Secure Computation from Leaky Correlated Randomness

Divya Gupta[1], Yuval Ishai[2], Hemanta K. Maji[1,3(✉)], and Amit Sahai[1]

[1] University of California, Los Angeles and Center for Encrypted Functionalities, Los Angeles, USA
{divyag,sahai}@cs.ucla.edu
[2] Technion, Haifa, Israel
yuvali@cs.technion.ac.il
[3] Purdue University, West Lafayette, USA
hemanta.maji@gmail.com

Abstract. Correlated secret randomness is an essential resource for information-theoretic cryptography. In the context of secure two-party computation, the high level of efficiency achieved by information-theoretic protocols has motivated a paradigm of starting with correlated randomness, specifically random oblivious transfer (OT) correlations. This correlated randomness can be generated and stored during an offline preprocessing phase, long before the inputs are known. But what if some information about the correlated randomness is leaked to an adversary or to the other party? Can we still recover "fresh" correlated randomness after such leakage has occurred?

This question is a direct analog of the classical question of privacy amplification, which addresses the case of a *shared* random secret key, in 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 question of how much leakage is tolerable when recovering OT correlations has remained wide 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. In our work, we show that starting with n

D. Gupta, H.K. Maji, A. Sahai–Research supported in part from a DARPA/ONR PROCEED award, a DARPA/ARL SAFEWARE award, NSF Frontier Award 1413955, NSF grants 1228984, 1136174, 1118096, and 1065276, a Xerox Faculty Research Award, a Google Faculty Research Award, an equipment grant from Intel, and an Okawa Foundation Research Grant. This material is based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014-11-1-0389. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense, the National Science Foundation, or the U.S. Government.
Y. Ishai–Research supported by the European Union's Tenth Framework Programme (FP10/2010-2016) under grant agreement no. 259426 ERC-CaC, ISF grant 1709/14 and BSF grant 2012378.

R. Gennaro and M. Robshaw (Eds.): CRYPTO 2015, Part II, LNCS 9216, pp. 701–720, 2015.
DOI: 10.1007/978-3-662-48000-7_34

random 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 can tolerate up to $1/2 - \epsilon$ fractional leakage, for any $\epsilon > 0$ (compared to the optimal $1/4$ fractional leakage for OT correlations).

1 Introduction

Secure two-party computation [17,39] allows two mutually distrusting parties to perform secure computation using their private inputs without revealing any extra information to each other. 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 [2,3,27,29,30]. For example, even the seemingly simple task of securely computing the AND of two bits is not possible. On the other hand, if suitable correlated randomness is provided as setup to the parties, then general secure two-party computation becomes possible [8,26,28]. A particularly useful type of correlated randomness is the random oblivious transfer (OT) correlation, where the sender gets two random bits (s_0, s_1) and the receiver gets (c, s_c), where c is a random bit.

One reason for the usefulness of OT correlations is the existence of highly efficient OT-based secure computation protocols both in theory and in practice. Indeed, protocols such as TinyOT [35] have popularized the approach of starting with random bit OT correlations for obtaining practically efficient secure computation protocols. Random OT correlations can be distributed or securely generated in an offline phase, long before the inputs are known, and later used in an online phase to perform a desired secure computation. But what if some information about the correlated randomness is leaked to an adversary? Can we still extract "fresh" correlated randomness after such leakage has occurred?

This question is a direct analog of the classical question of privacy amplification [4,5] that arose in the context of secure *communication*. Privacy amplification asks the following question: given shared secret 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 useful for secure *computation*. Note, however, that participants in a privacy amplification protocol protect their secret only from an outsider. Instead, in our setting, each party must protect its 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.

Quite surprisingly, very little is known about our question. This is in sharp contrast to the problem of privacy amplification, and despite decades of study of

OT-based secure computation. In particular, the question of how much leakage can be tolerated when recovering OT correlations has remained wide open. In our work, we resolve this question.

Prior to our work, Ishai, Kushilevitz, Ostrovsky and Sahai [24] studied this question, introducing the notion of correlation extractors. Concretely, they consider the setting of extracting fresh OT correlations from n independent copies of random OT correlations that have been subject to leakage. (One can either consider a deterministic leakage, captured by an arbitrary function f with t bits of output, or general probabilistic leakage, subject to the constraint that the secret has expected min-entropy of t bits conditioned on the leakage.) The main result of [24] is an interactive protocol for extracting OT correlations that remains secure even when some constant fraction of the $2n$ secret bits of each party can be leaked to the other party. Unfortunately, the concrete fractional leakage tolerated by this protocol is extremely small, approximately 10^{-7}. So, at best, this result serves as a proof of concept.

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, one can tolerate $(1-\varepsilon)n/2$ bits of leakage, for an arbitrarily small constant $\varepsilon > 0$, with negligible error. This leakage rate is near-optimal [25]. Moreover, in contrast to the previous protocol of [24], our protocol uses a minimal amount of interaction, requiring only two messages as opposed to the 4 messages that are inherently required by the technique from [24]. Finally, our protocol is conceptually simpler, completely avoiding the use of Algebraic-Geometric codes [16,19] needed in [24] and replacing them with simple families of binary linear codes.

Having settled 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. We answer this question in the affirmative. We show that the so-called inner product correlation, where parties receive random binary vectors and additive shares of their inner product, can tolerate a significantly higher fractional leakage. Concretely, we show how to extract a fresh OT from such an inner product correlation while tolerating up to $1/2-\epsilon$ fractional leakage, for any $\epsilon > 0$ (compared to the optimal leakage rate of $1/4$ for OT correlations). This opens up a new set of questions to explore in future work (for more discussion refer to the full version of the paper [20]).

Finally, we note that while the primary focus of this work is on the information-theoretic setting for secure computation, the problem we consider is well motivated even in the setting of computational security. The reason is twofold. First, the fresh OTs produced by our extraction procedures can be used by computationally secure OT-based protocols such as those based on garbled circuits [39]. Second, these extraction procedures can be applied even when a

computationally secure protocol is used for realizing the offline generation of correlated randomness. Suppose that a computationally secure two-party protocol Π is used for this purpose. If the leakage occurs after the execution of Π terminates (and the two parties erase everything but the output), then our protocols are guaranteed to produce clean OTs that can be consumed by subsequent (computational or information-theoretic) protocols. Moreover, if Π is so-called "leakage-tolerant" [6,7,21], then the same holds even if leakage can occur *during* the execution of Π. Such leakage-tolerant protocols can be constructed under standard intractability assumptions.

1.1 Our Contribution

In this section we give a more detailed overview of our main results.

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 primitive where client S receives random bits (s_0, s_1); and the client R receives random bit c and s_c. Random oblivious transfer correlations can be easily converted into standard oblivious transfers, where a receiver R selects one of two bits held by a sender S. The latter can serve as a basis for general secure multiparty computation.

More concretely, Oblivious Transfer (OT) is a two-party functionality where client S (sender) has inputs (s_0, s_1), client R (receiver) has input c, and client R obtains output s_c. A Random OT correlation, referred to as ROT, provides (s_0, s_1) to one party and (c, s_c) to the other party, where s_0, s_1, c are uniform random bits. We work in the ROT^n-hybrid model, 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, where by default we define t bits of leakage as the output of an adversarially chosen function with t output bits. (However, all of our results extend to leakage measured in terms of average conditional min-entropy.) 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 $(1-\varepsilon)$-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 a 2-message $(n, t_S, t_R, \varepsilon)$ OT Extractor protocol which produces a secure OT, such that $\varepsilon \leq 2^{-(g/4)+1}$ where $g := n - (t_S + t_R)$.*

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 [24] who can tolerate leakage up to 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. Moreover, our leakage resilience is (near) optimal due to the negative result of [25]. The negative result states that there does not exist any OT combiner (let alone an extractor) which can tolerate up to $n/2 - O(1)$ bits of leakage. Our protocol also improves upon the round complexity of [24] from 4 messages to 2 messages, which are clearly necessary.

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 $n, t_S, t_R \in \mathbb{N}$, such that $g = n - (t_S + t_R) = \Theta(n)$, and $\rho = \omega(\log n)$, there exists a 2-message (n, t_S, t_R) OT Extractor with production $p = n/\rho$ and $\varepsilon \leq \mathsf{negl}(n)$.*

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 sub-constant, 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 [24], 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 hidden constant in our asymptotic production rate is small, say upper bounded by 10^{-1}. So, in concrete terms, our production rate is $\sim (g/n)/10 \log^2 n$, which is higher than the rate achieved by [24] for a practical range of parameters (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 OTs even if the gap is an arbitrarily small linear function of n.

Overall, our construction significantly simplifies the prior construction of [24] at a conceptual level by forgoing usage of Algebraic Geometric [16,19] codes and instead relying on binary linear codes generated by generator matrices whose parity check matrices are random Toeplitz matrices.

Unlike [24], 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 $|T| \leq t_S$ and client R can send $|T| \leq t_R$. The leakage provided is (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 all bits can be reconstructed from input and one bit leakage).

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 a 2-message $(n, t_S, t_R, \varepsilon)$ OT-Combiner which produces one secure OT using $O(n)$ bits of communication, where $\varepsilon \leq 2^{-g/2}$ and $g := n - (t_S + t_R)$.*

Note that the construction presented in [25] achieves similar bounds but the communication complexity in their construction is quadratic in n; while ours is linear in n. We emphasize that the higher production result of Theorem 2 also applies to the setting of combiners.

Large Correlations. We show that, in fact, there are correlations that can tolerate a fractional leakage close to 1/2.

Theorem 4 (High Tolerance). *For any $s, t \in \mathbb{N}$, there exists a correlation (X, Y) over a pair of $(s+1)$-bit strings such that, even after any party leaks t bits on the correlation (X, Y), they can securely realize OT using 2-message communication with simulation error $\varepsilon \leq 2^{-(g/2+1)}$, where $g := s/2 - t$.*

The correlation (X, Y) used to prove the above theorem is the so-called *inner-product correlation*, where each party receives a random s-bit vector and the mod-2 inner product of the two vectors is secret-shared between the parties. Moreover, it is not hard to show that our protocol cannot tolerate leakage rate bigger than 1/2. We leave open the question whether the 1/2 leakage rate is optimal for arbitrary correlations.

1.2 Prior Related Works

Most relevant work to our work is the work of Ishai, Kushilevitz, Ostrovsky, and Sahai [24], 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.

A closely related concept is the notion of OT combiners, which are a restricted variant of OT extractors in which leakage is limited to local information about individual OT correlations and there is no global leakage. The study of OT combiners was initiated by Harnik et al. [23]. Since then, there has been work on several variants and extensions of OT combiners [22,26,32,33,37].

Recently, [25] constructed OT combiners with nearly optimal leakage parameters. Our protocols were inspired by the OT combiners from [25], but the results we achieve are stronger in several imporant ways. First, whereas [25] only considers t *physical* bits of leakage, we tolerate a arbitrary bits of leakage (similarly to [24], though with a much better leakage rate). Second, even in the case of physical leakage, our solutions improve over [25] by reducing the communication and randomness complexity from quadratic to linear. Finally, our

protocols can be used to produce a near-linear number of OTs without significantly compromising the leakage rate, whereas [25] only considers the case of producing a single OT.

Another related work is that of Dziembowski and Faust [14] which (similarly to our Theorem 4) obtains some form of leakage-resilient secure computation from the inner product correlation. However, the construction from [14] requires multiple independent instances of an inner product correlation even for producing just a single OT, and moreover the model considered in [14] assumes that the leakage applies *individually* to each instance. Even if the analysis of [14] could somehow be strengthened to tolerate some amount of global leakage, the tolerable leakage rate must inevitably be small (since even with a leakage rate of 1/4, one can entirely compromise one of the inner product instances in [14]). Thus, the approach of [14] does not seem relevant to our goal of maximizing the leakage rate.

1.3 Technical Overview

We provide a short overview of our construction which proves Theorem 1. Our construction is inspired by the Massey secret sharing scheme [31]. Our construction is closely related to the constructions of [24,25]. 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 [25] which considers physical bits of leakage) 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 [25].

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. In particular, $u_0 r_0 = \sum_{i \in [n]} u_i r_i$. 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 r_0 can be abstracted out as follows: For λ uniform random vector, given $(\lambda H' \oplus x_{[n]}, H)$, 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 codes of a fixed binary linear codespace $\mathcal{C}^{\perp}$ 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 random linear code or $\mathcal{C}_I$ 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 disjoint and each S_i indexes a set of servers. One OT is extracted by applying Theorem 1 on each index set S_i.

2 Preliminaries

Notations. We represent random variables by capital letters, for example X, and the values they take by small letters, for example $\Pr[X = x]$. The set $\{1, \ldots, n\}$ is represented by $[n]$, for $n \in \mathbb{N}$. Given a vector $v = (v_1, \ldots, v_n)$ and $T = \{i_1, \ldots, i_{|T|}\} \subseteq [n]$, we represent $(v_{i_1}, \ldots, v_{i_{|T|}})$ by v_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. The support of a probability distribution X, represented as $\mathsf{Supp}(X)$ is the set of elements in the sample space which are assigned non-zero probability by X. A uniform distribution over a set S is represented by $\mathbf{U}_S$. A probability distribution X over a universe U is a flat source if there exists a constant $c \in (0, 1]$ such that $\Pr[X = x]$ is either 0 or c, for all $x \in U$. Further, we say that X is a flat-source of size $1/c$. Given a joint distribution (X, Y) over sample space $U \times V$, the conditional distribution $(X|y)$ represents the distribution over sample space U such that the probability of $x \in U$ is $\Pr[X = x|Y = y]$.

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

Entropy Definitions. For a probability distribution X over a sample space U, we define entropy of x as $H_X(x) := -\lg \Pr[X = x]$, 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 $\mathbf{H}_\infty(X) \geq n$, then X can be written as convex linear combination of distributions, each of which are flat sources of size $\geq 2^n$. The average min-entropy [10], represented by $\widetilde{\mathbf{H}}_\infty(X|Y)$, is defined to be $-\lg \mathbb{E}_{y \sim Y} \left[2^{-\mathbf{H}_\infty(X|y)}\right]$. Following lemma is useful for lower bounding average min-entropy after leakage on a high min-entropy source.

Lemma 1 (Chain Rule [10]). *If* $\mathbf{H}_\infty(X) \geq n$ *and* L *be arbitrary* ℓ*-bit leakage on* X*, then* $\widetilde{\mathbf{H}}_\infty(X|L) \geq n - \ell$.

2.1 Elementary Fourier Analysis

We define character $\chi_S(x) = (-1)^{\sum_{i \in S} x_i}$, 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 $f(x) = \Pr[M = x]$.

Definition 1 (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:*

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

Definition 2 (Small-bias Distribution Family [11]). *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:*

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

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

Lemma 2 (Min-entropy Extraction [1,11,18,34]). *Let* $\mathcal{F} = \{F_1, \ldots, F_\mu\}$ *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* $\widetilde{\mathbf{H}}_\infty(M|L) \geq m$*. Then, the following holds:*

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

where I *is a uniform distribution over* $[\mu]$.

2.2 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($\mathbb{F}$). A random oblivious linear-function evaluation (ROLE) can be defined analogous to ROT.

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

Random Inner Product Correlation. This is an input-less two party functionality which samples $x_{[n]}, y_{[n]} \xleftarrow{\$} \{0,1\}^n$, $a \xleftarrow{\$} \{0,1\}$ and $b = a + \langle x_{[n]}, y_{[n]} \rangle$. It outputs $(x_{[n]}, a)$ to party A and $(y_{[n]}, b)$ to party B. Note that for $n = 1$, this is equivalent to random oblivious transfer correlation and oblivious linear function evaluation.

2.3 Combiners and Extractors

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

Definition 3 ($(n, p, t_S, t_R, \varepsilon)$ (Single Use) OT-Combiner). *An $(n, p, t_S, t_R, \varepsilon)$ (single use) OT-Combiner 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 on inputs from S and R. We consider a semi-honest adversary who can either corrupt the client S and t_S servers or client R and t_R servers. The protocol implements p independent copies of secure oblivious transfer instances with correctness and simulation error at most ε.*

The correctness conditions for the protocol says that the receiver's output is correct in all p-instances of OT with probability at least $(1 - \varepsilon)$.

The privacy requirement says that the adversary should not learn more than it should. Let $(s_0^{(i)}, s_1^{(i)})$ and $c^{(i)}$ be the inputs of the sender and the receiver, respectively, in i^{th} copy of OT produced. Then a corrupt sender (resp., corrupt receiver) cannot output $c^{(i)}$ (resp., $s_{1-c}^{(i)}$) with probability more than $\frac{1}{2} + \varepsilon$ for any instance of OT produced.

Leakage Model and Correlation Extractors. Here we begin by describing our leakage model for ROLE correlations formally followed by defining correlation extractors for OLE. Recall that OT and OLE are just local renaming of each other. Our leakage model is as follows:

1. **n-Random OLE Correlation Generation phase:** For $i \in [n]$, the sender S gets random $(a_i, b_i) \in \{0,1\}^2$ and receiver R gets (x_i, z_i), where $x_i \in \{0,1\}$ is chosen uniformly at random and $z_i = a_i x_i + b_i$.
2. **Corruption and leakage phase.** A semi-honest adversary corrupts either the sender and sends a leakage function $L : \{0,1\}^n \to \{0,1\}^{t_S}$. It receives $L(\{x_i\}_{i \in [n]})$. Or, it corrupts the receiver and send a leakage function $L : \{0,1\}^n \to \{0,1\}^{t_R}$. It receives $L(\{a_i\}_{i \in [n]})$.

   Note that without loss of generality any leakage on sender (resp., receiver) can be seen as a leakage on $\{a_i\}$ (resp., $\{x_i\}$).

Let (X, Y) be the random OT correlation. We denote (t_S, t_R)-leaky version of $(X, Y)^n$ described above as $((X, Y)^n)^{[t_S, t_R]}$.

Definition 4 ($(n, p, t_S, t_R, \varepsilon)$ OT-Extractor). *An $(n, p, t_S, t_R, \varepsilon)$ OT-Extractor is an interactive protocol between two parties S and R in the $((X, Y)^n)^{[t_S, t_R]}$ hybrid described above. The protocol implements p independent copies of secure oblivious transfer instances with simulation error ε.*

The correctness and privacy requirements are same as those defined above for $(n, p, t_S, t_R, \varepsilon)$ (Single Use) OT-Combiner.

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.

Remark: An $(n, p, t_S, t_R, \varepsilon)$ OT extractor is also an $(n, p, t_S, t_R, \varepsilon)$ OT combiner.

Noisy Leakage Model. The leakage model described above is referred to as the "bounded leakage" model since we restrict the number of bits output by the leakage function. But this model is sometimes too restrictive and does not capture many side channel attacks, which are the main cause of leakage in real world applications. A more realistic assumption one can make is to assume that leakages are sufficiently noisy. It is observed via experiments that the real-world physical leakages are inherently noisy. There have been many works trying to model noisy leakage and present solutions in this setting [9,12,13,15,36]. At a high level the noisy feature of a leakage function f is captured by assuming that an observation of $f(x)$ only implies a *bounded bias* in the probability distribution of x. More formally, f is said to be δ-*noisy* if

$$\delta = \mathrm{SD}\left((X), (X|f(X))\right).$$

Note that if $\mathbf{H}_\infty(X) \geq n$ then for any $k < n$, we can choose appropriate δ, such that $\widetilde{\mathbf{H}}_\infty(X|f(X)) \geq k$, where f is a δ-noisy channel.

We emphasize that all our protocols only rely on the fact that the initial correlation given to any party has high average min-entropy ($\widetilde{\mathbf{H}}_\infty$) after the leakage. Hence, all our protocols directly work even in the general setting of noisy leakage.

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 defined by its first row and first column. We shall consider uniform distribution 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 proved in Lemma 3.

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.

For a given $G \in \mathbb{T}_{(k,n)}$, the distribution F_G corresponds to a uniform distribution over the codewords generated by G. We have the following lemma, which will be used to prove the main unpredictability lemma in the next section.

Lemma 3. *For the distribution of matrices $\mathbb{T}_{(k,n)}$, the following holds. For any $\emptyset \neq T \subseteq [n]$,*

$$\mathbb{E}_{G\xleftarrow{\$}\mathbb{T}_{(k,n)}}\left[\mathsf{Bias}_T(F_G)^2\right] \leq 2^{-k}$$

Proof. Since F_G is a uniform distribution of codewords over a linear code, for any G, either $\mathsf{Bias}_T(F_G)^2$ is either 0 or 1. Moreover, $\mathsf{Bias}_T(F_G)^2 = 1$ if and only if $\sum_{i\in T} G_i = 0^k$. Hence, it suffices to show the following: For any fixed column $c \in \{0,1\}^k$ and non-empty set $T \subseteq [n]$, $\Pr[\sum_{i\in T} G_i = c] \leq 2^{-k}$. 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 (columns $\{k+1,\ldots,n\}$ form a Toeplitz matrix). Extending this argument, we get for any $T' \subseteq \{k+1,\ldots,n\}$, $\Pr[\sum_{i\in T'} G_i = c] \leq 2^{-k}$.

To prove full claim, note that $\Pr[\sum_{i\in T} G_i = c] = \Pr[\sum_{i\in T:i>k} G_i = c + \sum_{i\in T':i\leq k} G_i] \leq 2^{-k}$ using the above conclusion.

3 Unpredictability Lemma

In this section we present the main 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 $\mathcal{H}$ and an adversary $\mathcal{A}$:*

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 $\mathcal{A}$.*
4. *$\mathcal{A}$ outputs a bit $\tilde{y}$.*

The adversary $\mathcal{A}$ wins the game if $y_0 = \tilde{y}$. For any $\mathcal{A}$, the advantage of the adversary, i.e. $\mathsf{Adv}(\mathcal{A}) = \Pr(y_0 = \tilde{y}) - 1/2 \leq \frac{1}{2}\sqrt{\frac{2}{2^{k-t}}}$.

Proof. Let $\mathbb{G}$ be the distribution $\mathbb{T}_{(k,n+1)}$. The proof for the other case will work similarly.

Given a $G \in \mathbb{G}$, the distribution F_G corresponds to a uniform distribution over the codewords generated by G. Note that over choice of G, they form a $\delta^2 = 2^{-k}$ biased family of distributions (by Lemma 3).

By Lemma 1, $\widetilde{\mathbf{H}}_\infty(M_{[n]}|\mathcal{L}(M_{[n]})) \geq n - t$. Let $M = (0, M_{[n]})$, then putting these in Lemma 2, we get

$$\mathrm{SD}\left((F_G \oplus M, L, G), \left(U_{\{0,1\}^{n+1}}, L, G\right)\right) \leq \frac{1}{2}\sqrt{\frac{2^{n+1}}{2^{k+n-t}}}$$

The lemma follows by noting that $\mathsf{Adv}(\mathcal{A}) \leq \mathrm{SD}\Big((F_G \oplus M, L, G), \left(U_{\{0,1\}^{n+1}}, L, G\right)\Big)$.

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

4 Oblivious Transfer Extractor

4.1 Extracting One Oblivious Transfer

In this section, we shall prove Theorem 1 by presenting 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 [25]. But we deal with general leakage, instead of restricted leakage of physical bits in the combiner setting, using more sophisticated analysis tools. We also achieve lower communication complexity. In particular, we improve the communication complexity from $\Theta(n^2)$ in [25] to $\Theta(n)$ in the current work. When analyzed appropriately for the combiner setting, our current protocol 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) and client S replies with one message (step 2.d).

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 generated 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) For each $i \in [n]$, client R sets $m_i = x_i \oplus r_i$. Client R also sets $m = r_0 \oplus c$. Client R sends $(\{m_i\}_{i\in[n]}, m)$ to client S.
   (d) For each $i \in [n]$, client S sets $\alpha_i = a_i \oplus u_i$ and $\beta_i = a_i m_i \oplus b_i \oplus v_i$. Client S also sets $\alpha = u_0 \oplus s_0$ and $\beta = u_0 m \oplus v_0 \oplus s_1$. Client S sends $(\{(\alpha_i, \beta_i)\}_{i\in[n]}, \alpha, \beta)$ to client R.
   (e) Client R computes $t_i = \beta_i \oplus \alpha_i r_i \oplus z_i$ and $z = \oplus_{i\in[n]}\, t_i$. Finally, client R outputs $y = \beta \oplus \alpha c \oplus z$.

Fig. 1. Round optimal correlation extractor protocol which extracts one copy of Oblivious Linear Function Evaluation from n copies of Random Oblivious Linear Functions Evaluations.

No Corruption Case. We will first prove the correctness of the protocol presented in Fig. 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. Following lemma proves correctness.

Lemma 5. *In the protocol in Fig. 1 the client R outputs $y = s_0 c \oplus s_1$.*

Proof. We first show that $t_i = u_i r_i \oplus v_i$.

$$\begin{aligned} t_i &= \beta_i \oplus \alpha_i r_i \oplus z_i = (a_i m_i \oplus b_i \oplus v_i) \oplus (a_i r_i \oplus u_i r_i) \oplus z_i \\ &= a_i x_i \oplus a_i r_i \oplus b_i \oplus v_i \oplus a_i r_i \oplus u_i r_i \oplus a_i x_i \oplus b_i = u_i r_i \oplus v_i \end{aligned}$$

This shows that $z = \oplus_{i\in[n]} t_i = u_0 r_0 \oplus v_0$. This follows from $\oplus_{i=0}^{n} u_i \cdot r_i = 0$ and $\oplus_{i=0}^{n} v_i = 0$. Now for y we have the following:

$$\begin{aligned} y &= \beta \oplus \alpha c + z = (u_0 m \oplus v_0 \oplus s_1) \oplus (u_0 c \oplus s_0 c) \oplus z \\ &= u_0 r_0 \oplus u_0 c \oplus v_0 \oplus s_1 \oplus u_0 c \oplus s_0 c \oplus u_0 r_0 \oplus v_0 = s_0 c \oplus s_1. \end{aligned}$$

Sender Privacy and Receiver Privacy. In order to give a modular analysis, we consider a simpler protocol of 4-rounds which is equivalent to the protocol presented in Fig. 1. In the simpler protocol, the first two rounds correspond to ROLE extraction, where the receiver sends the messages $\{m_i\}_{i\in[n]}$ and receives $\{(\alpha_i, \beta_i)\}_{i\in[n]}$ and computes ROLE $z = u_0 r_0 \oplus v_0$. In the following, we will refer to this as *ROLE extraction phase*. In the next two rounds, it uses this ROLE to compute the OLE on inputs s_0, s_1, c as follows: Receiver sends message m and gets back α, β and computes y. Note that since we only consider semi-honest adversaries and leakage only occurs before the start of the protocol, these two protocols are equivalent in correctness and security guarantees.

Below, in order to prove the sender and receiver privacy we analyze this protocol. For security of both sides, it is sufficient to prove that extracted ROLE is secure in first phase.

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. We note that it suffices to show that at the end of the ROLE extraction phase (described above), the 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 $X_{[n]}$.

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

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

Below we show that for any semi-honest client S, we have $\Pr(S(\vartheta) = r_0)$ is at most $1/2 + 2^{-g/4-1}$.

Note that $\Pr(S(\vartheta) = r_0) = \Pr(S(H, m_{[n]}, L) = r_0)$, where H is the generator matrix for $\mathcal{C}^{\perp}$. Recall that $H \in \mathbb{T}_{\perp,(n+1-k,n+1)}$, where $k = t_R + g/2$. In Fig. 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: $\widetilde{\mathbf{H}}_{\infty}(X_{[n]}|L) \geq m = (n - t_S)$. By Lemma 4, the advantage of predicting $\langle H_0, w \rangle$ is at most: $2^{-g/4-1}$.

Sender Privacy. In order to prove sender privacy for Fig. 1, we need to show that the bit s_0 is hidden from the receiver after the protocol. Note that it suffices to show that at the end of the ROLE extraction phase (for the simpler protocol described above) bit u_0 is hidden.

Let L denote the random variable for leakage on vector $a_{[n]}$ obtained by the semi-honest adversary who corrupts the receiver after the random correlation generation phase. 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 $\widetilde{\mathbf{H}}_\infty(A_{[n]}|L) \geq m = n - t_R$.

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

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

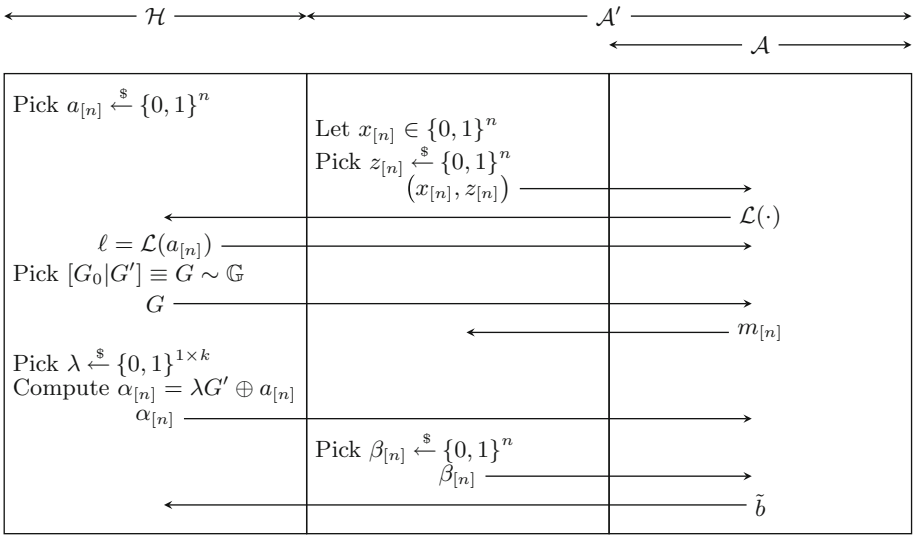

Fig. 2. Simulator for Sender Privacy. The distribution $\mathbb{G}$ is uniform distribution over $k \times (n+1)$ binary matrices in systematic form whose parity check matrices are uniform Toeplitz matrices.

Let U_0 denote the random variable for u_0. We are interested in the conditional distribution $(U_0|\vartheta)$. Below we will show that for any semi-honest client R, $\Pr(R(\vartheta) = u_0)$ is at most $1/2 + 2^{-(g/4)}$.

We show this via a reduction to Lemma 4 in Fig. 2. Given any adversary $\mathcal{A}$ who can predict u_0, we convert it into an adversary $\mathcal{A}'$ against the honest challenger $\mathcal{H}$ 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: $2^{-(g/4)}$.

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 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 disjoint correlations and, subsequently, run the protocol in Fig. 1 on those subsets independently. This increases the simulation error (due to smaller number of OTs used to output each fresh OT i.e. smaller value of n), 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 [38]). The sophisticated techniques of [38] 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 $c \leq 1$. In general c could have been a function of n, but we forgo those cases. The main technical lemma is the following:

Lemma 6 (Sub-sampling [38]). *Let $(A_{[n]}, L)$ be a joint distribution such that, there exists a constant $\mu \in (0,1)$ such that, $\widetilde{\mathbf{H}}_\infty(A_{[n]}|L) \geq \mu n$. For every constant $\varepsilon \in (0, \mu)$ 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 $|S_i| = \lambda\rho$. We have $S_i \cap S_j = \emptyset$, for all $i, j \in [m]$ and $i \neq j$.*
2. *High Entropy: $\widetilde{\mathbf{H}}_\infty(S_{i+1}|S_{[i]}, L) \geq (\mu - \varepsilon)\,|S_{i+1}|$.*

Obtaining the Result of Theorem 2. We obtain this theorem as a direct application of Lemma 6. Recall that we will be working in the setting when $g = n - (t_S + t_R) \geq cn$ for some constant $c \in (0,1]$. Now we apply Lemma 6 to obtain the disjoint sets $S_1, \ldots, S_m$ for $m = n/\rho$ where $\rho = \omega(\log n)$. Next, we apply the protocol in Fig. 1 to each of the sets independently for the following choice of parameters: $n' = |S_i|$, $t'_S = (\frac{t_S}{n} + \varepsilon)\,|S_i|$, and $t'_R = (\frac{t_R}{n} + \varepsilon)\,|S_i|$. Note that new gap $g' = (\frac{g}{n} - 2\varepsilon)\,|S_i|$. The simulation error obtained for any OT produced will be bounded by $2^{-\Theta(g')} = \mathsf{negl}(n)$.

We observe that the approach of subsampling to obtain "disjoint subsets" while preserving min-entropy is unlikely to yield constant production rate extractors.

5 Inner Product Correlation

In this section we prove Theorem 4. Our protocol is provided in Fig. 3.

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 $\widetilde{\mathbf{H}}_\infty(Y_{[n]}|L) \geq m = n - t$. For security, we

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 $(u_0, \ldots, u_n) \in \mathcal{C}$ and a 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 $\alpha_{[n]} = x_{[n]} \oplus u_{[n]}$ and $\beta = \langle x_{[n]}, m_{[n]} \rangle \oplus a \oplus v_0$ to client B.
   (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 $\langle u_{[n]}, r_{[n]} \rangle = u_0 r_0$.

Fig. 3. Random oblivious function evaluation extractor from one inner product correlation over n-bits.

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 punctured." 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 Fig. 2. 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 an adversary $\mathcal{A}'$ which uses $\mathcal{A}$ to break the unpredictability experiment of Lemma 4 with identical advantage using a simulation similar to Fig. 2.

Note that as before this will be 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 predict $u_0 = \lambda G_0$ then the adversary $\mathcal{A}'$ can also predict λG_0 with identical advantage. By Lemma 4, the distribution $(U_0|\vartheta)$ is at most $2^{-(g/2+1)}$ far from the uniform distribution over $\{0,1\}$.

References

1. Alon, N., Roichman, Y.: Random cayley graphs and expanders. Random Struct. Algorithms **5**(2), 271–285 (1994). http://dx.doi.org/10.1002/rsa.3240050203
2. Beaver, D.: Perfect privacy for two-party protocols. In: Feigenbaum, J., Merritt, M. (eds.) Proceedings of DIMACS Workshop on Distributed Computing and Cryptography. vol. 2, pp. 65–77. American Mathematical Society (1989)

3. Ben-Or, M., Goldwasser, S., Wigderson, A.: Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In: STOC. pp. 1–10 (1988)
4. Bennett, C.H., Brassard, G., Crépeau, C., Maurer, U.M.: Generalized privacy amplification. IEEE Trans. Inf. Theor. **41**(6), 1915–1923 (1995). http://dx.doi.org/10.1109/18.476316
5. Bennett, C.H., Brassard, G., Robert, J.: Privacy amplification by public discussion. SIAM J. Comput. **17**(2), 210–229 (1988). http://dx.doi.org/10.1137/0217014
6. Bitansky, N., Canetti, R., Halevi, S.: Leakage-tolerant interactive protocols. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 266–284. Springer, Heidelberg (2012)
7. Boyle, E., Garg, S., Jain, A., Kalai, Y.T., Sahai, A.: Secure computation against adaptive auxiliary information. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 316–334. Springer, Heidelberg (2013)
8. Canetti, R., Lindell, Y., Ostrovsky, R., Sahai, A.: Universally composable two-party and multi-party secure computation. In: STOC. pp. 494–503 (2002)
9. Chari, S., Jutla, C.S., Rao, J.R., Rohatgi, P.: Towards sound approaches to counteract power-analysis attacks. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, p. 398. Springer, Heidelberg (1999)
10. Dodis, Y., Ostrovsky, R., Reyzin, L., Smith, A.: Fuzzy extractors: how to generate strong keys from biometrics and other noisy data. SIAM J. Comput. **38**(1), 97–139 (2008). http://dx.doi.org/10.1137/060651380
11. Dodis, Y., Smith, A.: Correcting errors without leaking partial information. In: STOC. pp. 654–663 (2005)
12. Duc, A., Dziembowski, S., Faust, S.: Unifying leakage models: from probing attacks to noisy leakage. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 423–440. Springer, Heidelberg (2014)
13. Duc, A., Faust, S., Standaert, F.-X.: Making masking security proofs concrete. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9056, pp. 401–429. Springer, Heidelberg (2015)
14. Dziembowski, S., Faust, S.: Leakage-resilient circuits without computational assumptions. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 230–247. Springer, Heidelberg (2012)
15. Dziembowski, S., Faust, S., Skorski, M.: Noisy leakage revisited. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9057, pp. 159–188. Springer, Heidelberg (2015)
16. Garcia, A., Stichtenoth, H.: On the asymptotic behaviour of some towers of function fields over finite fields. J. Number Theor. **61**(2), 248–273 (1996)
17. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game or a completeness theorem for protocols with honest majority. In: STOC. pp. 218–229 (1987)
18. Goldreich, O., Wigderson, A.: Tiny families of functions with random properties: a quality-size trade-off for hashing. Random Struct. Algorithms **11**(4), 315–343 (1997). http://dx.doi.org/10.1002/(SICI)1098-2418(199712)11:4h(315:AID-RSA3)3.0.CO;2-1
19. Goppa, V.D.: Codes on algebraic curves. Soviet Math. Dokl **24**(1), 170–172 (1981)
20. Gupta, D., Ishai, Y., Maji, H.K., Sahai, A.: Secure computation from leaky correlated randomness (2014). (full version: www.cs.ucla.edu/hmaji/papers/GuptaIsMaSa14.pdf)
21. Halevi, S., Lin, H.: After-the-fact leakage in public-key encryption. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 107–124. Springer, Heidelberg (2011)

22. Harnik, D., Ishai, Y., Kushilevitz, E., Nielsen, J.B.: OT-combiners via secure computation. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 393–411. Springer, Heidelberg (2008)
23. Harnik, D., Kilian, J., Naor, M., Reingold, O., Rosen, A.: On robust combiners for oblivious transfer and other primitives. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 96–113. Springer, Heidelberg (2005)
24. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Extracting correlations. In: FOCS, pp. 261–270 (2009)
25. Ishai, Y., Maji, H.K., Sahai, A., Wullschleger, J.: Single-use ot combiners with near-optimal resilience. In: ISIT, IEEE (2014)
26. Ishai, Y., Prabhakaran, M., Sahai, A.: Founding cryptography on oblivious transfer – efficiently. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 572–591. Springer, Heidelberg (2008)
27. Kilian, J.: Founding cryptography on oblivious transfer. In: STOC, pp. 20–31 (1988)
28. Kilian, J.: More general completeness theorems for secure two-party computation. In: STOC, pp. 316–324 (2000)
29. Kushilevitz, E.: Privacy and communication complexity. In: FOCS, pp. 416–421. IEEE (1989)
30. Maji, H.K., Prabhakaran, M., Rosulek, M.: Complexity of multi-party computation problems: the case of 2-party symmetric secure function evaluation. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 256–273. Springer, Heidelberg (2009)
31. Massey, J.L.: Some applications of coding theory in cryptography. In: Codes and Ciphers: Cryptography and Coding IV. pp. 33–47 (1995)
32. Meier, R., Przydatek, B.: On robust combiners for private information retrieval and other primitives. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 555–569. Springer, Heidelberg (2006)
33. Meier, R., Przydatek, B., Wullschleger, J.: Robuster Combiners for Oblivious Transfer. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 404–418. Springer, Heidelberg (2007)
34. Naor, J., Naor, M.: Small-bias probability spaces: efficient constructions and applications. In: STOC, pp. 213–223 (1990)
35. Nielsen, J.B., Nordholt, P.S., Orlandi, C., Burra, S.S.: A new approach to practical active-secure two-party computation. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 681–700. Springer, Heidelberg (2012)
36. Prouff, E., Rivain, M.: Masking against side-channel attacks: a formal security proof. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 142–159. Springer, Heidelberg (2013)
37. Przydatek, B., Wullschleger, J.: Error-tolerant combiners for oblivious primitives. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfsdóttir, A., Walukiewicz, I. (eds.) ICALP 2008, Part II. LNCS, vol. 5126, pp. 461–472. Springer, Heidelberg (2008)
38. Vadhan, S.P.: Constructing locally computable extractors and cryptosystems in the bounded-storage model. J. Cryptol. **17**(1), 43–77 (2004). http://dx.doi.org/10.1007/s00145-003-0237-x
39. Yao, A.C.C.: How to generate and exchange secrets (extended abstract). In: FOCS, pp. 162–167 (1986)