joint distribution of sender-receiver outputs is called an ROT-correlation.

Oblivious Linear-function Evaluation. Let $(\mathbb{F}, +, \cdot)$ be an arbitrary field. An *Oblivious Linear-function Evaluation* over $\mathbb{F}$ is a two party functionality which takes inputs $(u, v) \in \mathbb{F}^2$ from the sender and $x \in \mathbb{F}$ from the receiver and outputs $u \cdot x + v$ to the receiver. This functionality is referred to as OLE($\mathbb{F}$).

The special case when $\mathbb{F} = \mathbb{GF}(2)$, is simply referred to as OLE.

Random Inner Product Correlation Party A gets $(x_{[n]}, a)$ and party B gets $(y_{[n]}, b)$ such that $x_{[n]}, y_{[n]} \xleftarrow{\$} \{0,1\}^n$, $a \xleftarrow{\$} \{0,1\}$ and $a + b = \langle x_{[n]}, y_{[n]} \rangle$. Note that for $n = 1$, this is equivalent to random oblivious transfer correlation and oblivious linear function evaluation.

2.2 Combiners and Extractors

In this section, we define oblivious transfer combiners and extractors.

Definition 1 ($(n, t_S, t_R, \varepsilon)$ (Single Use) OT-Combiner)**.** *An $(n, t_S, t_R, \varepsilon)$ (single use) OT-Combiner with production p is an interactive protocol in the clients-servers setting. There are two clients S and R; and n servers. Each server implements one instance of oblivious transfer. Client S can corrupt t_S servers and client R can corrupt t_R servers. The protocol implements p independent copies of secure oblivious transfer instances with simulation error ε.*

Definition 2 ($(n, t_S, t_R, \varepsilon)$ OT-Extractor)**.** *Let (X, Y) be the random oblivious transfer correlation. An $(n, t_S, t_R, \varepsilon)$ OT-Extractor with production rate p is an interactive protocol between two parties S and R in the $(X, Y)^n$ hybrid. Client S can leak t_S bits from the correlations and client R can leak t_R bits from the correlations. The protocol implements p independent copies of secure oblivious transfer instances with simulation error ε.*

Note that in our setting, in $(X, Y)^n$ hybrid, parties only get one sample from this correlation; unlike the typical setting where parties can invoke the trusted functionality of the hybrid multiple times. The maximum *fractional leakage resilience* is defined by the ordered tuple $(t_S/n, t_R/n)$; and the *production rate* is defined by p/n. Note that an $(n, t_S, t_R, \varepsilon)$ OT extractor with production p is also an $(n, t_S, t_R, \varepsilon)$ OT combiner with production p.

2.3 Elementary Fourier Analysis

We define $\chi_S(x) = (-1)^{\sum_{i \in S} x_i}$, where $S \subseteq [n]$ and $x \in \{0,1\}^n$. The inner product of two functions $f \colon \{0,1\}^n \to \mathbb{R}$ and $g \colon \{0,1\}^n \to \mathbb{R}$ is defined by $\mathbb{E}_{x \xleftarrow{\$} \{0,1\}^n}[f(x)g(x)]$. Given a probability distribution M over the sample space $\{0,1\}^n$, the function $f = M$ represents the function $f(x) = \mathrm{P}[M = x]$.

Definition 3 (Bias of a Distribution)**.** *Let $f : \{0,1\}^n \to \mathbb{R}$ be a probability function. The bias of f with respect to subset $S \subseteq [n]$ is defined to be:*

$$\mathsf{Bias}_S(f) := \left| \Pr_{x \sim f}[\chi_S(x) = 0] - \Pr_{x \sim f}[\chi_S(x) = 1] \right|$$

Definition 4 (Small-bias Distribution Family [DS05])**.** *Let $\mathcal{F} = \{F_1, \ldots, F_k\}$ be a family of distributions over sample space $\{0,1\}^n$ such that for every $\emptyset \neq S \subseteq [n]$, we have:*

$$\mathop{\mathbb{E}}_{i \xleftarrow{\$} [k]} \left[\mathsf{Bias}_S(F_i)^2\right] \leqslant \delta^2$$

Then the distribution family $\mathcal{F}$ is called an δ^2-biased family.

Lemma 3 (Min-entropy Extraction [NN90, AR94, GW97, DS05])**.** *Let $\mathcal{F} = \{F_1, \ldots, F_k\}$ be δ^2-biased family of distributions over the sample space $\{0,1\}^n$. Let (M, L) be a joint distribution such that the marginal distribution M is over $\{0,1\}^n$ and $\tilde{\mathbf{H}}_\infty(M|L) \geqslant m$. Then, the following holds:*

$$\mathrm{SD}\left((F_I \oplus M, L, I), \left(U_{\{0,1\}^n}, L, I\right)\right) \leqslant \frac{\delta}{2}\left(\frac{2^n}{2^m}\right)^{1/2},$$

where I is a uniform distribution over $[k]$.

2.4 Distribution over Matrices

An $k \times n$ matrix M with $\{0,1\}$ entries is in systematic form if $M = [I_{k\times k} \| P]$, where $I_{k\times k}$ is the identity matrix of dimension k and P is the parity check matrix of dimension $k \times (n-k)$. The matrix P is a Toeplitz matrix if $P_{i,j} = P_{i-1,j-1}$, for all $i \in (1,k]$ and $j \in (1, n-k]$. So, a Toeplitz matrix is uniquely define by its first row and column. We shall consider uniform distributions over $k \times n$ binary matrices in systematic form such that their parity check matrices are uniformly chosen Toeplitz matrices. A salient feature of family of such matrices is mentioned in Lemma 6.

Let $\mathbb{T}_{(k,n)}$ is a uniform distribution over matrices M of the following form. Let $M \equiv \left[I_{k\times k} \middle| P_{k\times(n-k)}\right]$, where P is a binary Toeplitz matrix of dimension $k \times (n-k)$.

Define $\mathbb{T}_{\perp,(k,n)}$ is a uniform distribution over matrices M of the following form. Let $M \equiv \left[P_{k\times(n-k)} \middle| I_{k\times k}\right]$, where P is a binary Toeplitz matrix of dimension $k \times (n-k)$.

Note that there exists an bijection between the matrices in $\mathbb{T}_{(k,n)}$ and $\mathbb{T}_{\perp,(n-k,n)}$ established by the function which maps dual matrices to each other.

3 Unpredictability Lemma

In this section we present the main unpredictability lemma.

Let us now see how Lemma 3 will be used in our results. Let $\mathbb{G}$ be the distribution $\mathbb{T}_{(k,n+1)}$. The I in Lemma 3 corresponds to $\mathbb{G}$.

Given I, the distribution F_I corresponds to a uniform distribution over the codewords generated by $G \in \mathbb{G}$. Note that, over choices of I, they form a $\delta^2 = 2^{-k}$ biased family of distributions (by Lemma 6).

For these setting of parameters, the bound of Lemma 3 reduces to:

$$\frac{1}{2}\sqrt{\frac{2^n}{2^{k+m}}}$$

This result is also true when $\mathbb{G} \equiv \mathbb{T}_{\perp,}(k, n+1)$ because they also form small biased distribution family (see Lemma 6).

Thus, as a direct consequence of Lemma 6, we obtain the following unpredictability lemma:

Lemma 4 (Unpredictability Lemma)**.** *Let* $\mathbb{G} \in \left\{\mathbb{T}_{(k,n+1)}, \mathbb{T}_{\perp,(k,n+1)}\right\}$*. Consider the following game between a honest challenger and an adversary:*

1. $\mathcal{H}$ *samples* $m_{[n]} \sim U_{\{0,1\}^n}$*.*
2. $\mathcal{A}$ *sends a leakage function* $\mathcal{L}\colon \{0,1\}^n \to \{0,1\}^t$*.*
3. $\mathcal{H}$ *sends* $\mathcal{L}(m_{[n]})$ *to* $\mathcal{A}$*.* $\mathcal{H}$ *samples* $x_{[k]} \sim U_{\{0,1\}^k}$*,* $G \sim \mathbb{G}$*; and computes* $y_{\{0\}\cup[n]} = x \cdot G \oplus (0, m_{[n]})$*.* $\mathcal{H}$ *sends* $(y_{[n]}, G)$ *to* $\mathcal{A}$*.* $\mathcal{H}$ *picks* $b \xleftarrow{\$} \{0,1\}$*. If* $b = 0$*, then she sends* $\mathsf{chal} = y_0$ *to* $\mathcal{A}$*; otherwise (if* $b = 1$*) then she sends* $\mathsf{chal} = u \sim U_{\{0,1\}}$ *to* $\mathcal{A}$*.*
4. $\mathcal{A}$ *replies a bit* $\tilde{b}$*.*

The adversary $\mathcal{A}$ *wins the game if* $b = \tilde{b}$*. For any* $\mathcal{A}$*, the advantage of the adversary is* $\leqslant \frac{1}{2}\sqrt{\frac{1}{2^{k-t}}}$*.*

All our security proofs will directly reduce to this unpredictability lemma, i.e. Lemma 4.

4 Oblivious Transfer Extractor

In this section, we shall prove Theorem 1.

4.1 Extracting One Oblivious Transfer

In this section, we present the proof of our $(n, t_S, t_R, \varepsilon)$ OT extractor which extracts one copy of secure OT. For ease of presentation, we provide our construction in the random oblivious linear evaluation (ROLE) correlation hybrid; and also produce one secure copy of oblivious linear evaluation. Recall that a ROLE correlation provides $(a, b) \xleftarrow{\$} \{0,1\}^2$ to the sender and $(x, z = ax \oplus b)$, where $x \xleftarrow{\$} \{0,1\}$, to the receiver. The security requirement insists that the sender cannot predict x and the receiver cannot predict a. Note that $(s_0 \oplus s_1)c \oplus s_0$ is identical to oblivious transfer. So, oblivious transfer and OLE are equivalent to each other; consequently, it suffices to construct a OLE extractor in ROLE^n hybrid.

The construction provided here is similar to the construction provided in [IMSW14]. But we achieve lower communication complexity and deal with general leakage instead of restricted leakage of the combiner setting. When analyzed appropriately for the combiner setting, our current protocol

Extract-One (n, t_S, t_R):
Define $g := n - (t_S + t_R)$.
Private Inputs: The clients S and R have private inputs $(s_0, s_1) \in \{0,1\}^2$ and $c \in \{0,1\}$, respectively.
Hybrid (Random Correlations): For $i \in [n]$, client S gets random $(a_i, b_i) \in \{0,1\}^2$ and client R gets (x_i, z_i), such that $x_i \in \{0,1\}$ is chosen uniformly at random and $z_i = a_i x_i \oplus b_i$.

1. *Random Code Generation.* Client R picks a binary matrix $G = [I_{k\times k} \| P_{k\times(n+1-k)}]$ of dimension $k \times (n+1)$, where $k = \lceil t_R + g/2 \rceil$ and $P_{k\times(n+1-k)}$ is a uniformly random Toeplitz matrix. Let $\mathcal{C}$ be the code generate by the generator matrix G; and H be a generator matrix for the dual code $\mathcal{C}^\perp$. If the first column of H is all-zero column then abort; otherwise continue.

2. *Random OLE Extraction.*

   (a) Client S picks a random $(u_0, \ldots, u_n) \in \mathcal{C}$. Let $\mathcal{C}_{\mathsf{parity}} \subseteq \{0,1\}^{n+1}$ be the (linear) code consisting of every length $(n+1)$ string of even parity. Client S picks a random $(v_0, \ldots, v_n) \in \mathcal{C}_{\mathsf{parity}}$.

   (b) Client R picks a random $(r_0, \ldots, r_n) \in \mathcal{C}^\perp$.

   (c) (In parallel) For each $i \in [n]$, do the following:

      i. Client R checks whether $x_i = r_i$ or not. If identical, then $m_i = \mathsf{same}$; otherwise $m_i = \mathsf{diff}$. Send m_i to client S.

      ii. If $m_i = \mathsf{same}$, define $\alpha_{0,i} = v_i \oplus b_i$ and $\alpha_{1,i} = (u_i \oplus v_i) \oplus (a_i \oplus b_i)$. Otherwise define: $\alpha_{0,i} = v_i \oplus (a_i \oplus b_i)$ and $\alpha_{1,i} = (u_i \oplus v_i) \oplus b_i$. Send $(\alpha_{0,i}, \alpha_{1,i})$ to client R.

   (d) Client R computes $t_i = \alpha_{r_i,i} \oplus z_i$ and $z = \oplus_{i\in[n]} t_i$.
   Note that $z = ax + b$, for $a = u_0$, $b = v_0$ and $x = r_0$.

3. *OLE Extraction.*

   (a) Client R checks whether $r_0 = c$ or not. If identical, then $m = \mathsf{same}$; otherwise $m = \mathsf{diff}$. Send m to the client S.

   (b) If $m = \mathsf{same}$, define $\alpha_0 = s_0 \oplus v_0$ and $\alpha_1 = (s_0 \oplus s_1) \oplus (u_0 \oplus v_0)$. Otherwise define: $\alpha_0 = s_0 \oplus (u_0 \oplus v_0)$ and $\alpha_1 = (s_0 \oplus s_1) \oplus v_0$. Send (α_0, α_1) to client R.

   (c) Client R outputs $y = \alpha_c \oplus z$.

Figure 1: Correlation Extractor Protocol which extracts one copy of Oblivious Linear Function Evaluation from n copies of Random Oblivious Linear Functions Evaluations.

achieves identical simulation error as in that paper (but reduces the communication complexity to linear from quadratic).

Note that after the correlation generation step, the protocol is only two rounds, i.e. client R sends one message (by combining steps 1 and 2.c.i) and client S replies with one message (step 2.c.ii), which is followed by one round of messages in OLE extraction phase.

No Corruption Case. We will first prove the correctness of the protocol presented in Figure 1 for the case when all clients and servers are honest and there is no leakage.

The construction does not output abort with probability $1-2^{-(n+1-k)}$, because the algorithm aborts if and only if the first row of the parity check matrix of G is all 0s. Conditioned on not aborting, we show that the protocol is perfectly correct.

First we claim that: $t_i = u_i \cdot r_i \oplus v_i$, for all $i \in [n]$. This is exhibited by the case analysis provided in Figure 2.

	$m_i = \mathsf{same}$	$m_i = \mathsf{diff}$
$x_i = 0, z_i = b_i$	$r_i = 0$	$r_i = 1$
	$\alpha_{r_i,i} = v_i \oplus b_i$	$\alpha_{r_i,i} = (u_i \oplus v_i) \oplus b_i$
	$t_i = v_i$	$t_i = u_i \oplus v_i$
$x_i = 1, z_i = a_i \oplus b_i$	$r_i = 1$	$r_i = 0$
	$\alpha_{r_i,i} = (u_i \oplus v_i) \oplus (a_i \oplus b_i)$	$\alpha_{r_i,i} = v_i \oplus (a_i \oplus b_i)$
	$t_i = a_i \oplus b_i$	$t_i = b_i$

Figure 2: Case Analysis for Correctness.

Now, we have $z = \oplus_{i\in[n]} t_i = \oplus_{i\in[n]} u_i \cdot r_i \oplus v_i = u_0 \cdot r_0 \oplus v_0$. This follows from $\oplus_{i=0}^{n} u_i \cdot r_i = 0$ and $\oplus_{i=0}^{n} v_i = 0$.

Using a similar case analysis as above, it can be shown that $y = s_1 \cdot c + s_0$.

Receiver privacy. In order to prove receiver privacy, we need to show that the choice bit c is hidden from the semi-honest sender who can obtain t_S bits of leakage. Note that it suffices to show that at the end of the random OLE extraction phase, choice bit r_0 is hidden.

Let L denote the random variable for leakage obtained by the semi-honest sender. We will denote the random variable for the choice bit vector $x_{[n]}$ for the receiver in the correlation generation phase by $X_{[n]}$. Note that $X_{[n]}$ is identical to uniform distribution over $\{0,1\}^n$. Note that L has at most t_S bits of leakage on X.

The view of client S at the end of the random correlation extraction phase is:

$$\vartheta = (a_{[n]}, b_{[n]}, G, (u_0, \ldots, u_n), (v_0, \ldots, v_n), m_{[n]}, L = \ell)$$

We will show that for any semi-honest client S, we have $\mathrm{P}(S(\vartheta) = r_0)$ close to $1/2$. Note that this is identical to $\mathrm{P}(S(H, m_{[n]}, L) = r_0)$. In Figure 1, the client R picks a random codeword $(r_0, \ldots, r_n) \in \mathcal{C}^{\perp}$. Alternatively, this can be done by picking $w \xleftarrow{\$} \{0,1\}^{n+1-k}$ and $(r_0, \ldots, r_n) = w \cdot H$, where H is the generator matrix for $\mathcal{C}^{\perp}$. Note that $m_{[n]} = (w \cdot H)_{[n]} \oplus x_{[n]}$ and $r_0 = \langle H_0, w \rangle$.

Since, the sender can leak t_S bits on $x_{[n]}$, we have: $\tilde{\mathbf{H}}_\infty(X_{[n]}|L) \geqslant m = (n - t_S)$. By Lemma 4, we have that the advantage of predicting $\langle H_0, w\rangle$ is at most: $2^{-(g/4+1)}$.

This shows that the probability distribution $(R_0|\vartheta)$ is $2^{-(g/4+1)}$ close to the uniform distribution. Here R_0 is the random variable for the choice bit of the receiver at the end of the random correlation extraction phase.

Sender privacy. In order to prove sender privacy in OLE, we need to show that the bit s_1 is hidden from the receiver after the protocol. Note that it suffices to show that at the end of the random OLE extraction phase, bit u_0 is hidden.

Let L denote the random variable for leakage obtained by the semi-honest receiver after the random correlation generation phase. Note that in the absence of any leakage, after this phase, the vector $a_{[n]}$ is hidden from the receiver. We will denote the random variable for the bit vector $a_{[n]}$ for the sender in the correlation generation phase by $A_{[n]}$. Note that $A_{[n]}$ is identical to uniform distribution over $\{0,1\}^n$ and L has at most t_R bits of leakage on $A_{[n]}$. So, we get $\tilde{\mathbf{H}}_\infty(A_{[n]}|L) \geqslant m = n - t_R$.

The view of client R at the end of the random correlation extraction phase is:

$$\vartheta = (x_{[n]}, z_{[n]}, G, (r_0, \ldots, r_n), m_{[n]}, \alpha_{\{0,1\},[n]}, L = \ell)$$

(x_i, m_i)	$\alpha_{0,i}$	$\alpha_{1,i}$
$(0, \mathsf{same})$	$v_i \oplus z_i$	$\gamma_i \oplus v_i \oplus z_i$
$(0, \mathsf{diff})$	$\tilde{v}_i \oplus z_i$	$\gamma_i \oplus \tilde{v}_i \oplus z_i$
$(1, \mathsf{diff})$	$v_i \oplus z_i$	$\gamma_i \oplus v_i \oplus z_i$
$(1, \mathsf{same})$	$\tilde{v}_i \oplus z_i$	$\gamma_i \oplus \tilde{v}_i \oplus z_i$

Table 1: Reconstruction algorithm for $\alpha_{\{0,1\},i}$.

We are interested in the conditional distribution $(U_0|\vartheta)$. We will show that for any semi-honest client R, $\mathrm{P}(R(\vartheta) = u_0)$ is close to $1/2$. We show this via a reduction to Lemma 4 in Figure 3. Given any adversary $\mathcal{A}$ who can distinguish u_0 from a uniform bit, we convert it into an adversary $\mathcal{A}$ against the honest experiment $\mathcal{H}$ of Lemma 4 with identical advantage. It is easy to see that this reduction is perfect. Note that the only difference in the simulator from the actual protocol is that the generator matrix G is being generated by the honest party $\mathcal{H}$ instead of being obtained from $\mathcal{A}$. This does not cause any issues, because we are only dealing with semi-honest adversaries. At the end of random correlation extraction phase, the advantage in predicting U_0 is at most: $2^{-(g/4+1)}$.

This shows that $(U_0|\vartheta)$ distribution is $2^{-(g/4+1)}$ close to the uniform distribution against a semi-honest receiver who leaks at most t_R bits.

Note that our simulation works even for arbitrary choice of $x_{[n]}$ and $m_{[n]}$. In particular, it works when these vectors are chosen uniformly at random.

4.2 Tighter Analysis for Combiners

The protocol provided in Figure 1 is also (by definition) an oblivious transfer combiner. We provide an extremely simple but tighter analysis of that protocol to obtain better simulation error bounds

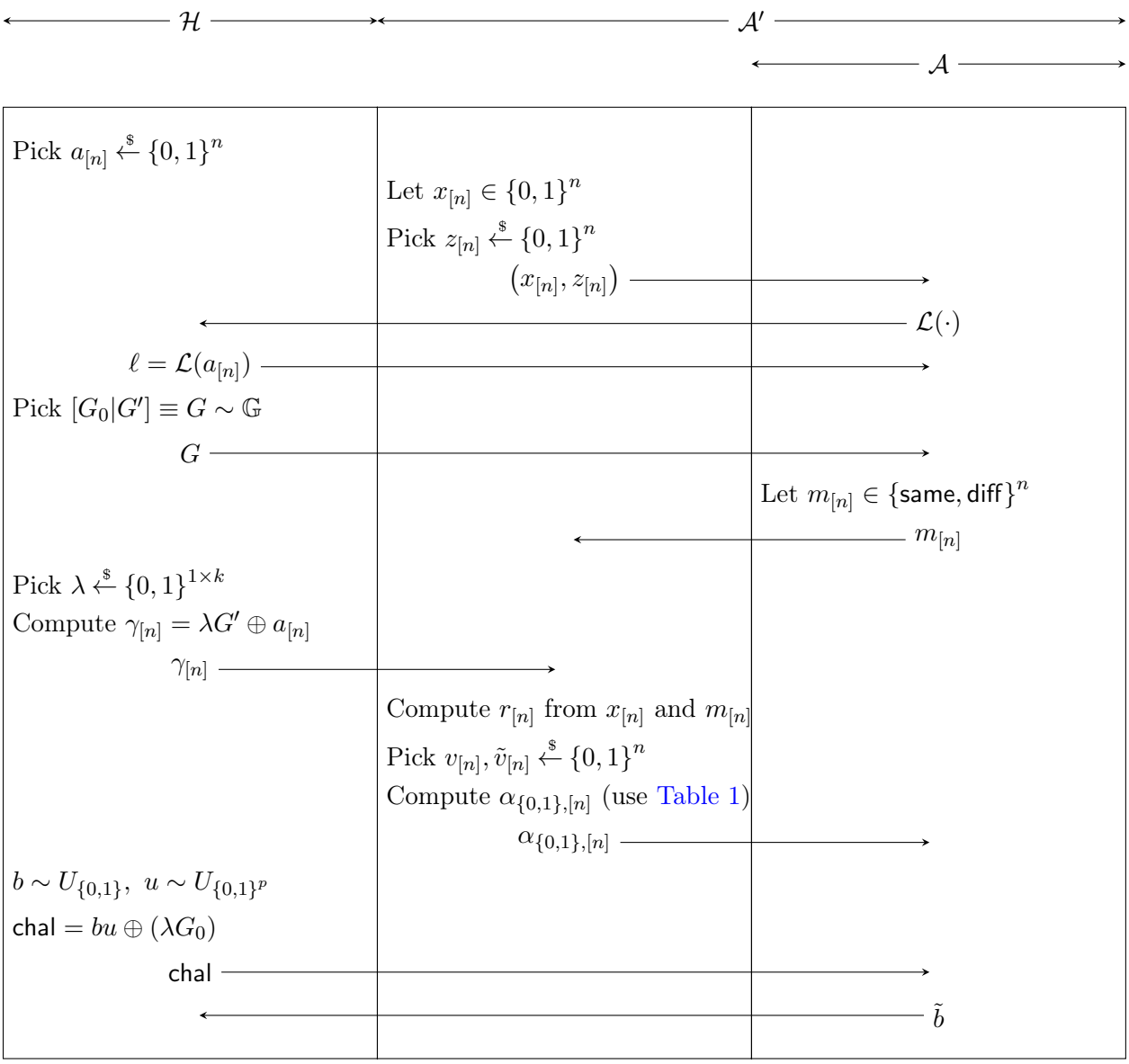

Figure 3: Simulator for Sender Privacy. The distribution $\mathbb{G}$ is uniform distribution over $k \times (n+1)$ binary matrices in systematic form whose parity check matrices are uniform Toeplitz matrices. To reconstruct r_i, define $r_i = 0$ if ($x_i = 0$ and $m_i = \mathsf{same}$) or ($x_i = 1$ and $m_i = \mathsf{diff}$).

(see Theorem 3).

Suppose the client S is semi-honest corrupt and it corrupts t_S servers indexed by $T \subseteq [n]$. Note that x_0 is perfectly hidden from the client S if and only if H_0 does not lie in the span of H_T. This happens with probability at least $1-2^{-(n+1-k)+t_S} = 1-2^{-(g/2+1)}$ (by Lemma 6). Therefore, $(X_0|\vartheta)$ is $2^{-(g/2+1)}$ to the uniform distribution over $\{0,1\}$.

Similarly, when the receiver is semi-honest corrupt, we shall show that the sender bit U_0 is perfectly hidden if and only if G_0 does not lie in the span of G_T, where $T \subseteq [n]$ is the set of servers corrupted by the client R of size at most t_R. This happens with probability at least $1 - 2^{-g/2}$ (by Lemma 6). Next, we provide the proof that U_0 is perfectly hidden if G_0 does not lie in the span of G_T. We prove the following technical claim:

Claim 1. *The distribution $(U_0|G, x, z, u_T)$ is identical to the distribution $(U_0|G, u_T)$.*

Proof. Let $\mathsf{span}\,\langle\cdot\rangle$ represent the span of its arguments. We prove this by considering the following set of exhaustive cases:

Case 1. If $G_0 \in \mathsf{span}\,\langle G_T\rangle$ then it is trivial to see that: $(U_0|G, x, z, u_T) \equiv (U_0|G, u_T)$.

Case 2. Otherwise, if $G_0 \notin \mathsf{span}\,\langle G_T\rangle$, we know that U_0 is a uniform bit conditioned on u_T. If $x_0 = 0$, then it is trivial to see that $(U_0|G, x, z, u_T) \equiv (U_0|G, u_T) \equiv U_{\{0,1\}}$.

Case 3. Now, we consider the case when $G_0 \notin \mathsf{span}\,\langle G_T\rangle$ and $x_0 = 1$. We shall prove this by exhibiting a bijection σ between the sets S_0 and S_1 defined below. Let S be the set of all (u, v) codeword-pairs which are consistent with client R's view. Define:

$$S_0 = \{(u, v) : u_0 = 0 \text{ and } (u, v) \in S\}$$
$$S_1 = \{(u, v) : u_0 = 1 \text{ and } (u, v) \in S\}$$

Let $T_0 = \{u : \exists (u, v) \in S_0\}$ and $T_1 = \{u : \exists (u, v) \in S_1\}$. Since, $G_0 \notin \mathsf{span}\,\langle G_T\rangle$ there exists a bijection μ between T_0 and T_1.

Consider fixed $u \in T_0$ and $u' \in T_1$ such that $u \mapsto_\mu u'$. Define $d_i = u_i \cdot x_i$ and $d'_i = u'_i \cdot x_i$, where $i \in [n]$. Note that $\bigoplus_{i\in[n]} d_i = 0$ and $\bigoplus_{i\in[n]} d'_i = 1$. This implies that the set $I = \{i : d_i \neq d'_i \text{ and } i \in [n] \setminus T\}$ has odd size (in particular, I is non-empty).

For every v such that $(u, v) \in S_0$ we define v' such that $v'_i = \begin{cases} v_i, & \text{if } i \notin I \\ \overline{v_i}, & \text{if } i \in I \end{cases}$. We establish the map $\sigma : S_0 \to S_1$ as follows: $(u, v) \mapsto_\sigma (u', v')$. It is easy to see that σ is a bijection. □

Note that $(U_0|G, u_T)$ is a uniform bit if and only if $G_0 \notin \mathsf{span}\,\langle G_T\rangle$.

4.3 Trading off Simulation Error with Production Rate

In this section we use sub-sampling techniques to trade-off simulation-error to get improved production rate. The main idea is to sample small subsets of distinct correlations and, subsequently, run the protocol in Figure 1 on those subsets independently. This increases the simulation error, but yields higher production rates.

In our case, we use the trivial sub-sampling technique of picking indices at random with suitable probability; in case of a sample repeating itself, we discard it and re-sample. This technique yields distinct samples and has identical properties as the naïve subsampling technique (see [Vad04]). The sophisticated techniques of [Vad04] are also relevant to our setting; but they do not yield any reduction in "simulation error increase." They are useful only to reduce the communication complexity of the protocols.

We only work in the setting where $g = n - (t_S + t_R)$ is at least cn, for some constant $c \leqslant 1$. In general c could have been a function of n, but we forgo those cases. The main technical lemma is the following:

Lemma 5 (Sub-sampling [Vad04]). *Let $(A_{[n]}, L)$ be a joint distribution such that, there exists a constant $c \in (0,1)$ such that, $\tilde{\mathbf{H}}_\infty(A_{[n]}|L) \geqslant cn$. For every constant $\varepsilon \in (0, c)$ and $\rho = \omega(\log n)$, there exists an efficient algorithm which outputs $(S_1, \ldots, S_m) \in \left(2^{[n]}\right)^m$ such that $m = n/\rho$ and with probability $1 - \mathsf{negl}(n)$, the following holds:*

1. *Large and Distinct: There exists a constant $\lambda \in (0,1)$ such that $|S_i| = \lambda n$. We have $S_i \cap S_j = \emptyset$, for all $i, j \in [m]$ and $i \neq j$.*
2. *High Entropy: $\tilde{\mathbf{H}}_\infty(S_{i+1}|S_{[i]}, L) \geqslant (c - \varepsilon)n$.*

By directly applying this result, we obtain the tradeoff of Theorem 2. This result is obtained by direct application of the subsampling algorithm in Lemma 5 and applying the protocol in Figure 1 to correlations indexed by each individual subsets S_i. We observe that the approach of subsampling to obtain "distinct subsets" while preserving min-entropy is unlikely to yield constant production rate extractors.

4.4 Extraction from Larger Correlations

In this section, we shall use slightly more complex correlations and obtain a higher rate of production than Theorem 2; albeit the production rate is still not linear.

First, we describe the correlation. Let $\{p_1, \ldots, p_m\}$ be the set of all t-bit prime numbers. Let $K = p_1 \cdots p_m$. We shall use $ROLE(\mathbb{F})$ correlations, where $\mathbb{F} = \mathbb{GF}(K)$.

Let A be any number with a bit representation. Note that the number of bits in the product of all a-bit primes which are $\leqslant A$ is $\Theta(A)$. This follows from density of primes. So, number of bits needed to represent elements in $\mathbb{F}$ is $\Theta(2^t)$.

The intuition of our algorithm is the following: Initially, we have n bits of randomness and we lose $(t_S + t_R) = \Theta(n)$ bits to leakage. So, we extract one secure $\mathbb{F}$ instance (by running the variant of Figure 1 for OLE($\mathbb{F}$)) and n is chosen such that $n - (t_S + t_R) > \log K + \omega(\log n)$, where $\log K = \Theta(2^t)$.

Finally, we use the single copy of secure $\mathbb{F}$ to implement $m\sqrt{t} = \Theta(2^t/\sqrt{t}) = \Theta(n/\sqrt{\log n})$ secure random oblivious transfer correlations. So, the production rate is $\Theta(1/\sqrt{\log n})$.

Extracting from a Secure ROLE($\mathbb{F}$). Let us assume that we have extracted one instance of ROLE($\mathbb{F}$). By Chinese Remainder theorem, we get ROLE($\mathbb{Z}_{p_i}$), for all $i \in [m]$. From each ROLE($\mathbb{Z}_{p_i}$) we shall extract $\Theta(\sqrt{t})$ ROLE correlations. This step is explained below.

Let $a = \sum_{i=0}^{q-1} a_i 2^{i(q-1)}$ be a suitable number in binary representation. Let $x = \sum_{i=0}^{q-1} x_i 2^i$ Note that the ax has $a_i x_i$ at iq-th bit position. Now consider b such that all its bits are random except the iq-th bits which are set to $b_i = 0$, $0 \leqslant i < q$.

Note that if $ax + b < p_j$, when $j \in [m]$, then the iq-th bits performs a OLE of (a_i, b_i) and x_i. We set $q = \Theta(\sqrt{t})$ and we are done.

Extracting one Secure ROLE($\mathbb{F}$). The idea is to use a modification of Figure 1 over $\mathbb{F}$ instead of the field $\mathbb{Z}_2$. And use leftover hash lemma (see Lemma 2) to extract one random element of $\mathbb{F}$. The extractor protocol is provided below.

Extract-ROLE (n, t_S, t_R):
Define $g := n - (t_S + t_R)$.
Private Inputs: The clients S and R have private inputs $(s_0, s_1) \in \mathbb{F}^2$ and $c \in \mathbb{F}$, respectively.
Hybrid (Random Correlations): Let $n' = n/t$, where $t = \log \mathbb{F}$. For $i \in [n']$, client S gets random $(a_i, b_i) \in \mathbb{F}^2$ and client R gets (x_i, z_i), such that $x_i \in \mathbb{F}$ is chosen uniformly at random and $z_i = a_i x_i + b_i$.

1. *Random Code Generation.* Client R picks a random matrix G of dimension $k' \times n'$ such that each of its elements is chosen uniformly at random from $\mathbb{F}$. Use $k' = (t_R + g/2)/t$, where $g = n - (t_S + t_R)$. Let $\mathcal{C}$ be the code generate by the generator matrix G; and H be a generator matrix for the dual code $\mathcal{C}^\perp$. If $\mathsf{rank}(G) < k$ or the first column of G or H is all-zero column then abort; otherwise continue.

2. *Random OLE($\mathbb{F}$) Extraction.*

   (a) Client S picks a random $(u_0, \ldots, u_{n'}) \in \mathcal{C}$. Let $\mathcal{C}_{\mathsf{parity}} \subseteq \{0,1\}^{n'+1}$ be the (linear) code consisting of every length $(n+1)$ string in $\mathbb{F}^{n'+1}$ such that their sum is 0. Client S picks a random $(v_0, \ldots, v_n) \in \mathcal{C}_{\mathsf{parity}}$.

   (b) Client R picks a random $(r_0, \ldots, r_n) \in \mathcal{C}^\perp$.

   (c) (In parallel) For each $i \in [n']$, do the following:

      i. Client R sends $m_i = x_i + r_i$. Send m_i to client S.

      ii. Client R sends $\alpha_i = u_i + a_i$ and $\beta_i = a_i m_i + b_i + v_i$.

   (d) Client R computes $t_i = \beta_i + \alpha_i r_i + z_i$ and $z = \sum_{i \in [n']} t_i$.
   Note that $z = ax + b$, for $a = u_0$, $b = v_0$ and $x = r_0$.

Figure 4: Correlation Extractor Protocol which extracts one copy of random Oblivious Linear Function Evaluation (over $\mathbb{F}$) from n' copies of Random Oblivious Linear Functions Evaluations (over $\mathbb{F}$).

It is easy to see that the protocol is correct. Note that $t_i = \beta_i + \alpha_i r_i + z_i = (a_i x_i + a_i r_i + b_i + v_i) + (u_i r_i + a_i r_i) + (a_i x_i + b_i) = u_i r_i + v_i$. Finally, $\sum_{i \in [n']} t_i = \sum_{i \in [n']} u_i r_i + t_i = u_0 r_0 + t_0$.

For the case of privacy of client R against semi-honest client S, we need to show that r_0 is hidden. Interpret the dual matrix H as $[H_0 | H']$, where H_0 is the first column of H. We shall argue for every fixing of H'. Note that H_0 is uniformly random for every fixed H' (because dual of random codes are themselves (close to) random codes). Now, we interpret the code r_0 and $(r_1, \ldots, r_{n'})$ as λH_0

and $\lambda H'$, where λ is a random entry in $\mathbb{F}^{n'+1-k'}$. The sender gets $(r_1, \ldots, r_{[n']}) + (x_1, \ldots, x_{[n']})$, where $\tilde{\mathbf{H}}_\infty(X_{[n']}|L) \geqslant g/2$ (because she only leaks t_S bits from $X_{[n]}$). Now, by direct application of left-over hash lemma (see Lemma 2), we get the privacy of x_0.

For the case of privacy of client S against semi-honest client R, we construct a simulator similar to Figure 3. We need the generalization of Lemma 4 where the matrices are over fields $\mathbb{F}$. The result is true, because it follows directly by application of Lemma 2 instead of Lemma 3. The upper bound on the advantage is bounded by: $\frac{1}{2}\sqrt{\frac{1}{2^{-m+\Theta(n)}}}$, where m is the amount of average min-entropy left in X. Given an adversary $\mathcal{A}$ who distinguishes u_0 from a uniform element in $\mathbb{F}$, we shall construct an adversary $\mathcal{A}'$ who can obtain advantage in the game of Lemma 4 with comparable advantage.

1. $\mathcal{H}$ samples random $a_{[n']} \in \mathbb{F}^{n'}$.
2. $\mathcal{A}'$ samples random $x_{[n']}$ and $z_{[n']}$ in $\mathbb{F}^{n'}$. $\mathcal{A}'$ sends $(x_{[n']}, z_{[n']})$ to $\mathcal{A}$.
3. $\mathcal{A}$ sends leakage function $\mathcal{L}$ which is forwarded to $\mathcal{H}$.
4. $\mathcal{H}$ replies back with the leakage $\mathcal{L}(a_{[n']})$ and is forwarded to $\mathcal{A}$.
5. $\mathcal{A}$ sends $m_{[n']}$.
6. $\mathcal{H}$ sends G and $\alpha_{[n']}$ which is forwarded by $\mathcal{A}'$ to $\mathcal{A}$.
7. $\mathcal{A}'$ sends random $\beta_{[n']}$ in $\mathbb{F}^{n'}$ to $\mathcal{A}$.

The last step of the simulation is perfect because $v_{[n']}$ is uniformly random in $\mathbb{F}^{n'}$. This completes the proof of Theorem 4.

5 Inner Product Correlation

In this section we prove Theorem 5. Our protocol is provided in Figure 5.

When both parties are honest, we need to prove the correctness of the protocol. Which trivially follows.

Sender Corrupt. Suppose a semi-honest client A can leak t bits on information from $(y_{[n]}, b)$. In this case, we have $\tilde{\mathbf{H}}_\infty(Y_{[n]}|L) \geqslant m = n - t$. For security, we need to prove the hiding of the bit r_0 given $r_{[n]} \oplus y_{[n]}$, where $r_{[n]}$ is a uniformly chosen codeword from the image of "H with its first column punctures." Now, we can directly invoke Lemma 4 and get that the distribution $(R_0|\vartheta)$ is $= 2^{-(g/2+1)}$ close to the uniform distribution over $\{0, 1\}$, where $\vartheta$is the view of client A at the end of the protocol and $g = n/2 - t$.

Receiver Corrupt. For this case, we construct a reduction similar to the reduction provided in Figure 3. Again, in this case we assume that client A sends the matrix G instead of client B (which is acceptable because the adversaries are semi-honest). Suppose there exists an adversary $\mathcal{A}$ which can distinguish U_0 from a uniformly random bit with certain advantage. We shall construct

Extract-IP (n):
Hybrid (Random Correlations): Client A gets random $(x_{[n]}, a) \in \{0,1\}^{n+1}$ and client B gets random $(y_{[n]}, b) \in \{0,1\}^{n+1}$, such that $a + b = \langle x_{[n]}, y_{[n]} \rangle$.

1. *Random Code Generation.* Client R picks a binary matrix $G = [I_{k\times k} \| P_{k\times(n+1-k)}]$ of dimension $k \times (n+1)$, where $k = n/2$ and $P_{k\times(n+1-k)}$ is a uniformly chosen random Toeplitz matrix. Let $\mathcal{C}$ be the code generate by the generator matrix G; and H be a generator matrix for the dual code $\mathcal{C}^{\perp}$. If the first column of H is all-zero column then abort; otherwise continue.

2. *Random ROLE Extraction.*

   (a) Client A picks a random $(u_0, \ldots, u_n) \in \mathcal{C}$. Client A picks random $v_0 \in \{0,1\}$.

   (b) Client B picks a random $(r_0, \ldots, r_n) \in \mathcal{C}^{\perp}$.

   (c) Client B sends $m_{[n]} = y_{[n]} \oplus r_{[n]}$ to client A.

   (d) Client A sends $\alpha_{[n]} = x_{[n]} \oplus u_{[n]}$ to client B. Client S sends $\beta = \langle x_{[n]}, m_{[n]} \rangle \oplus a \oplus v_0$.

   (e) Client B computes $z = \beta \oplus b \oplus \langle \alpha_{[n]}, r_{[n]} \rangle$.

   (f) Client A outputs (u_0, v_0) and client B outputs (r_0, z).
   Note that $z = u_0 r_0 \oplus v_0$, because $\langle u_{[n]}, r_{[n]} \rangle = u_0 r_0$.

Figure 5: Random Oblivious Function Evaluation extractor from one Inner Product Correlation over n-bits.

an adversary $\mathcal{A}'$ which uses $\mathcal{A}$ to break the unpredictability experiment of Lemma 4 with identical advantage.

1. $\mathcal{H}$ picks $x_{[n]}$ randomly from $\{0,1\}^n$.
2. $\mathcal{A}'$ picks $y_{[n]}$ randomly from $\{0,1\}^n$ and $b \in \{0,1\}$. It provides $(y_{[n]}, b)$ to the adversary $\mathcal{A}$.
3. $\mathcal{A}$ provides the leakage function $\mathcal{L}$; which $\mathcal{A}'$ forwards to $\mathcal{H}$.
4. $\mathcal{H}$ applies $\mathcal{L}$ on $x_{[n]}$ and obtains the leakage ℓ. $\mathcal{H}$ provides ℓ to $\mathcal{A}'$, who forwards it to $\mathcal{A}$.
5. $\mathcal{H}$ sends G to $\mathcal{A}'$, who forwards it to $\mathcal{A}$.
6. $\mathcal{H}$ sends $\alpha_{[n]} = \lambda G'$, where G' is G with its first column punctured and λ is a uniformly random vector in $\{0,1\}^k$. $\mathcal{A}'$ forwards $\alpha_{[n]}$ to $\mathcal{A}$.
7. $\mathcal{A}'$ simulates β by sampling a uniformly random bit and sends it to $\mathcal{A}$.

Note that this is a perfect simulation of the view of $\mathcal{A}$ because the bit v_0 is uniformly random in the actual protocol. Thus, if $\mathcal{A}$ can distinguish $u_0 = \lambda G_0$ from a uniformly random bit then the adversary $\mathcal{A}'$ can also distinguish λG_0 from a uniformly random bit with identical advantage. By Lemma 4, the advantage is at most $2^{-(g/2+1)}$, where $g = n/2 - t$. This shows that the distribution $(U_0|\vartheta)$ is at most $2^{-(g/2+1)}$ far from the uniform distribution over $\{0,1\}$.

6 Open Problems

As mentioned earlier in the introduction, the fractional leakage thresholds for which correlations can still help perform secure computation is mostly unexplored. Feasibility results exist but are extremely inefficient prior to this work. Impossibility results are even rarer. We summarize some of the the open problems in the semi-honest information theoretic setting. The malicious setting is mostly unexplored.

The biggest problem left open by this work is the construction of oblivious transfer extractors which produce linear number of secure oblivious transfers, if the gap is arbitrary linear function of n. Specific to the oblivious transfer extractor, it will be interesting to reduce the communication complexity of the construction presented in this work. The communication per oblivious transfer produced is linear in n, while it can possibly be reduced to a constant multiplicative overhead (while preserving high fractional leakage resilience and production rate). Further, construction of extractors with high fractional leakage resilience and production rate which is also error tolerant is not known.

Typically, cryptographic constructions which morph one form of correlation into another are amortized, i.e. they use large number of copies of the former to yield large number of copies of the the latter. It is interesting that we do not even know whether one copy of $\mathrm{ROLE}(\mathbb{GF}[2^k])$ can be used to obtain $\Theta(k)$ copies of secure OLE. In this paper, we show that $\Theta(\sqrt{k})$ copies can be securely realized. An improvement in understanding of this problem will have direct consequence on the production rate of Theorem 4.

Finally, in the setting of secure computation in presence of leakage on general correlations, it is unclear whether the fractional leakage resilience can be higher than 1/2. We conjecture that $t_S/s \geqslant 1/2$ and $t_R/s \geqslant 1/2$ is impossible.

References

[AR94] Noga Alon and Yuval Roichman. Random cayley graphs and expanders. *Random Struct. Algorithms*, 5(2):271–285, 1994. 8

[BBCM95] Charles H. Bennett, Gilles Brassard, Claude Crépeau, and Ueli M. Maurer. Generalized privacy amplification. *IEEE Transactions on Information Theory*, 41(6):1915–1923, 1995. 1

[BBR88] Charles H. Bennett, Gilles Brassard, and Jean-Marc Robert. Privacy amplification by public discussion. *SIAM J. Comput.*, 17(2):210–229, 1988. 1

[Bea89] Donald Beaver. Perfect privacy for two-party protocols. In Joan Feigenbaum and Michael Merritt, editors, *Proceedings of DIMACS Workshop on Distributed Computing and Cryptography*, volume 2, pages 65–77. American Mathematical Society, 1989. 1

[CLOS02] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In John H. Reif, editor, *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*, pages 494–503. ACM, 2002. 1

[DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008. 6

[DS05] Yevgeniy Dodis and Adam Smith. Correcting errors without leaking partial information. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 654–663. ACM, 2005. 8

[Gop81] Valerii Denisovich Goppa. Codes on algebraic curves. In *Soviet Math. Dokl*, pages 170–172, 1981. 1, 3

[GS96] Arnaldo Garcia and Henning Stichtenoth. On the asymptotic behaviour of some towers of function fields over finite fields. *Journal of Number Theory*, 61(2):248–273, 1996. 1, 3

[GW97] Oded Goldreich and Avi Wigderson. Tiny families of functions with random properties: A quality-size trade-off for hashing. *Random Struct. Algorithms*, 11(4):315–343, 1997. 8

[HIKN08] Danny Harnik, Yuval Ishai, Eyal Kushilevitz, and Jesper Buus Nielsen. Ot-combiners via secure computation. In Ran Canetti, editor, *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, volume 4948 of *Lecture Notes in Computer Science*, pages 393–411. Springer, 2008. 4

[HKN+05] Danny Harnik, Joe Kilian, Moni Naor, Omer Reingold, and Alon Rosen. On robust combiners for oblivious transfer and other primitives. In Ronald Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 96–113. Springer, 2005. 4

[IKOS09] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Extracting correlations. In *FOCS*, pages 261–270. IEEE Computer Society, 2009. 1, 2, 3, 4

[IMSW14] Yuval Ishai, Hemanta K. Maji, Amit Sahai, and Jürg Wullschleger. Single-use ot combiners with near-optimal resilience. In *Proceedings of the 2014 IEEE International Symposium on Information Theory, Honolulu, USA, June 29 - July 4, 2014*. IEEE, 2014. 1, 2, 3, 4, 5, 9

[IPS08] Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding cryptography on oblivious transfer - efficiently. In David Wagner, editor, *CRYPTO*, volume 5157 of *Lecture Notes in Computer Science*, pages 572–591. Springer, 2008. 1, 4

[Kil88] Joe Kilian. Founding cryptography on oblivious transfer. In *STOC*, pages 20–31, 1988. 1

[Kil00] Joe Kilian. More general completeness theorems for secure two-party computation. In *STOC*, pages 316–324, 2000. 1, 4

[Kus89] Eyal Kushilevitz. Privacy and communication complexity. In *FOCS*, pages 416–421. IEEE, 1989. 1

[Mas95] James Lee Massey. Some applications of coding theory in cryptography. In *Codes and Ciphers: Cryptography and Coding IV*, pages 33–47, 1995. 4

[MP06] Remo Meier and Bartosz Przydatek. On robust combiners for private information retrieval and other primitives. In Cynthia Dwork, editor, *Advances in Cryptology - CRYPTO 2006, 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006, Proceedings*, volume 4117 of *Lecture Notes in Computer Science*, pages 555–569. Springer, 2006. 4

[MPR09] Hemanta K. Maji, Manoj Prabhakaran, and Mike Rosulek. Complexity of multi-party computation problems: The case of 2-party symmetric secure function evaluation. In Omer Reingold, editor, *TCC*, volume 5444 of *Lecture Notes in Computer Science*, pages 256–273. Springer, 2009. Full version available from IACR Eprint Archive: http://eprint.iacr.org. 1

[MPW07] Remo Meier, Bartosz Przydatek, and Jürg Wullschleger. Robuster combiners for oblivious transfer. In Salil P. Vadhan, editor, *Theory of Cryptography, 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007, Proceedings*, volume 4392 of *Lecture Notes in Computer Science*, pages 404–418. Springer, 2007. 4

[NN90] Joseph Naor and Moni Naor. Small-bias probability spaces: Efficient constructions and applications. In Harriet Ortiz, editor, *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 213–223. ACM, 1990. 8

[NNOB12] Jesper Buus Nielsen, Peter Sebastian Nordholt, Claudio Orlandi, and Sai Sheshank Burra. A new approach to practical active-secure two-party computation. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, volume 7417 of *Lecture Notes in Computer Science*, pages 681–700. Springer, 2012. 1

[PW08] Bartosz Przydatek and Jürg Wullschleger. Error-tolerant combiners for oblivious primitives. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part II - Track B: Logic, Semantics, and Theory of Programming & Track C: Security and Cryptography Foundations*, volume 5126 of *Lecture Notes in Computer Science*, pages 461–472. Springer, 2008. 4

[Vad04] Salil P. Vadhan. Constructing locally computable extractors and cryptosystems in the bounded-storage model. *J. Cryptology*, 17(1):43–77, 2004. 15

A Mathematical Tools

A.1 Toeplitz Matrices

Lemma 6 (Toeplitz Property)**.** *Let c be a fixed binary column of length k and $T \subseteq [n]$ be a set of indices. For a random binary Toeplitz matrix P of dimension $k \times (n+1-k)$ and $G := [I_{k\times k} \parallel P]$, the probability that:*

1. *$\sum_{i\in T} G_i = c$ is at most 2^{-k}, and*
2. *There exists $T' \subseteq T$ such that $\sum_{i\in T'} G_i = c$ is at most $2^{-k+|T|}$.*

Proof. We prove this using a sequence of observations.

Note that: $G_i = c$, for $i > k$, happens with probability 2^{-k}.

Next, we claim that: $G_i + G_j = c$, for $i > j > k$, happens with probability 2^{-k}. This is so because the probability that the $G_{i,k} + G_{j,k} = c_k$ happens with probability $1/2$. Fixing the values of $G_{i,k}$ and $G_{j,k}$, the probability that we have $G_{i,k-1} + G_{j,k-1} = c_{k-1}$ is $1/2$; because the random variable $G_{j,k-1}$ is not fixed (since P is a Toeplitz matrix). Extending this argument, we get the claim.

Similarly, we claim that: $\sum_{i\in T':i>k} G_i = c + \sum_{i\in T':i\leqslant k} G_i$ with probability 2^{-k}, for any $T' \subseteq [n]$. That is: $\sum_{i\in T'} G_i = c$ with probability at most 2^{-k}.

Using union bound over $T' \subseteq T$, we get: the probability that there exists $T' \subseteq T$ such that $\sum_{i\in T'} G_i = c$ is at most $2^{-k+|T|}$. □