



Secure Computation Using Leaky Correlations (Asymptotically Optimal Constructions)

Alexander R. Block^{1(✉)}, Divya Gupta², Hemanta K. Maji¹,
and Hai H. Nguyen¹

¹ Department of Computer Science, Purdue University, West Lafayette, IN, USA

{block9,hmaji,nguye245}@purdue.edu

² Microsoft Research, Bangalore, India

divya.gupta@microsoft.com

Abstract. Most secure computation protocols can be effortlessly adapted to offload a significant fraction of their computationally and cryptographically expensive components to an offline phase so that the parties can run a fast online phase and perform their intended computation securely. During this offline phase, parties generate private shares of a sample generated from a particular joint distribution, referred to as the *correlation*. These shares, however, are susceptible to leakage attacks by adversarial parties, which can compromise the security of the secure computation protocol. The objective, therefore, is to preserve the security of the honest party despite the leakage performed by the adversary on her share.

Prior solutions, starting with n -bit leaky shares, either used 4 messages or enabled the secure computation of only sub-linear size circuits. Our work presents the first 2-message secure computation protocol for 2-party functionalities that have $\Theta(n)$ circuit-size despite $\Theta(n)$ -bits of leakage, a qualitatively optimal result. We compose a suitable 2-message secure computation protocol in parallel with our new 2-message *correlation extractor*. Correlation extractors, introduced by Ishai, Kushilevitz, Ostrovsky, and Sahai (FOCS-2009) as a natural generalization of privacy amplification and randomness extraction, recover “fresh” correlations from the leaky ones, which are subsequently used by other cryptographic protocols. We construct the first 2-message correlation extractor that produces $\Theta(n)$ -bit fresh correlations even after $\Theta(n)$ -bit leakage.

Our principal technical contribution, which is of potential independent interest, is the construction of a family of multiplication-friendly linear secret sharing schemes that is simultaneously a family of small-bias distributions. We construct this family by randomly “twisting then permuting” appropriate Algebraic Geometry codes over constant-size fields.

A. R. Block, H. K. Maji, and H. H. Nguyen—The research effort is supported in part by an NSF CRII Award CNS-1566499, an NSF SMALL Award CNS-1618822, and an REU CNS-1724673.

© International Association for Cryptologic Research 2018

A. Beimel and S. Dziembowski (Eds.): TCC 2018, LNCS 11240, pp. 36–65, 2018.

https://doi.org/10.1007/978-3-030-03810-6_2

1 Introduction

Secure multi-party computation (MPC) allows mutually distrusting parties to compute securely over their private data. Secure computation of most functionalities requires expensive public-key primitives such as oblivious transfer, even in the semi-honest setting.¹ We can effortlessly adjust most of these existing secure computation protocols so that they offload a significant fraction of their complex operations to an offline preprocessing phase. Subsequently, during an online phase, parties can implement extremely fast secure computation protocols. In fact, several specialized protocols optimize MPC for this online-offline paradigm [4, 6, 7, 15, 18, 28, 29, 31, 37].

For instance, in the two-party setting, we envision this offline phase as a secure implementation of a *trusted dealer* who generates private albeit correlated shares (r_A, r_B) for Alice and Bob, respectively, sampled from an appropriate joint distribution (R_A, R_B) , referred to as a *correlation*. This versatile framework allows the implementation of this trusted dealer using computational hardness assumptions, secure hardware, trusted hardware, or physical processes. Furthermore, this offline phase is independent of the final functionality to be computed, as well as the parties' private inputs.

A particularly useful correlation is the *random oblivious transfer correlation*, represented by ROT. One sample of this correlation generates three random bits x_0, x_1, b and provides private shares $r_A = (x_0, x_1)$ to Alice, and $r_B = (b, x_b)$ to Bob. Note that Alice does not know the choice bit b , and Bob does not know the other bit x_{1-b} . Let \mathcal{F} be the class of functionalities that admit 2-message secure computation protocols in the ROT-hybrid [10, 26]. Note that \mathcal{F} includes the powerful class of functions that have a decomposable randomized encoding [3, 5, 25]. Alice and Bob can compute the required ROTs in the offline phase. Then, they can compute *any* functionality from this class using 2-messages, a protocol exhibiting optimal message complexity² and (essentially) optimal efficiency in the usage of cryptographic resources.

However, the private share of the honest party is susceptible to leakage attacks by an adversary, both during the generation of the shares and the duration of storing the shares. We emphasize that the leakage need not necessarily reveal individual bits of the honest party's share. The leakage can be on the entire share and encode crucial global information that can potentially jeopardize the security of the secure computation protocol. This concern naturally leads to the following fundamental question.

“Can we preserve the security and efficiency of the secure computation during the online phase despite the adversarial leakage on the honest party's shares?”

Using the class \mathcal{F} of functionalities (defined above) as a yardstick, let us determine the primary hurdle towards a positive resolution of this question. In

¹ A semi-honest adversary follows the prescribed protocol but is curious to find additional information.

² Message complexity refers to the number of messages exchanged between Alice and Bob.

the sequel, $\mathcal{F}_{m/2} \subset \mathcal{F}$ is the set of all two-party functionalities that have a 2-message protocol in $\text{ROT}^{m/2}$ -hybrid, i.e., parties start with $m/2$ independent samples³ from the ROT correlation. In the leaky correlation setting where an adversary has already leaked global information from the private share of the honest party, our objective is to design an (asymptotically) optimal secure computation protocol for the functionalities in $\mathcal{F}_{m/2}$. That is, starting with leaky correlations (of size n), we want to compute any $F \in \mathcal{F}_{m/2}$ such that $m = \Theta(n)$ via a 2-message protocol despite $t = \Theta(n)$ bits of leakage. We note that this task is equivalent to the task of constructing a secure computation protocol for the particular functionality $\text{ROT}^{m/2}$ that also belongs to $\mathcal{F}_{m/2}$. This observation follows from the parallel composition of the secure protocol implementing the functionality $\text{ROT}^{m/2}$ from leaky correlations with the 2-message protocol for F in the $\text{ROT}^{m/2}$ -hybrid. To summarize, our overall objective of designing optimal secure computation protocols from leaky ROT correlations reduces to the following equivalent goal.

“Construct a 2-message protocol to compute $\text{ROT}^{m/2}$ securely, where $m = \Theta(n)$, from the leaky $\text{ROT}^{n/2}$ correlation in spite of $t = \Theta(n)$ bits of leakage.”

Note that in the $\text{ROT}^{n/2}$ -hybrid, both parties have private share of size n bits. The above problem is identical to *correlation extractors* introduced in the seminal work of Ishai, Kushilevitz, Ostrovsky, and Sahai [26].

Correlation Extractors. Ishai et al. [26] introduced the notion of correlation extractors as an interactive protocol that takes a leaky correlation as input and outputs a new correlation that is secure. Prior correlation extractors either used four messages [26] or had a sub-linear production [9, 22], i.e., $m = o(n)$. We construct the first 2-message correlation extractor that has a linear production and leakage resilience, that is, $m = \Theta(n)$ and $t = \Theta(n)$. Note that even computationally secure protocols can use the output of the correlation extractor in the online phase. Section 1.1 formally defines correlation extractors, and we present our main contributions in Sect. 1.2.

1.1 Correlation Extractors and Security Model

We consider the standard model of Ishai et al. [26], which is also used by the subsequent works, for 2-party semi-honest secure computation in the preprocessing model. In the preprocessing step, a trusted dealer draws a sample of shares (r_A, r_B) from the joint distribution of correlated private randomness (R_A, R_B) . The dealer provides the secret share r_A to Alice and r_B to Bob. Moreover, the adversarial party can perform an arbitrary t -bits of leakage on the secret share of the honest party at the end of the preprocessing step. We represent this leaky correlation hybrid⁴ as $(R_A, R_B)^{[t]}$.

³ Each sample of ROT gives two bits to each party; (x_0, x_1) to the first party and (b, x_b) to the second party. Therefore each party receives m -bit shares.

⁴ That is, the functionality samples secret shares (r_A, r_B) according to the correlation (R_A, R_B) . The adversarial party sends a t -bit leakage function \mathcal{L} to the functionality

Definition 1 (Correlation Extractor). *Let (R_A, R_B) be a correlated private randomness such that the secret share of each party is n -bits. An (n, m, t, ε) -correlation extractor for (R_A, R_B) is a two-party interactive protocol in the $(R_A, R_B)^{[t]}$ -hybrid that securely implements the $\text{ROT}^{m/2}$ functionality against information-theoretic semi-honest adversaries with ε simulation error.*

Note that the size of the secret shares output by the correlation extractor is m -bits. We emphasize that no leakage occurs during the correlation extractor execution. The t -bit leakage cumulatively accounts for all the leakage before the beginning of the online phase. We note that, throughout this work, we shall always normalize the total length of the input shares of each party to n -bits.

1.2 Our Contribution

Recall that $\mathcal{F}_{m/2} \subset \mathcal{F}$ is the set of all two-party functionalities that have a 2-message protocol in the $\text{ROT}^{m/2}$ -hybrid. We prove the following results.

Theorem 1 (Asymptotically Optimal Secure Computation from Leaky Correlations). *There exists a correlation (R_A, R_B) that produces n -bit secret shares such that for all $F \in \mathcal{F}_{m/2}$ there exists a 2-message secure computation protocol for F in the leaky $(R_A, R_B)^{[t]}$ -hybrid, where $m = \Theta(n)$ and $t = \Theta(n)$, with exponentially low simulation error.*

The crucial ingredient of Theorem 1 is our new 2-message (n, m, t, ε) -correlation extractor for $\text{ROT}^{n/2}$. We compose the 2-message secure computation protocol for functionalities in $\mathcal{F}_{m/2}$ in the $\text{ROT}^{m/2}$ -hybrid with our correlation extractor. Our work presents the first 2-message correlation extractor that has a linear production and a linear leakage resilience (along with exponentially low insecurity).

Theorem 2 (Asymptotically Optimal Correlation Extractor for ROT).

There exists a 2-message (n, m, t, ε) -correlation extractor for $\text{ROT}^{n/2}$ such that $m = \Theta(n)$, $t = \Theta(n)$, and $\varepsilon = \exp(-\Theta(n))$.

The technical heart of the correlation extractor of Theorem 2 is another correlation extractor (see Theorem 3) for a generalization of the ROT correlation. For any finite field \mathbb{F} , the *random oblivious linear-function evaluation* correlation over \mathbb{F} [36, 42], represented by $\text{ROLE}(\mathbb{F})$, samples random $a, b, x \in \mathbb{F}$ and defines $r_A = (a, b)$ and $r_B = (x, z)$, where $z = ax + b$. Note that, for $\mathbb{F} = \text{GF}[2]$, we have $(x_0 + x_1)b + x_0 = x_b$; therefore, the $\text{ROLE}(\text{GF}[2])$ correlation is identical to the ROT correlation. One share of the $\text{ROLE}(\mathbb{F})$ correlation has secret share size $2 \lg |\mathbb{F}|$. In particular, the correlation $\text{ROLE}(\mathbb{F})^{n/2 \lg |\mathbb{F}|}$ provides each party with $n/2 \lg |\mathbb{F}|$ independent samples from the $\text{ROLE}(\mathbb{F})$ correlation and the secret share size of each party is n -bits for suitable constant sized field \mathbb{F} .

and receives the leakage $\mathcal{L}(r_A, r_B)$ from the functionality. The functionality sends r_A to Alice and r_B to Bob. Note that the adversary does not need to know its secret share to construct the leakage function because the leakage function gets the secret shares of both parties as input.

Theorem 3 (Asymptotically Optimal Correlation Extractor for $\text{ROLE}(\mathbb{F})$). *There exists a 2-message (n, m, t, ε) -correlation extractor for $\text{ROLE}(\mathbb{F})^{n/2 \lg |\mathbb{F}|}$ such that $m = \Theta(n)$, $t = \Theta(n)$, and $\varepsilon = \exp(-\Theta(n))$.*

In Fig. 4, we present our correlation extractor that outputs fresh samples from the same $\text{ROLE}(\mathbb{F})$ correlation. Finally, our construction obtains multiple ROT samples from *each* output $\text{ROLE}(\mathbb{F})$ sample using the OT embedding technique of [9]. Figure 1 positions our contribution vis-à-vis the previous state-of-the-art. In particular, Fig. 1 highlights the fact that our result simultaneously achieves the best qualitative parameters. Our results are also quantitatively better than the previous works and we discuss the concrete performance numbers we obtain for Theorem 3 and Theorem 2 below. For more detailed numerical comparison with prior works [9, 22, 26], refer to Sect. 5.

	Correlation Description	Message Complexity	Number of OTs Produced ($m/2$)	Number of Leakage bits (t)	Simulation Error (ε)
IKOS [26]	$\text{ROT}^{n/2}$	4	$\Theta(n)$	$\Theta(n)$	$2^{-\Theta(n)}$
GIMS [22]	$\text{ROT}^{n/2}$	2	$n / \text{poly } \lg n$	$(1/4 - g)n$	$2^{-gn/m}$
	$\text{IP}(\mathbb{K}^{n/ \mathbb{g} \mathbb{K} })$	2	1	$(1/2 - g)n$	2^{-gn}
BMN [9]	$\text{IP}(\mathbb{K}^{n/ \mathbb{g} \mathbb{K} })$	2	$n^{1-o(1)}$	$(1/2 - g)n$	2^{-gn}
Our Result	$\text{ROT}^{n/2}$	2	$\Theta(n)$	$\Theta(n)$	$2^{-\Theta(n)}$
	$\text{ROLE}(\mathbb{F})^{n/2 \lg \mathbb{F} }$	2	$\Theta(n)$	$\Theta(n)$	$2^{-\Theta(n)}$

Fig. 1. A qualitative summary of our correlation extractor constructions and a comparison to prior relevant works. Here \mathbb{K} is a finite field and \mathbb{F} is a finite field of constant size. The $\text{IP}(\mathbb{K}^s)$ is a correlation that samples random $r_A = (u_1, \dots, u_s) \in \mathbb{K}^s$ and $r_B = (v_1, \dots, v_s) \in \mathbb{K}^s$ such that $u_1 v_1 + \dots + u_s v_s = 0$. All correlations are normalized so that each party gets an n -bit secret share. The parameter g is the gap to maximal leakage resilience such that. $g > 0$.

Performance of Correlation Extractors for $\text{ROLE}(\mathbb{F})$ (Theorem 3). Our correlation extractor for $\text{ROLE}(\mathbb{F})$ relies on the existence of suitable Algebraic Geometry (AG) codes⁵ over finite field \mathbb{F} , such that $|\mathbb{F}|$ is an even power of a prime and $|\mathbb{F}| \geq 49$. We shall use \mathbb{F} that is a finite field with characteristic 2.

As the size of the field \mathbb{F} increases, the “quality” of the Algebraic Geometry codes get better. However, the efficiency of the BMN OT embedding protocol [9] used to obtain the output ROT in our construction decreases with increasing $|\mathbb{F}|$. For example, with $\mathbb{F} = \mathbb{GF}[2^{14}]$ we achieve the highest production rate $m/n = 16.32\%$ if the fractional leakage rate is $t/n = 1\%$. For leakage rate $t/n = 10\%$, we achieve production rate $m/n = 10\%$. Figure 7 (Sect. 5) and Fig. 9 (Sect. 6) summarize these tradeoffs for various choices of the finite field \mathbb{F} .

⁵ Once the parameters of the AG code are fixed, it is a one-time cost to construct its generator matrix.

Performance of Correlation Extractors for ROT (Theorem 2). We know extremely efficient algorithms that use multiplications over $\mathbb{GF}[2]$ to emulate multiplications over any $\mathbb{GF}[2^s]$ [12, 14]. For example, we can use 15 multiplications over $\mathbb{GF}[2]$ to emulate one multiplication over $\mathbb{GF}[2^6]$. Therefore, we can use 15 samples of $\text{ROLE}(\mathbb{GF}[2])$ to perform one $\text{ROLE}(\mathbb{GF}[2^6])$ with perfect semi-honest security. Note that, by applying this protocol, the share sizes reduce by a factor of 6/15. In general, using this technique, we can convert the leaky $\text{ROLE}(\mathbb{GF}[2])$ (equivalently, ROT) correlation, into a leaky $\text{ROLE}(\mathbb{F})$ correlation, where \mathbb{F} is a finite field of characteristic 2, by incurring a slight multiplicative loss in the share size. Now, we can apply the correlation extractor for $\text{ROLE}(\mathbb{F})$ discussed above. By optimizing the choice of the field \mathbb{F} (in our case $\mathbb{F} = \mathbb{GF}[2^{10}]$), we can construct a 2-message correlation extractor for ROT with fractional leakage rate $t/n = 1\%$ and achieve production rate of $m/n = 4.20\%$ (see Fig. 8, Sect. 5). This is several orders of magnitude better than the production and resilience of the IKOS correlation extractor and uses less number of messages.⁶

High Leakage Resilience Setting. Ishai et al. [27] showed that $t < n/4$ is necessary to extract even one new sample of ROT from the leaky $\text{ROLE}(\mathbb{F})^{n/2 \lg \mathbb{F}}$ correlation. Our construction, when instantiated with a suitably large constant-size field \mathbb{F} , demonstrates that if $t \leq (1/4 - g)n$ then we can extract $\Theta(n)$ new samples of the ROT correlation. The prior construction of [22] only achieves a sub-linear production by using sub-sampling techniques.

Theorem 4 (Near Optimal Resilience with Linear Production). *For every $g \in (0, 1/4]$, there exists a finite field \mathbb{F} with characteristic 2 and a 2-message (n, m, t, ε) -correlation extractor for $(R_A, R_B) = \text{ROLE}(\mathbb{F})^{n/2 \lg |\mathbb{F}|}$, where $t = (1/4 - g)n$, $m = \Theta(n)$, and $\varepsilon = \exp(-\Theta(n))$.*

The production $m = \Theta(n)$ depends on the constant g , the gap to optimal fractional resilience. We prove Theorem 4 in the full version of our work [8]. Section 5 shows that we can achieve linear production even for $t = 0.22n$ bits of leakage using $\mathbb{F} = \mathbb{GF}[2^{10}]$.

Correlation Extractors for Arbitrary Correlations. Similar to the construction of IKOS, we can also construct a correlation extractor from *any* correlation and output samples of *any* correlation; albeit it is not round optimal anymore. However, our construction achieves overall better production and leakage resilience than IKOS because our correlation extractor for ROT has higher production and resilience. Figure 2 outlines a comparison of these two correlation extractor construction for the general case.

⁶ Even optimistic estimates of the parameters m/n and t/n for the IKOS construction are in the order of 10^{-6} .

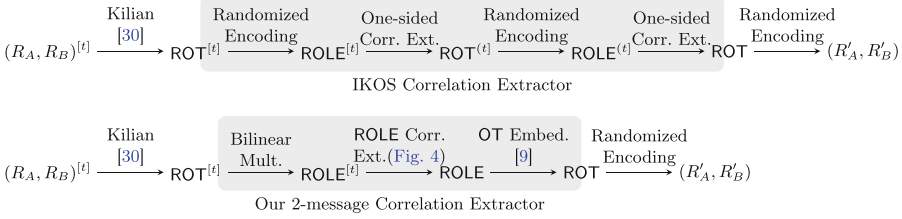


Fig. 2. General correlation extractors that extract arbitrary correlations from arbitrary correlations. Above is the expanded IKOS [26] correlation extractor and below is ours. Our main contribution is shown in highlighted part. For brevity, it is implicit that there are *multiple samples* of the correlations. The **ROLE** correlations are over suitable constant size fields. The superscript “ $[t]$ ” represents that the correlation is secure against adversarial leakage of only one a priori fixed party.

1.3 Other Prior Relevant Works

Figure 1 already provides the summary of the current state-of-the-art in correlation extractors. In this section, we summarize works related to combiners: extractors where the adversary is restricted to leaking individual bits of the honest party’s secret share. The study of OT combiners was initiated by Harnik et al. [24]. Since then, there has been work on several variants and extensions of OT combiners [23, 28, 33, 34, 39]. Recently, Ishai et al. [27] constructed OT combiners with nearly optimal leakage resilience. Among these works, the most relevant to our paper are the ones by Meier, Przydatek, and Wullschleger [34] and Przydatek, and Wullschleger [39]. They use Reed-Solomon codes to construct two-message error-tolerant⁷ combiners that produce fresh ROLES over large fields⁸ from ROLES over the same field. Using multiplication friendly secret sharing schemes based on Algebraic Geometry Codes introduced by Chen and Cramer [13], a similar construction works with ROLES over fields with appropriate constant size. We emphasize that this construction is *insecure* if an adversary can perform even 1-bit global leakage on the whole secret of the other party. In our construction, we crucially rely on a family of linear codes instead of a particular choice of the linear code to circumvent this bottleneck. Section 1.4 provides the principal technical ideas underlying our correlation extractor construction.

In the malicious setting, the feasibility result on malicious-secure combiners for ROT is reported in [28]. Recently, Cascudo et al. construct a malicious-secure combiner with high resilience, but $m = 1$ [11]. The case of malicious-secure correlation extractors remains entirely unexplored.

⁷ A sample (r_A, r_B) is an *erroneous sample* if it is not in the support of the distribution (R_A, R_B) , i.e., it is an incorrect sample. An error-tolerant combiner is a combiner that is secure even if a few of the input samples are erroneous.

⁸ The size of the fields increases with n , the size of the secret shares produced by the preprocessing step.

1.4 Technical Overview

At the heart of our correlation extractor constructions is a 2-message $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor, where we start with leaky $(R_A, R_B)^{[t]} = \left(\text{ROLE}(\mathbb{F})^{n/2 \lg |\mathbb{F}|}\right)^{[t]}$ and produce fresh secure sample of $\text{ROLE}(\mathbb{F})^{m/2 \lg |\mathbb{F}|}$. The field \mathbb{F} is a constant-size field with characteristic 2, say $\mathbb{F} = \mathbb{GF}[2^6]$, and each party gets n -bit shares. Below, we discuss some of the technical ideas underlying this construction.

This correlation extractor relies on the existence of a family of linear codes over \mathbb{F} with suitable properties that we define below. For this discussion, let us assume that $s \in \mathbb{N}$ is the block-length of the codes. Let \mathcal{J} be an index set, and we denote the family of linear codes with block-length s as follows: $\mathcal{C} = \{C_j : j \in \mathcal{J}\}$. This family of code \mathcal{C} needs to have the following properties.

1. **Multiplication Friendly Good Codes.** Each code $C_j \subseteq \mathbb{F}^s$ in the family \mathcal{C} is a *good code*, i.e., its rate and distance is $\Theta(s)$. Further, the Schur-product⁹ of the codes, i.e., $C_j * C_j$, is a linear code with distance $\Theta(s)$. Such codes can be used to perform the multiplication of two secrets by multiplying their respective secret shares in secure computation protocols, hence the name.
2. **Small Bias Family.** Intuitively, a small bias family defines a pseudorandom distribution for linear tests. Let $S = (S_1, \dots, S_s) \in \mathbb{F}^s$ and its corresponding linear test be defined as $L_S(x_1, \dots, x_s) := S_1 x_1 + \dots + S_s x_s$. Consider the distribution D of $L_S(c)$ for a random $j \in \mathcal{J}$ and a randomly sampled codeword $c \in C_j$. If \mathcal{C} is a family of ρ -biased distributions, then the distribution D has statistical distance at most ρ from the output of $L_S(u)$ for random element $u \in \mathbb{F}^s$. For brevity, we say that the family \mathcal{C} “ ρ -fools the linear test L_S .” The concept of small bias distributions was introduced in [1,35] and has found diverse applications, for example, [2,17,20,35].

An interesting property of any linear code $C \subseteq \mathbb{F}^s$ is the following. A random codeword $c \in C$ can 0-fool every linear test L_S such that S is *not* a codeword in the dual of C . However, if S is a codeword in the dual of the code C , then the linear test L_S is clearly not fooled.

So, a randomly chosen codeword from one fixed linear code cannot fool *all* linear tests. However, when we consider an appropriate family of linear codes, then a randomly chosen codeword from a randomly chosen code in this family can fool *every* linear test.

We construct such a family of codes over small finite fields \mathbb{F} that can be of potential independent interest. Our starting point is an explicit Algebraic Geometry code $C \subseteq \mathbb{F}^s$ that is multiplication friendly [19,21]. Given one such code C , we randomly “twist then permute” the code to define the family \mathcal{C} . We emphasize that the production of our correlation extractor relies on the bias

⁹ Consider a linear code $C \subseteq \mathbb{F}^s$. Let $c = (c_1, \dots, c_s)$ and $c' = (c'_1, \dots, c'_s)$ be two codewords in the code C . We define $c * c' = (c_1 c'_1, \dots, c_s c'_s) \in \mathbb{F}^s$. The Schur-product $C * C$ is defined to be the linear span of all $c * c'$ such that $c, c' \in C$.

being small. So, it is crucial to construct a family with extremely small bias. Next, we describe our “twist then permute” operation.

Twist then Permute.¹⁰ Suppose $C \subseteq \mathbb{F}^s$ is a linear code. Pick any $\lambda = (\lambda_1, \dots, \lambda_s) \in (\mathbb{F}^*)^s$, i.e., for all $i \in [s]$, $\lambda_i \neq 0$. A λ -twist of the code C is defined as the following linear code

$$C_\lambda := \{(\lambda_1 c_1, \dots, \lambda_s c_s) : (c_1, \dots, c_s) \in C\}.$$

Let $\pi: \{1, \dots, s\} \rightarrow \{1, \dots, s\}$ be a permutation. The π -permutation of the λ -twist of C is defined as the following linear code

$$C_{\pi, \lambda} := \{(\lambda_{\pi(1)} c_{\pi(1)}, \dots, \lambda_{\pi(s)} c_{\pi(s)}) : (c_1, \dots, c_s) \in C\}.$$

Define \mathcal{J} as the set of all (π, λ) such that $\lambda \in (\mathbb{F}^*)^s$ and π is a permutation of the set $\{1, \dots, s\}$. Note that if C is multiplication friendly good code, then the code $C_{\pi, \lambda}$ continues to be multiplication friendly good code. A key observation towards demonstrating that \mathcal{C} is a family of small bias distributions is that the following two distributions are identical (see Claim 2).

1. Fix $S \in \mathbb{F}^s$. The output distribution of the linear test L_S on a random codeword $c \in C_j$, for a random index $j \in \mathcal{J}$.
2. Let $T \in \mathbb{F}^s$ be a random element of the same weight¹¹ as S . The output distribution of the linear test L_T on a random codeword $c \in C$.

Based on this observation, we can calculate the bias of the family of our codes. Note that there are a total of $\binom{s}{w}(q-1)^w$ elements in \mathbb{F}^s that have weight w . Let A_w denote the number of codewords in the dual of C that have weight w . Our family of codes \mathcal{C} fools the linear test L_S with $\rho = A_w \cdot \binom{s}{w}^{-1} (q-1)^{-w}$, where w is the weight of $S \in \mathbb{F}^s$.

We obtain precise asymptotic bounds on the weight enumerator A_w of the dual of the code C to estimate the bias ρ , for $w \in \{0, 1, \dots, s\}$. This precise bound translates into higher production m , higher resilience t , and exponentially low simulation error ε of our correlation extractor. We remark that for our construction if C has a small dual-distance, then the bias cannot be small.

Remark. The performance of the code C supersedes the elementary Gilbert-Varshamov bound. These Algebraic Geometry codes are one of the few codes in mathematics and computer science where explicit constructions have significantly better quality than elementary randomized constructions. So, elementary randomization techniques are unlikely to produce any (qualitatively) better

¹⁰ In the literature there are multiple definitions for the *equivalence* of two linear codes.

In particular, one such notion (cf., [38]), states that two codes are equivalent to each other if one can be twisted-and-permuted into the other code. For clarity, we have chosen to explicitly define the “twist then permute” operation.

¹¹ The weight of $S \in \mathbb{F}^s$ is defined as the number of non-zero elements in S .

parameters for this approach, given that the estimations of the weight enumerator in this work are asymptotically optimal. Therefore, finding *better randomized techniques* to construct the family of multiplication friendly good codes that is also a family of small-bias distributions is the research direction that has the potential to reduce the bias. This reduction in the bias can further improve the production and leakage resilience of our correlation extractors.

2 Preliminaries

We denote random variables by capital letters, for example X , and the values taken by small letters, for example $X = x$. For a positive integer n , we write $[n]$ and $[-n]$ to denote the sets $\{1, \dots, n\}$ and $\{-n, \dots, -1\}$, respectively. Let \mathcal{S}_n be the set of all permutations $\pi : [n] \rightarrow [n]$. We consider the field $\mathbb{F} = \mathbb{GF}[q]$, where $q = p^a$, for a positive integer a and prime p . For any $c = (c_1, \dots, c_\eta) \in \mathbb{F}^\eta$, define the function $\text{wt}(c)$ as the cardinality of the set $\{i : c_i \neq 0\}$. For any two $x, y \in \mathbb{F}^\eta$, we let $x * y$ represent the point-wise product of x and y . That is, $x * y = (x_1 y_1, x_2 y_2, \dots, x_\eta y_\eta) \in \mathbb{F}^\eta$. For a set Y , U_Y denotes the uniform distribution over the set Y , and $y \stackrel{\$}{\leftarrow} Y$ denotes sampling y according to U_Y . For any vector $x \in \mathbb{F}^\eta$ and a permutation $\pi \in \mathcal{S}_\eta$, we define $\pi(x) := (x_{\pi(1)}, \dots, x_{\pi(\eta)})$.

2.1 Correlation Extractors

We denote the functionality of 2-choose-1 bit Oblivious Transfer as OT and Oblivious Linear-function Evaluation over a field \mathbb{F} as OLE(\mathbb{F}). Also, we denote the Random Oblivious Transfer Correlation as ROT and Random Oblivious Linear-function Evaluation Correlation over the field \mathbb{F} as ROLE(\mathbb{F}). When $\mathbb{F} = \mathbb{GF}[2]$, we denote ROLE(\mathbb{F}) by ROLE.

Let η be such that $2\eta \lg |\mathbb{F}| = n$. In this work, we consider the setting when Alice and Bob start with η samples of the ROLE(\mathbb{F}) correlation and the adversary performs t -bits of leakage. We give a secure protocol for extracting multiple secure OTs in this hybrid. Below we define such a correlation extractor formally using initial ROLE(\mathbb{F}) correlations.

Leakage Model. We define our leakage model for ROLE(\mathbb{F}) correlations as follows:

1. **η -ROLE correlation generation phase.** Alice gets $r_A = \{(a_i, b_i)\}_{i \in [\eta]} \in \mathbb{F}^{2\eta}$ and Bob gets $r_B = \{(x_i, z_i)\}_{i \in [\eta]} \in \mathbb{F}^{2\eta}$ such that for all $i \in [\eta]$, a_i, b_i, x_i is uniformly random and $z_i = a_i x_i + b_i$. Note that the size of secret share of each party is n bits.
2. **Corruption and leakage phase.** A semi-honest adversary corrupts either the sender and sends a leakage function $L : \mathbb{F}^\eta \rightarrow \{0, 1\}^t$ and gets back $L(x_{[\eta]})$. Or, it corrupts the receiver and sends a leakage function $L : \mathbb{F}^\eta \rightarrow \{0, 1\}^t$ and gets back $L(a_{[\eta]})$. Note that w.l.o.g. any leakage on the sender (resp., receiver) can be seen as a leakage on $a_{[\eta]}$ (resp., $x_{[\eta]}$). We again emphasize that this leakage need not be on individual bits of the shares, but on the entire share, and thus can encode crucial global information.

We denote by (R_A, R_B) the above correlated randomness and by $(R_A, R_B)^{[t]}$ its t -leaky version. Recall the definition for (n, m, t, ε) -correlation extractor (see Definition 1, Sect. 1.1). Below, we give the correctness and security requirements.

The *correctness* condition says that the receiver's output is correct in all $m/2$ instances of ROT. The *privacy* requirement says the following: Let $(s_0^{(i)}, s_1^{(i)})$ and $(c^{(i)}, z^{(i)})$ be the output shares of Alice and Bob, respectively, in the i^{th} ROT instance. Then a corrupt sender (resp., receiver) cannot distinguish between $\{c^{(i)}\}_{i \in [m/2]}$ (resp., $\{s_{1-c^{(i)}}^{(i)}\}_{i \in [m/2]}$) and $r \xleftarrow{\$} \{0, 1\}^{m/2}$ with advantage more than ε . The *leakage rate* is defined as t/n and the *production rate* is defined as m/n .

2.2 Fourier Analysis over Fields

We give some basic Fourier definitions and properties over finite fields, following the conventions of [40]. To begin discussion of Fourier analysis, let η be any positive integer and let \mathbb{F} be any finite field. We define the *inner product* of two complex-valued functions.

Definition 2 (Inner Product). *Let $f, g: \mathbb{F}^\eta \rightarrow \mathbb{C}$. We define the inner product of f and g as*

$$\langle f, g \rangle := \mathbb{E}_{x \xleftarrow{\$} \mathbb{F}^\eta} [f(x) \cdot \overline{g(x)}] = \frac{1}{|\mathbb{F}|^\eta} \sum_{x \in \mathbb{F}^\eta} f(x) \cdot \overline{g(x)},$$

where $\overline{g(x)}$ is the complex conjugate of $g(x)$.

Next, we define general *character functions* for both \mathbb{F} and \mathbb{F}^η .

Definition 3 (General Character Functions). *Let $\psi: \mathbb{F} \rightarrow \mathbb{C}^*$ be a group homomorphism from the additive group \mathbb{F} to the multiplicative group \mathbb{C}^* . Then we say that ψ is a character function of \mathbb{F} .*

Let $\chi: \mathbb{F}^\eta \times \mathbb{F}^\eta \rightarrow \mathbb{C}^$ be a bilinear, non-degenerate, and symmetric map defined as $\chi(x, y) = \psi(x \cdot y) = \psi(\sum_i x_i y_i)$. Then, for any $S \in \mathbb{F}^\eta$, the function $\chi(S, \cdot) := \chi_S(\cdot)$ is a character function of \mathbb{F}^η .*

Given χ , we have the *Fourier Transformation*.

Definition 4 (Fourier Transformation). *For any $S \in \mathbb{F}^\eta$, let $f: \mathbb{F}^\eta \rightarrow \mathbb{C}$ and χ_S be a character function. We define the map $\widehat{f}: \mathbb{F}^\eta \rightarrow \mathbb{C}$ as $\widehat{f}(S) := \langle f, \chi_S \rangle$. We say that $\widehat{f}(S)$ is a Fourier Coefficient of f at S and the linear map $f \mapsto \widehat{f}$ is the Fourier Transformation of f .*

Note that this transformation is an invertible linear map. The Fourier inversion formula is given by the following lemma.

Lemma 1 (Fourier Inversion). *For any function $f: \mathbb{F}^\eta \rightarrow \mathbb{C}$, we can write $f(x) = \sum_{S \in \mathbb{F}^\eta} \widehat{f}(S) \chi_S(x)$.*

2.3 Distributions and Min-Entropy

For a probability distribution X over a sample space U , entropy of $x \in X$ is defined as $H_X(x) = -\lg \Pr[X = x]$. The min-entropy of X , represented by $\mathbf{H}_\infty(X)$, is defined to be $\min_{x \in \text{Supp}(X)} H_X(x)$. The binary entropy function, denoted by $\mathbf{h}_2(x) = -x \lg x - (1-x) \lg(1-x)$ for every $x \in (0, 1)$.

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 $\sum_{x \in U} \Pr[X = x, Y = y]$. The conditional distribution $(X|y)$ represents the distribution over sample space U such that the probability of $x \in U$ is $\Pr[X = x|Y = y]$. The average min-entropy [16], represented by $\tilde{\mathbf{H}}_\infty(X|Y)$, is defined to be $-\lg \mathbb{E}_{y \sim Y}[2^{-\mathbf{H}_\infty(X|y)}]$.

Imported Lemma 1 ([16]). *If $\mathbf{H}_\infty(X) \geq k$ and L is an arbitrary ℓ -bit leakage on X , then $\tilde{\mathbf{H}}_\infty(X|L) \geq k - \ell$.*

Lemma 2 (Fourier Coefficients of a Min-Entropy Distribution). *Let $X: \mathbb{F}^\eta \rightarrow \mathbb{R}$ be a min-entropy source such that $\mathbf{H}_\infty(X) \geq k$. Then $\sum_S |\hat{X}(S)|^2 \leq |\mathbb{F}|^{-\eta} \cdot 2^{-k}$.*

2.4 Family of Small-Bias Distributions

Definition 5 (Bias of a Distribution). *Let X be a distribution over \mathbb{F}^η . Then the bias of X with respect to $S \in \mathbb{F}^\eta$ is defined as $\text{Bias}_S(X) := |\mathbb{F}|^\eta \cdot |\hat{X}(S)|$.*

Dodis and Smith [17] defined small-bias distribution families for distributions over $\{0, 1\}^\eta$. We generalize it naturally for distributions over \mathbb{F}^η .

Definition 6 (Small-bias distribution family). *A family of distributions $\mathcal{F} = \{F_1, F_2, \dots, F_k\}$ over sample space \mathbb{F}^η is called a ρ^2 -biased family if for every non-zero vector $S \in \mathbb{F}^\eta$ following holds:*

$$\mathbb{E}_{i \leftarrow [k]} \text{Bias}_S(F_i)^2 \leq \rho^2.$$

Following extraction lemma was proven in previous works over $\{0, 1\}^\eta$.

Imported Lemma 2 ([2, 17, 20, 35]). *Let $\mathcal{F} = \{F_1, \dots, F_\mu\}$ be ρ^2 -biased family of distributions over the sample space $\{0, 1\}^\eta$. Let (M, L) be a joint distribution such that the marginal distribution M is over $\{0, 1\}^\eta$ and $\tilde{\mathbf{H}}_\infty(M|L) \geq m$. Then, the following holds: Let J be a uniform distribution over $[\mu]$. Then,*

$$\text{SD}((F_J \oplus M, L, J), (U_{\{0,1\}^\eta}, L, J)) \leq \frac{\rho}{2} \left(\frac{2^\eta}{2^m} \right)^{1/2}.$$

A natural generalization of above lemma for distributions over \mathbb{F}^η gives the following.

Theorem 5 (Min-entropy extraction via masking with small-bias distributions). *Let $\mathcal{F} = \{F_1, \dots, F_\mu\}$ be a ρ^2 -biased family of distributions over the sample space \mathbb{F}^η for field \mathbb{F} of size q . Let (M, L) be a joint distribution such that the marginal distribution M is over \mathbb{F}^η and $\tilde{H}_\infty(M|L) \geq m$. Then, the following holds: Let J be a uniform distribution over $[\mu]$. Then,*

$$\text{SD}((F_J \oplus M, L, J), (U_{\mathbb{F}^\eta}, L, J)) \leq \frac{\rho}{2} \left(\frac{|\mathbb{F}|^\eta}{2^m} \right)^{1/2}.$$

We provide the proof of this result in the full version of our work [8].

2.5 Distribution over Linear Codes

Let $C = [\eta, \kappa, d, d^{(2)}]_{\mathbb{F}}$ be a linear code over \mathbb{F} with generator matrix $G \in \mathbb{F}^{\kappa \times \eta}$. We also use C to denote the uniform distribution over codewords generated by G . For any $\pi \in \mathcal{S}_\eta$, define $G_\pi = \pi(G)$ as the generator matrix obtained by permuting the columns of G under π .

The **dual code** of C , represented by C^\perp , is the set of all codewords that are orthogonal to every codeword in C . That is, for any $c^\perp \in C^\perp$, it holds that $\langle c, c^\perp \rangle = 0$ for all $c \in C$. Let $H \in \mathbb{F}^{(\eta-\kappa) \times \eta}$ be a generator matrix of C^\perp . The distance of C^\perp is d^\perp .

The **Schur product code** of C , represented by $C^{(2)}$, is the span of all codewords obtained as a Schur product of codewords in C . That is, $C^{(2)} = C * C := \langle c * c' : c, c' \in C \rangle \subseteq \mathbb{F}^\eta$, where $c * c'$ denotes the coordinate-wise product of c and c' . The distance of $C^{(2)}$ is $d^{(2)}$.

3 Family of Small-Bias Distributions with Erasure Recovery

In this section, we give our construction of the family of small-bias distributions $\{C_j\}_{j \in \mathcal{J}}$ such that each C_j is a linear code and $C_j * C_j$ supports erasure recovery. Recall that $C_j * C_j$ is the linear span of all $c * c'$ such that $c, c' \in C_j$. We formally define the requirements for our family of distributions in Property 1.

Property 1. A family of linear code distributions $\mathcal{C} = \{C_j : j \in \mathcal{J}\}$ over \mathbb{F}^{η^*} satisfy this property with parameters δ and γ if the following conditions hold.

1. **$2^{-\delta}$ -bias family of distributions.** For any $0^{\eta^*} \neq S \in \mathbb{F}^{\eta^*}$, $\mathbb{E} [\text{Bias}_S(C_j)^2] \leq 2^{-\delta}$, where the expectation is taken over $j \xleftarrow{\$} \mathcal{J}$.
2. **γ -erasure recovery in Schur Product.** For all $j \in \mathcal{J}$, the Schur product code of C_j , that is $C_j * C_j = C_j^{(2)}$, supports the erasure recovery of the first γ coordinates. Moreover, the first γ coordinates of C_j and $C_j^{(2)}$ are linearly independent of each other.

3.1 Our Construction

Figure 3 presents our construction of a family of linear codes which satisfies Property 1 and Theorem 6 gives the parameters for our construction.

Family of small-bias distributions with erasure recovery in the product distribution:

Fix a linear code $C = [\eta^*, \kappa, d, d^\perp, d^{(2)}]_{\mathbb{F}}$ with generator matrix $G \in \mathbb{F}^{\kappa \times \eta^*}$, where $|\mathbb{F}| = q$ and $\kappa \geq d^{(2)}$, where $d^{(2)}$ is the distance of $C * C$. Let γ be a fixed natural number (to be determined later during parameter setting) such that $C * C$ supports γ -erasure recovery. We construct the family of small-bias distributions $\{C_{\pi, \lambda}: \pi \in \mathcal{S}_{\eta^*}, \lambda \in (\mathbb{F}^*)^{\eta^*}\}$ over \mathbb{F}^{η^*} as follows.

1. Let $\lambda \in (\mathbb{F}^*)^{\eta^*}$. Define $G_\lambda = [\lambda_1 G_1, \dots, \lambda_{\eta^*} G_{\eta^*}] \in \mathbb{F}^{\kappa \times \eta^*}$, where G_i is the i^{th} column of G and $\lambda_i G_i$ is the multiplication of G_i by λ_i .
2. Let $\pi \in \mathcal{S}_{\eta^*}$. Define $G_{\pi, \lambda} = \pi(G_\lambda) \in \mathbb{F}^{\kappa \times \eta^*}$, where $\pi(G_\lambda)$ is the permutation of the columns of G_λ according to permutation π . Then $C_{\pi, \lambda}$ is the uniform distribution over the linear code generated by $G_{\pi, \lambda}$.

(Enc, Dec) for $C_{\pi, \lambda}$: Let $(\text{Enc}_C, \text{Dec}_C)$ be the encoder and decoder for the code C .

- **Enc**(m): Compute $c = (c_1, \dots, c_{\eta^*}) = \text{Enc}_C(m)$. Compute $c * \lambda = (\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*})$. Output $\pi(c * \lambda)$.
- **Dec**(x): Compute $c' = (c'_1, \dots, c'_{\eta^*}) = \pi^{-1}(x)$. Compute $c' * \lambda' = (\lambda_1^{-1} c'_1, \dots, \lambda_{\eta^*}^{-1} c'_{\eta^*})$. Output $\text{Dec}_C(c' * \lambda')$.

(Enc, Dec) for $(C_{\pi, \lambda} * C_{\pi, \lambda})$: Let $(\text{Enc}_{C^{(2)}}, \text{Dec}_{C^{(2)}})$ be the Encoder and Decoder for the linear code $C^{(2)} = C * C$.

- **Enc**(m): Compute $c = (c_1, \dots, c_{\eta^*}) = \text{Enc}_{C^{(2)}}(m)$. Compute $c * \lambda * \lambda = (\lambda_1^2 c_1, \dots, \lambda_{\eta^*}^2 c_{\eta^*})$. Output $\pi(c * \lambda * \lambda)$.
- **Dec**(x): Compute $c' = (c'_1, \dots, c'_{\eta^*}) = \pi^{-1}(x)$. Compute $c' * \lambda' * \lambda' = (\lambda_1^{-2} c'_1, \dots, \lambda_{\eta^*}^{-2} c'_{\eta^*})$. Output $\text{Dec}_{C^{(2)}}(c' * \lambda' * \lambda')$.

Fig. 3. Our construction of a family of small bias linear code distributions.

At a high level, the linear code C is a suitable algebraic geometric code over constant size field \mathbb{F} of block length $\eta^* = \gamma + \eta$. The parameters of the code C are chosen such that C is a $2^{-\delta}$ -biased family of distributions under our “twist-then-permute” operation, and $C * C$ supports erasure recovery of any γ coordinates. The precise calculation of the parameters of the code C can be found in the full version of our work [8]. Our family of linear codes satisfies the following theorem.

Theorem 6. *The family of linear code distributions $\{C_{\pi, \lambda}: \pi \in \mathcal{S}_{\eta^*}, \lambda \in (\mathbb{F}^*)^{\eta^*}\}$ over \mathbb{F}^{η^*} given in Fig. 3 satisfies Property 1 for any $\gamma < d^{(2)}$, where $d^{(2)}$ is the distance of the Schur product code of C , and $\delta = [d^\perp + \eta^*/(\sqrt{q}-1) - 1] \cdot [\lg(q-1) - \mathbf{h}_2(1/(q+1))] - (\eta^*/(\sqrt{q}-1)) \lg q$, where \mathbf{h}_2 denotes the binary entropy function.*

Proof. We first prove erasure recovery followed by the small-bias property.

γ -erasure Recovery in Schur Product code. First we note that permuting or re-ordering the columns of a generator matrix does not change its distance, distance of the Schur product, or its capability of erasure recovery (as long as we know the mapping of new columns vis-à-vis old columns). Let $\mathcal{I}_\gamma = \{i_1, \dots, i_\gamma\}$ be the indices of the erased coordinates of codeword in $C_{\pi,\lambda}^{(2)}$. Hence to show erasure recovery of the coordinates \mathcal{I}_γ of a codeword of $C_{\pi,\lambda}^{(2)}$, it suffices to show erasure recovery of the γ erased coordinates $\mathcal{J}_\gamma = \{j_1, \dots, j_\gamma\}$ of a codeword of $C_\lambda^{(2)}$, where C_λ is the uniform codespace generated by G_λ , and $\pi(j_k) = i_k, \forall k \in [\gamma]$.

Note that since $\gamma < d^{(2)}$, the code $C^{(2)}$ supports erasure recovery of *any* γ coordinates. Thus it suffices to show that this implies that $C_\lambda^{(2)}$ also supports the erasure recovery of *any* γ coordinates. Note that since $\lambda \in (\mathbb{F}^*)^{\eta^*}$, multiplication of the columns of G according to λ does not change its distance or distance of the Schur product. Then we do the following to perform erasure recovery of γ coordinates in $C_\lambda^{(2)}$. Let $c^{(2)} \in C_\lambda^{(2)}$ be a codeword with erased coordinates $\mathcal{J}_\gamma = \{j_1, \dots, j_\gamma\}$, and let $\mathcal{J}_\eta = \{j'_1, \dots, j'_\eta\}$ be the coordinates of $c^{(2)}$ that have not been erased. For every $j \in \mathcal{J}_\eta$, compute $c_j = (\lambda_j^{-1})^2 c_j^{(2)}$. Then the vector $(c_j)_{j \in \mathcal{J}_\eta}$ is a codeword of $C^{(2)}$ with coordinates c_i erased for $i \in \mathcal{J}_\gamma$. Since $C^{(2)}$ has γ erasure recovery, we can recover the c_i for $i \in \mathcal{J}_\gamma$. Once recovered, for every $i \in \mathcal{J}_\gamma$, compute $c_i^{(2)} = \lambda_i^2 c_i$. This produces the γ erased coordinates of $c^{(2)}$ in $C_\lambda^{(2)}$. Finally, one can map the $c_i^{(2)}$ for $i \in \mathcal{J}_\gamma$ to the coordinates \mathcal{I}_γ using π , recovering the erasures in $C_{\pi,\lambda}^{(2)}$.

$2^{-\delta}$ -bias Family of Distributions. Let $C, C_\lambda, C_{\pi,\lambda}$ be the uniform distribution over the linear codes generated by $G, G_\lambda, G_{\pi,\lambda}$, respectively. Recall that d^\perp is the dual distance for C . Note that $C_\lambda, C_{\pi,\lambda}$ have dual-distance d^\perp as well. Let $\eta^* = \eta + \gamma$. Since $\text{Bias}_S(C_{\pi,\lambda}) = |\mathbb{F}^{\eta^*}| \widehat{C_{\pi,\lambda}}(S)$ for every $S \in \mathbb{F}^{\eta^*}$, it suffices to show that

$$\mathbb{E}_{\pi,\lambda} \left[\widehat{C_{\pi,\lambda}}(S)^2 \right] \leq \frac{1}{|\mathbb{F}|^{2\eta^*} \cdot 2^\delta}.$$

To begin, first recall the definition of $C_{\pi,\lambda}$:

$$C_{\pi,\lambda} := \{\pi(\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*}) \mid (c_1, \dots, c_{\eta^*}) \in C\}.$$

Next, given any $S \in \mathbb{F}^{\eta^*}$, define $\mathfrak{S}(S) := \{\pi(\lambda_1 S_1, \dots, \lambda_{\eta^*} S_{\eta^*}) \in \mathbb{F}^{\eta^*} \mid \forall \pi \in \mathfrak{S}_{\eta^*} \wedge \lambda \in (\mathbb{F}^*)^{\eta^*}\}$. Note that $\mathfrak{S}(S)$ is equivalently characterized as

$$\mathfrak{S}(S) = \{T = (T_1, \dots, T_{\eta^*}) \in \mathbb{F}^{\eta^*} \mid \text{wt}(T) = \text{wt}(S)\}.$$

It is easy to see that $|\mathcal{S}(S)| = \binom{\eta^*}{w_0} (q-1)^{\eta^* - w_0}$, where $w_0 = \eta^* - \text{wt}(S)$; i.e., w_0 is the number of zeros in S . We prove the following claim.

Claim 1. *For any $S \in \mathbb{F}^{\eta^*}$, we have $\widehat{C}_{\pi, \lambda}(S) = \widehat{C}(\pi^{-1}(S) * \lambda)$.*

Proof. Notice that by definition for any $x \in C_{\pi, \lambda}$, we have $C_{\pi, \lambda}(x) = C(c)$ since $x = \pi(\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*})$ for $c \in C$. This is equivalently stated as $C_{\pi, \lambda}(\pi(c * \lambda)) = C(c)$. For $x = \pi(\lambda_1 y_1, \dots, \lambda_{\eta^*} y_{\eta^*}) \in \mathbb{F}^{\eta^*}$ and any $S \in \mathbb{F}^{\eta^*}$, we have

$$S \cdot x = \sum_{i=1}^{\eta^*} S_i x_i = \sum_{i=1}^{\eta^*} S_i (\lambda_{\pi(i)} y_{\pi(i)}) = \sum_{i=1}^{\eta^*} (S_{\pi^{-1}(i)}) \lambda_i y_i = (\pi^{-1}(S) * \lambda) \cdot y.$$

where $S \cdot x$ is the vector dot product. By definition of $\chi_S(x)$, this implies $\chi_S(x) = \chi_y(\pi^{-1}(S) * \lambda)$. Using these two facts and working directly from the definition of Fourier Transform, we have

$$\begin{aligned} \widehat{C}_{\pi, \lambda}(S) &= \frac{1}{|\mathbb{F}^{\eta^*}|} \sum_{x \in \mathbb{F}^{\eta^*}} C_{\pi, \lambda}(x) \overline{\chi_S(x)} \\ &= \frac{1}{|\mathbb{F}^{\eta^*}|} \sum_{c \in \mathbb{F}^{\eta^*}} C_{\pi, \lambda}(\pi(\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*})) \overline{\chi_S(\pi(\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*}))} \\ &= \frac{1}{|\mathbb{F}^{\eta^*}|} \sum_{c \in \mathbb{F}^{\eta^*}} C(c) \overline{\chi_c(\pi^{-1}(S) * \lambda)} = \widehat{C}(\pi^{-1}(S) * \lambda). \end{aligned}$$

This proves Claim 1. \square

It is easy to see that $\text{wt}(\pi^{-1}(S) * \lambda) = \text{wt}(S)$, so $(\pi^{-1}(S) * \lambda) = T \in \mathcal{S}(S)$. From this fact and Claim 1, we prove the following claim.

Claim 2. *For any $S \in \mathbb{F}^n$, $\mathbb{E}_{\pi, \lambda} [\widehat{C}_{\pi, \lambda}(S)^2] = \mathbb{E}_{T \leftarrow \mathcal{S}(S)} [\widehat{C}(T)^2]$.*

Proof. Suppose we have codeword $x \in C_{\pi, \lambda}$ such that $\pi(\lambda_1 c_1, \dots, \lambda_{\eta^*} c_{\eta^*}) = x$, for some codeword $c \in C$. Let $\{i_1, \dots, i_{w_0}\}$ be the set of indices of 0 in c ; that is, $c_j = 0$ for all $j \in \{i_1, \dots, i_{w_0}\}$. Then for any permutation π , the set $\{\pi(i_0), \dots, \pi(i_{w_0})\}$ is the set of zero indices in x . Note also that for any index $j \notin \{\pi(i_0), \dots, \pi(i_{w_0})\}$, we have $x_j \neq 0$. If this was not the case, then we have $x_j = c_{\pi^{-1}(j)} \lambda_{\pi^{-1}(j)} = 0$. Since $j \notin \{\pi(i_0), \dots, \pi(i_{w_0})\}$, this implies $\pi^{-1}(j) \notin \{i_0, \dots, i_{w_0}\}$, which further implies that $c_{\pi^{-1}(j)} \neq 0$. This is a contradiction since $\lambda \in (\mathbb{F}^*)^{\eta^*}$. Thus any permutation π must map the zeros of S to the zeros of c , and there are $w_0!(\eta^* - w_0)!$ such permutations. Notice now that for any $c_k = 0$, λ_k can take any value in \mathbb{F}^* , so we have $(q-1)^{w_0}$ such choices. Furthermore, if $c_k \neq 0$ and $\lambda_k c_k = x_{\pi^{-1}(k)} \neq 0$, then there is exactly one value $\lambda_k \in \mathbb{F}^*$ which

satisfies this equation. Putting it all together, we have

$$\begin{aligned}
 \mathbb{E}_{\pi, \lambda} \left[\widehat{C}_{\pi, \lambda}(S)^2 \right] &= \frac{1}{\eta^*!(q-1)^{\eta^*}} \sum_{\pi, \lambda} \widehat{C}_{\pi, \lambda}(S)^2 = \frac{1}{\eta^*!(q-1)^{\eta^*}} \sum_{\pi, \lambda} \widehat{C}(\pi^{-1}(S) * \lambda)^2 \\
 &= \frac{(w_0!(\eta^* - w_0)!(q-1)^{w_0})}{\eta^*!(q-1)^{\eta^*}} \sum_{T \in \mathcal{S}(S)} \widehat{C}(T)^2 \\
 &= \frac{w_0!(\eta^* - w_0)!}{\eta^*!(q-1)^{\eta^* - w_0}} \sum_{T \in \mathcal{S}(S)} \widehat{C}(T)^2 \\
 &= \left(\binom{\eta^*}{w_0} (q-1)^{\eta^* - w_0} \right)^{-1} \sum_{T \in \mathcal{S}(S)} \widehat{C}(T)^2 = \mathbb{E}_{T \in \mathcal{S}(S)} \left[\widehat{C}(T)^2 \right].
 \end{aligned}$$

where the first line of equality follows from Claim 1. This proves Claim 2. \square

With Claim 2, we now are interested in finding δ such that for $0^{\eta^*} \neq S \in \mathbb{F}^{\eta^*}$

$$\mathbb{E}_{T \in \mathcal{S}(S)} \left[\widehat{C}(T)^2 \right] \leq \frac{1}{|\mathbb{F}|^{2\eta^*} 2^\delta}.$$

We note that since C is a linear code, C has non-zero Fourier coefficients only at codewords in C^\perp .

Claim 3. For all $S \in \mathbb{F}^{\eta^*}$, $\widehat{C}(S) = \begin{cases} \frac{1}{|\mathbb{F}|^{\eta^*}} & S \in C^\perp \\ 0 & \text{otherwise.} \end{cases}$

Let $A_w = |C^\perp \cap \mathcal{S}(S)|$, where $w = \eta^* - w_0 = \text{wt}(S)$. Intuitively, A_w is the number of codewords in C^\perp with weight w . Then from Claim 3, we have

$$\mathbb{E}_{T \in \mathcal{S}(S)} \left[\widehat{C}(T)^2 \right] = \frac{|C^\perp \cap \mathcal{S}(S)|}{|\mathbb{F}|^{2\eta^*} (\eta^* - \text{wt}(S)) (q-1)^{\text{wt}(S)}} = \frac{A_w}{|\mathbb{F}|^{2\eta^*} \binom{\eta^*}{w} (q-1)^w}$$

Now, our goal is to upper bound A_w . Towards this goal, the weight enumerator for the code C^\perp is defined as the following polynomial:

$$W_{C^\perp}(x) = \sum_{c \in C^\perp} x^{\eta^* - \text{wt}(c)}.$$

This polynomial can equivalently be written in the following manner:

$$W_{C^\perp}(x) = \sum_{w \in \{0, \dots, \eta^*\}} A_w x^{\eta^* - w}.$$

Define $a = \eta^* - d^\perp$.

Imported Theorem 1 (Exercise 1.1.15 from [41]). We have the relation

$$W_{C^\perp}(x) = x^{\eta^*} + \sum_{i=0}^a B_i (x-1)^i, \text{ where}$$

$$B_i = \sum_{j=\eta^*-a}^{\eta^*-i} \binom{\eta^*-j}{i} A_j \geq 0 \quad A_i = \sum_{j=\eta^*-i}^a (-1)^{\eta^*+i+j} \binom{j}{\eta^*-i} B_j.$$

For weight $w \in \{d^\perp, \dots, \eta^*\}$, we use the following expression to estimate A_w .

$$A_w = \binom{\eta^*-w}{\eta^*-w} B_{\eta^*-w} - \binom{\eta^*-w+1}{\eta^*-w} B_{\eta^*-w+1} + \dots \pm \binom{\eta^*-d^\perp}{\eta^*-w} B_{\eta^*-d^\perp}$$

Since we are interested in the asymptotic behavior (and not the exact value) of A_w , we note that $\lg A_w \sim \lg \Gamma(w)$, where

$$\Gamma(w) = \max \left\{ \binom{\eta^*-w}{\eta^*-w} B_{\eta^*-w}, \binom{\eta^*-w+1}{\eta^*-w} B_{\eta^*-w+1}, \dots, \binom{\eta^*-d^\perp}{\eta^*-w} B_{\eta^*-d^\perp} \right\}.$$

Thus, it suffices to compute $\Gamma(w)$ for every w , and then the bias. We present this precise asymptotic calculation in the full version of our work [8]. This calculation yields

$$\delta = \left(d^\perp + \frac{\eta^*}{\sqrt{q}-1} - 1 \right) \left(\lg(q-1) - \mathbf{h}_2 \left(\frac{1}{q+1} \right) \right) - \frac{\eta^*}{\sqrt{q}-1} \lg q,$$

which completes the proof. \square

4 Construction of Correlation Extractor

Our main sub-protocol for Theorem 3 takes $\text{ROLE}(\mathbb{F})$ as the initial correlation and produces secure $\text{ROLE}(\mathbb{F})$. Towards this, we define a $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor formally below.

Definition 7 ($(\eta, \gamma, t, \varepsilon)$ - $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor). *Let $(R_A, R_B) = (\text{ROLE}(\mathbb{F}))^\eta$ be correlated randomness. An $(\eta, \gamma, t, \varepsilon)$ - $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor is a two-party interactive protocol in the $(R_A, R_B)^{[t]}$ -hybrid that securely implements the $(\text{ROLE}(\mathbb{F}))^\gamma$ functionality against information-theoretic semi-honest adversaries with ε simulation error.*

Let $(u_i, v_i) \in \mathbb{F}^2$ and $(r_i, z_i) \in \mathbb{F}^2$ be the shares of Alice and Bob, respectively, in the i^{th} output ROLE instance. The *correctness* condition says that the receiver's output is correct in all γ instances of ROLE , i.e., $z_i = u_i r_i + v_i$ for all $i \in [\gamma]$. The *privacy* requirement says the following: A corrupt sender (resp., receiver) cannot distinguish between $\{r_i\}_{i \in [\gamma]}$ (resp., $\{u_i\}_{i \in [\gamma]}$) and $U_{\mathbb{F}^\gamma}$ with advantage more than ε .

In Sect. 4.1, we give our construction for Theorem 3. Later, in Sect. 4.3, we build on this to give our construction for Theorem 2.

4.1 Protocol for $\text{ROLE}(\mathbb{F})$ correlation extractor

As already mentioned in Sect. 1.4, to prove Theorem 3, our main building block will be $(\eta, \gamma, t, \varepsilon)$ - $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor (see Definition 7). That is, the parties start with η samples of the $\text{ROLE}(\mathbb{F})$ correlation such that size of each party's share is $n = 2\eta \log |\mathbb{F}|$ bits. The adversarial party gets t bits of leakage. The protocol produces $(\text{ROLE}(\mathbb{F}))^\gamma$ with simulation error ε . We give the formal description of the protocol, inspired by the Massey secret sharing scheme [32], in Fig. 4. Note that our protocol is round-optimal and uses a family of distributions $\mathcal{C} = \{C_j\}_{j \in \mathcal{J}}$ that satisfies Property 1 with parameters δ and γ .

$(\eta, \gamma, t, \varepsilon)$ - $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ Extractor:

Let $\mathcal{C} = \{C_j : j \in \mathcal{J}\}$ be a family of distributions over $\mathbb{F}^{\eta+\gamma}$ satisfying Property 1 for appropriate values of δ and γ .

Hybrid (Random Correlations): Client A gets random $(a_{[\eta]}, b_{[\eta]}) \in \mathbb{F}^{2\eta}$ and Client B gets random $(x_{[\eta]}, z_{[\eta]}) \in \mathbb{F}^{2\eta}$ such that for all $i \in \{1, 2, \dots, \eta\}$, $a_i x_i + b_i = z_i$.

1. *Code Generation.* Client B samples $j \xleftarrow{\$} \mathcal{J}$.
2. *ROLE Extraction Protocol.*
 - (a) Client B picks random $r = (r_{-\gamma}, \dots, r_{-1}, r_1, \dots, r_\eta) \sim C_j$ and computes $m_{[\eta]} = r_{[\eta]} + x_{[\eta]}$. Client B sends $(m_{[\eta]}, j)$ to client A .
 - (b) Client A picks the same distribution C_j as client B . Client A picks random $u = (u_{-\gamma}, \dots, u_{-1}, u_1, \dots, u_\eta) \sim C_j$ and random $v = (v_{-\gamma}, \dots, v_{-1}, v_1, \dots, v_\eta) \sim C_j^{(2)}$. Client A computes $\alpha_{[\eta]} = u_{[\eta]} - a_{[\eta]}$, and $\beta_{[\eta]} = a_{[\eta]} * m_{[\eta]} + b_{[\eta]} + v_{[\eta]}$ and sends $(\alpha_{[\eta]}, \beta_{[\eta]})$ to Client B .
 - (c) Client B computes $t_{[\eta]} = (\alpha_{[\eta]} * r_{[\eta]}) + \beta_{[\eta]} - z_{[\eta]}$. Client B performs erasure recovery on $t_{[\eta]}$ for $C_j^{(2)}$ to obtain $t_{[-\gamma]}$.
 - (d) Client A outputs $\{u_i, v_i\}_{i \in \{-\gamma, \dots, -1\}}$ and Client B outputs $\{r_i, t_i\}_{i \in \{-\gamma, \dots, -1\}}$

Fig. 4. $\text{ROLE}(\mathbb{F})$ -to- $\text{ROLE}(\mathbb{F})$ extractor protocol.

Next, we use the ROT embedding technique from [9] to embed σ ROTs in each fresh $\text{ROLE}(\mathbb{F})$ obtained from above protocol. For example, we can embed two ROTs into one $\text{ROLE}(\mathbb{GF}[2^6])$. Using this we get production $m = 2\sigma\gamma$, i.e., we get $m/2 = \sigma\gamma$ secure ROTs. We note that the protocol from [9] is round-optimal, achieves perfect security and composes in parallel with our protocol in Fig. 4. Hence, we maintain round-optimality (see Sect. 4.2).

Correctness of Fig. 4. The following lemma characterizes the correctness of the scheme presented in Fig. 4.

Lemma 3 (Correctness). *If the family of distributions $\mathcal{C} = \{C_j\}_{j \in \mathcal{J}}$ satisfies Property 1, i.e., erasure recovery of first γ coordinates in Schur product, then for all $i \in \{-\gamma, \dots, -1\}$, it holds that $t_i = u_i r_i + v_i$.*

Proof. First, we prove the following claim.

Claim 4. *For all $i \in [\eta]$, it holds that $t_i = u_i r_i + v_i$.*

This claim follows from the following derivation.

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

From the above claim, we have that $t_{[\eta]} = u_{[\eta]} * r_{[\eta]} + v_{[\eta]}$. From the protocol, we have that $u, r \in C_j$ and $v \in C_j^{(2)}$. Consider $\tilde{t} = u * r + v \in C_j^{(2)}$. Note that $t_i = \tilde{t}_i$ for all $i \in [\eta]$. Hence, when client B performs erasure recovery on $t_{[\eta]}$ for a codeword in $C_j^{(2)}$, it would get $\tilde{t}_{[-\gamma]}$. This follows from erasure recovery guarantee for first γ coordinates by Property 1. \square

Security of Fig. 4. To argue the security, we prove that the protocol is a secure implementation of $(\text{ROLE}(\mathbb{F}))^\gamma$ functionality against an information-theoretic semi-honest adversary that corrupts either the sender or the receiver and leaks at most t -bits from the secret share of the honest party at the beginning of the protocol. At a high level, we prove the security of our protocol by reducing it exactly to our unpredictability lemma.

Lemma 4 (Unpredictability Lemma). *Let $\mathcal{C} = \{C_j : j \in \mathcal{J}\}$ be a $2^{-\delta}$ -biased family of linear code distributions over \mathbb{F}^{η^*} , where $\eta^* = \gamma + \eta$. Consider the following game between an honest challenger \mathcal{H} and an adversary \mathcal{A} :*

1. \mathcal{H} samples $m_{[\eta]} \sim U_{\mathbb{F}^\eta}$.
2. \mathcal{A} sends a leakage function $\mathcal{L}: \mathbb{F}^\eta \rightarrow \{0, 1\}^t$.
3. \mathcal{H} sends $\mathcal{L}(m_{[\eta]})$ to \mathcal{A} .
4. \mathcal{H} samples $j \xleftarrow{\$} \mathcal{J}$. \mathcal{H} samples a uniform random $(r_{-\gamma}, \dots, r_{-1}, r_1, \dots, r_\eta) \in C_j$. \mathcal{H} computes $y_{[\eta]} = r_{[\eta]} + m_{[\eta]}$ and sends $(y_{[\eta]}, j)$ to \mathcal{A} .
 \mathcal{H} picks $b \xleftarrow{\$} \{0, 1\}$. If $b = 0$, then \mathcal{H} sends $\text{chal} = r_{[-\gamma]}$ to \mathcal{A} ; otherwise (if $b = 1$) \mathcal{H} sends $\text{chal} = u_{[\gamma]} \sim U_{\mathbb{F}^\gamma}$.
5. \mathcal{A} sends $\tilde{b} \in \{0, 1\}$.

The adversary \mathcal{A} wins the game if $b = \tilde{b}$. For any \mathcal{A} , the advantage of the adversary is $\varepsilon \leq \frac{1}{2} \sqrt{\frac{|\mathbb{F}|^\gamma 2^t}{2^\delta}}$.

Proof. Let $M_{[\eta]}$ be the distribution corresponding to $m_{[\eta]}$. Consider $M'_{[\eta+\gamma]} = (0^\gamma, M_{[\eta]})$. By Imported Lemma 1, $\tilde{\mathbf{H}}_\infty(M' | \mathcal{L}(M')) \geq \eta \log |\mathbb{F}| - t$. Recall that $\mathcal{C} = \{C_j : j \in \mathcal{J}\}$ is a $2^{-\delta}$ -bias family of distributions over $\mathbb{F}^{\eta+\gamma}$. Then, by Theorem 5, we have the following as desired:

$$\text{SD}((C_{\mathcal{J}} \oplus M', \mathcal{L}(M'), \mathcal{J}), (U_{\mathbb{F}^{\eta+\gamma}}, \mathcal{L}(M'), \mathcal{J})) \leq \frac{1}{2} \left(\frac{2^t \cdot |\mathbb{F}|^{\eta+\gamma}}{2^\delta \cdot |\mathbb{F}|^\eta} \right)^{\frac{1}{2}} = \frac{1}{2} \sqrt{\frac{|\mathbb{F}|^\gamma 2^t}{2^\delta}}.$$

\square

We note that this lemma crucially relies on a family of small-bias distributions. Next, we prove the following security lemma.

Lemma 5. *The simulation error of our protocol is $\varepsilon \leq \sqrt{\frac{|\mathbb{F}|^\gamma 2^t}{2^\delta}}$, where t is the number of bits of leakage, and γ and δ are the parameters in Property 1 for the family of distributions \mathcal{C} .*

Proof. We first prove Bob privacy followed by Alice privacy.

Bob Privacy. In order to prove privacy of client B against a semi-honest client A , it suffices to show that the adversary cannot distinguish between Bob's secret values $(r_{-\gamma}, \dots, r_{-1})$ and $U_{\mathbb{F}^\gamma}$. We show that the statistical distance of $(r_{-\gamma}, \dots, r_{-1})$ and $U_{\mathbb{F}^\gamma}$ given the view of the adversary is at most ε , where ε is defined above.

We observe that client B 's privacy reduces directly to our unpredictability lemma (Lemma 4) for the following variables. Let $X_{[\eta]}$ be the random variable denoting B 's input in the initial correlations. Then, $X_{[\eta]}$ is uniform over \mathbb{F}^η . Note that the adversary gets $L = \mathcal{L}(X_{[\eta]})$ that is at most t -bits of leakage. Next, the honest client B picks $j \xleftarrow{\$} \mathcal{J}$ and a random $r = (r_{-\gamma}, \dots, r_{-1}, r_1, \dots, r_\eta) \in C_j$. Client B sends $m_{[\eta]} = r_{[\eta]} + x_{[\eta]}$. This is exactly the game between the honest challenger and an semi-honest adversary in the unpredictability lemma (see Lemma 4). Hence, the adversary cannot distinguish between $r_{[-\gamma]}$ and $U_{\mathbb{F}^\gamma}$ with probability more than ε .

Alice Privacy. In order to prove privacy of client A against a semi-honest client B , it suffices to show that the adversary cannot distinguish between Alice's secret values $(u_{-\gamma}, \dots, u_{-1})$ and $U_{\mathbb{F}^\gamma}$. We show that the statistical distance of $(u_{-\gamma}, \dots, u_{-1})$ and $U_{\mathbb{F}^\gamma}$ given the view of the adversary is at most ε , where ε is defined above by reducing to our unpredictability lemma (see Lemma 4).

Let $A_{[\eta]}$ denote the random variable corresponding to the client A 's input $a_{[\eta]}$ in the initial correlations. Then, without loss of generality, the adversary receives t -bits of leakage $\mathcal{L}(A_{[\eta]})$. We show a formal reduction to Lemma 4 in Fig. 5. Given an adversary \mathcal{A} who can distinguish between $(u_{-\gamma}, \dots, u_{-1})$ and $U_{\mathbb{F}^\gamma}$, we construct an adversary \mathcal{A}' against an honest challenger \mathcal{H} of Lemma 4 with identical advantage. It is easy to see that this reduction is perfect. The only differences in the simulator from the actual protocol are as follows. In the simulation, the index j of the distribution is picked by the honest challenger \mathcal{H} instead of client B . This is identical because client B is a semi-honest adversary.

Also, the simulator \mathcal{A}' generates $\beta_{[\eta]}$ slightly differently. We claim that the distribution of $\beta_{[\eta]}$ in the simulation is identical to that of real protocol.

This holds by correctness of the protocol: $t_{[\eta]} = u_{[\eta]} * r_{[\eta]} + v_{[\eta]} = (\alpha_{[\eta]} * r_{[\eta]}) + \beta_{[\eta]} - z_{[\eta]}$. Hence, $\beta_{[\eta]} = (u_{[\eta]} * r_{[\eta]} + v_{[\eta]}) - (\alpha_{[\eta]} * r_{[\eta]}) + z_{[\eta]} = w_{[\eta]} - (\alpha_{[\eta]} * r_{[\eta]}) + z_{[\eta]}$, where $w_{[-\gamma, \eta]}$ is chosen as a random codeword in $C_j^{(2)}$. This holds because in the real protocol $v_{[-\gamma, \eta]}$ is chosen as a random codeword in $C_j^{(2)}$ and $u_{[-\gamma, \eta]} * r_{[-\gamma, \eta]} \in C_j^{(2)}$. Here, we denote by $[-\gamma, \eta]$ the set $\{-\gamma, \dots, -1, 1, \dots, \eta\}$. \square

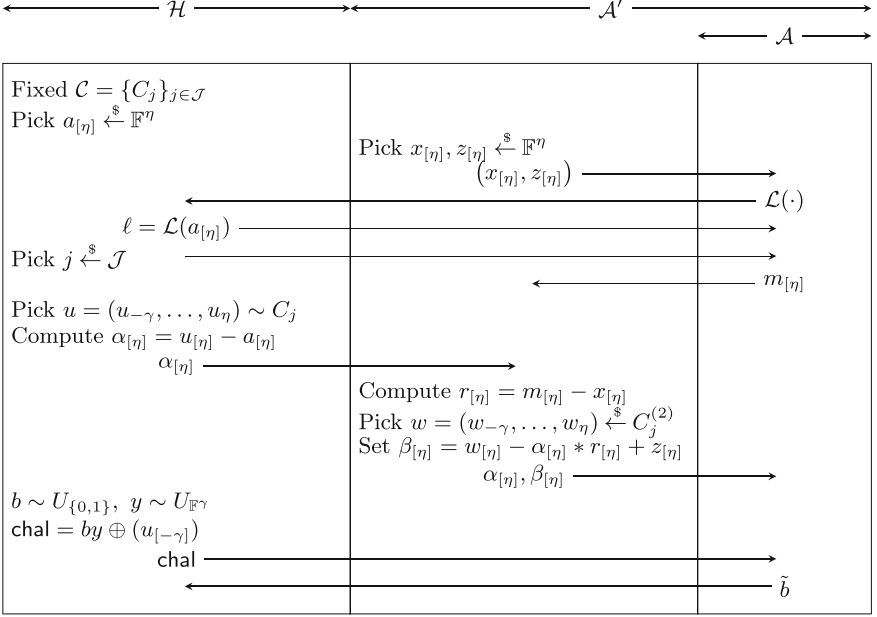


Fig. 5. Simulator for Alice Privacy.

4.2 OT Embedding

The second conceptual block is the ROT embedding protocol from [9], referred to as the BMN embedding protocol, that embeds a constant number of ROT samples into one sample of $\text{ROLE}(\mathbb{F})$, where \mathbb{F} is a finite field of characteristic 2. The BMN embedding protocol is a two-message perfectly semi-honest secure protocol. For example, asymptotically, [9] embeds $(s)^{1-o(1)}$ samples of ROT into one sample of the $\text{ROLE}(\mathbb{GF}[2^s])$ correlation. However, for reasonable values of s , say for $s \leq 2^{50}$, a recursive embedding embeds $s^{\log 10 / \log 32}$ samples of ROT into one sample of the $\text{ROLE}(\mathbb{GF}[2^s])$ correlation, and this embedding is more efficient than the asymptotically good one. Below, we show that this protocol composes in parallel with our protocol in Fig. 4 to give our overall round optimal protocol for (n, m, t, ε) -correlation extractor for $\text{ROLE}(\mathbb{F})$ correlation satisfying Theorem 3.

We note that the BMN embedding protocol satisfies the following additional properties. (1) The first message is sent by client B , and (2) this message depends only on the first share of client B in $\text{ROLE}(\mathbb{F})$ (this refers to r_i in Fig. 4) and does not depend on the second share (this refers to t_i in Fig. 4). With these properties, the BMN embedding protocol can be run in parallel with the protocol in Fig. 4. Also, since the BMN protocol satisfies perfect correctness and perfect security, to prove overall security, it suffices to prove the correctness and security of our protocol in Fig. 4. This holds because we are in the semi-honest information theoretic setting.

4.3 Protocol for ROT Extractor (Theorem 2)

In this section, we describe a protocol to construct $(\text{ROLE}(\mathbb{F}^\eta)^{[t]})$ using $(\text{ROLE}^\eta)^{[t]}$, that is the starting point of our protocol in Sect. 4.1. This would prove Theorem 2. Here, $\text{ROLE} := \text{ROLE}(\mathbb{GF}[2])$. Recall that ROLE and ROT are equivalent.

One of the several fascinating applications of algebraic function fields pioneered by the seminal work of Chudnovsky and Chudnovsky [14], is the application to efficiently multiply over an extension field using multiplications over the base field. For example, 6 multiplications over $\mathbb{GF}[2]$ suffice to perform one multiplication over $\mathbb{GF}[2^3]$, or 15 multiplications over $\mathbb{GF}[2]$ suffice for one multiplication over $\mathbb{GF}[2^6]$ (cf., Table 1 in [12]).

Our first step of the correlation extractor for $(\text{ROLE}^\eta)^{[t]}$ uses these efficient multiplication algorithms to (perfectly and securely) implement $(\text{ROLE}(\mathbb{F}^\eta)^{[t]})$, where $\mathbb{F} = \mathbb{GF}(2^\alpha)$ is a finite field with characteristic 2.

We start by describing a protocol for realizing one $\text{ROLE}(\mathbb{F})$ using ROLE^ℓ , i.e., ℓ independent samples of ROLE (in the absence of leakage) in Fig. 6. Our protocol implements, for instance, one sample of $\text{ROLE}(\mathbb{GF}[2^3])$ correlation using 6 samples from the ROT correlation in two rounds. Our protocol uses a multiplication friendly code \mathcal{D} over $\{0, 1\}^\ell$ and encodes messages in \mathbb{F} . That is, $\mathcal{D} * \mathcal{D} = \mathcal{D}^{(2)} \subset \{0, 1\}^\ell$ is also a code for \mathbb{F} . Later, we show how to extend this to the leakage setting.

Protocol for $\text{ROLE}(\mathbb{F})$ in ROLE^ℓ hybrid:

Let $\mathcal{D} \subset \{0, 1\}^\ell$ be a multiplication friendly code that encodes messages in $\mathbb{F} = \mathbb{GF}(2^\alpha)$. Let $(\text{Enc}_{\mathcal{D}}, \text{Dec}_{\mathcal{D}})$ (resp., $\text{Enc}_{\mathcal{D}^{(2)}}, \text{Dec}_{\mathcal{D}^{(2)}}$) be encoding and decoding procedures for \mathcal{D} (resp., $\mathcal{D}^{(2)}$).

Hybrid ROLE^ℓ : Client A and client B have access to a single call to ROLE^ℓ functionality. Client A will play as the sender and client B will play as the receiver.

Inputs: Client A has inputs $a_0, b_0 \in \mathbb{F}$ and client B has inputs $x_0 \in \mathbb{F}$.

1. Client A picks a random codeword $a_{[\ell]} \sim \text{Enc}_{\mathcal{D}}(a_0)$ and $b_{[\ell]} \sim \text{Enc}_{\mathcal{D}^{(2)}}(b_0)$ Client A sends $a_{[\ell]}, b_{[\ell]}$ as sender inputs to ROLE^ℓ functionality.
2. Client B picks a random codeword $x_{[\ell]} \sim \text{Enc}_{\mathcal{D}}(x_0)$ and sends $x_{[\ell]}$ as receiver input to ROLE^ℓ . Client B gets $z_{[\ell]} \in \{0, 1\}^\ell$ as output. Client B runs $\text{Dec}_{\mathcal{D}^{(2)}}(z_{[\ell]})$ to obtain $z_0 \in \mathbb{F}$.
3. Client A outputs a_0, b_0 and Client B outputs x_0, z_0 .

Fig. 6. Perfectly secure protocol for $\text{ROLE}(\mathbb{F})$ in ROLE^ℓ hybrid.

Security Guarantee. It is easy to see that the protocol in Fig. 6 is a perfectly secure realization of $\text{ROLE}(\mathbb{F})$ in the ROLE^ℓ -hybrid against a semi-honest adversary using the fact that \mathcal{D} is a multiplication friendly code for \mathbb{F} . Moreover, [26] proved the following useful lemma to argue t -leaky realization of $\text{ROLE}(\mathbb{F})$ if the perfect oracle call to ROLE^ℓ is replaced by a t -leaky oracle.

Imported Lemma 3 ([26]). *Let π be a perfectly secure (resp., statistically ε secure) realization of f in the g -hybrid model, where π makes a single call to g . Then, π is also a perfectly secure (resp., statistically ε secure) realization of $f^{[t]}$ in the $g^{[t]}$ -hybrid model.*

Using the above lemma, we get that the protocol in Fig. 6 is a perfect realization of $(\text{ROLE}(\mathbb{F}))^{[t]}$ in $(\text{ROLE}^\ell)^{[t]}$ -hybrid. Finally, by running the protocol of Fig. 6 in parallel for η samples of $\text{ROLE}(\mathbb{F})$ and using Imported Lemma 3, we get a perfectly secure protocol for $(\text{ROLE}(\mathbb{F})^\eta)^{[t]}$ in $(\text{ROLE}^{\eta\ell})^{[t]}$ -hybrid.

Round Optimality. To realize the round-optimality in Theorem 2, we can run the protocols in Figs. 6 and 4 in parallel. We note that the first messages of protocols in Figs. 6 and 4 can be sent together. This is because the first message of client B in protocol of Fig. 4 is independent of the second message in Fig. 6. The security holds because we are in the semi-honest information theoretic setting. Hence, overall round complexity is still 2.

5 Parameter Comparison

5.1 Correlation Extractor from $\text{ROLE}(\mathbb{F})$ (Theorem 3)

In this section, we compare our correlation extractor for $\text{ROLE}(\mathbb{F})$ correlation, where \mathbb{F} is a constant size field, with the BMN correlation extractor [9].

BMN Correlation Extractor [9]. The BMN correlation extractor emphasizes high resilience while achieving multiple ROTs as output. Roughly, they show the following. If parties start with the $\text{IP}(\mathbb{GF}[2^{\Delta n}]^{1/\Delta})$ correlation, then they (roughly) achieve $\frac{1}{2} - \Delta$ fractional resilience with production that depends on (Δn) . Here, Δ has to be the inverse of an even natural number ≥ 4 .

In particular, the $\text{IP}(\mathbb{GF}[2^{n/4}]^4)$ correlation¹² achieves the highest production using the BMN correlation extractor. The resilience of this correlation is $(\frac{1}{4} - g)$, where $g \in (0, 1/4]$ is a positive constant. Then the BMN correlation extractor produces at most $(n/4)^{\log_{10}/\log_{38}} \approx (n/4)^{0.633}$ fresh samples from the ROT correlation as output when $n \leq 2^{50}$. This implies that the production is $m \approx 2 \cdot (n/4)^{0.633}$, because each ROT sample produces private shares that are two-bits long. For $n = 10^3$, the production is $m \leq 66$, for $n = 10^6$ the production is $m \leq 5,223$, and for $n = 10^9$ the production is $m \leq 413,913$. We emphasize that the BMN extractor *cannot* increase its production any further by sacrificing its leakage resilience and going below $1/4$.

Our Correlation Extractor for $\text{ROLE}(\mathbb{F})$. We shall use \mathbb{F} such that $q = |\mathbb{F}|$ is an even power of 2. For the suitable Algebraic Geometry codes [19] to exist, we need $q \geq 49$. Since, the last step of our construction uses the OT embedding

¹² Recall that the inner-product correlation $\text{IP}(\mathbb{K}^s)$ over finite field \mathbb{K} samples random $r_A = (u_1, \dots, u_s) \in \mathbb{K}^s$ and $r_B = (v_1, \dots, v_s) \in \mathbb{K}^s$ such that $u_1 v_1 + \dots + u_s v_s = 0$.

Field \mathbb{F}	# of OTs Embedded Per $\text{ROLE}(\mathbb{F})$ [9]	Production ($\alpha = m/n$)
$\mathbb{GF}[2^6]$	2	4.83%
$\mathbb{GF}[2^8]$	3	11.39%
$\mathbb{GF}[2^{10}]$	4	15.59%
$\mathbb{GF}[2^{14}]$	5	16.32%
$\mathbb{GF}[2^{20}]$	6	14.31%

Fig. 7. The production rate of our correlation extractor for $\text{ROLE}(\mathbb{F})$, where $\beta = t/n = 1\%$ rate of leakage using different finite fields.

Field $\mathbb{F} = \mathbb{GF}[2^s]$	Bilinear Comp. Mult. $\mu_2(s)$ [12]	$n' = \frac{s}{\mu_2(s)}n$	$\beta' = \frac{\mu_2(s)}{s}\beta$	OT Embed. [9]	α'	$\alpha = \frac{s}{\mu_2(s)}\alpha'$
$\mathbb{GF}[2^6]$	15	$\frac{6}{15}n$	2.50%	2	4.05%	1.62%
$\mathbb{GF}[2^8]$	24	$\frac{8}{24}n$	3.00%	3	10.07%	3.35%
$\mathbb{GF}[2^{10}]$	33	$\frac{10}{33}n$	3.30%	4	13.86%	4.20%
$\mathbb{GF}[2^{14}]$	51	$\frac{14}{51}n$	3.64%	5	14.46%	3.97%
$\mathbb{GF}[2^{20}]$	81	$\frac{20}{81}n$	4.05%	6	12.48%	3.08%

Fig. 8. The production rate of our correlation extractor for ROT. We are given n -bit shares of the $\text{ROT}^{n/2}$ correlation, and fix $\beta = t/n = 1\%$ fractional leakage. Each row corresponds to using our $\text{ROLE}(\mathbb{F})$ -to-ROT correlation extractor as an intermediate step. The final column represents the production rate $\alpha = m/n$ of our ROT-to-ROT correlation extractor corresponding to the choice of the finite field \mathbb{F} .

technique introduced by BMN [9], we need to consider only the smallest fields that allow a particular number of OT embeddings. Based on this observation, for fractional resilience $\beta = (t/n) = 1\%$, Fig. 7 presents the achievable production rate $\alpha = (m/n)$. Note that the Algebraic Geometry codes become better with increasing q , but the BMN OT embedding becomes worse. So, the optimum $\alpha = 16.32\%$ is achieved for $\mathbb{F} = \mathbb{GF}[2^{14}]$. For $n = 10^3$, for example, the production is $m = 163$, for $n = 10^6$ the production is $m = 163,200$, and for $n = 10^9$ the production is $m = 163,200,000$. In Fig. 9 (Sect. 6), we demonstrate the trade-off between leakage rate (Y-axis) with production rate (X-axis). We note that even in the high leakage setting, for instance, for $\beta = 20\%$, we have $\alpha \approx 3\%$. Hence, the production is $m \approx 30$, for $n = 10^6$ the production is $m \approx 30,000$, and for $n = 10^9$ the production is $m \approx 30,000,000$. Our production is overwhelmingly higher than the BMN production rate.

5.2 Correlation Extractor for ROT (Theorem 2)

In this section we compare our construction with the GIMS [22] correlation extractor from ROT. The IKOS [26] correlation extractor is a feasibility result with minuscule fractional resilience and production rate.

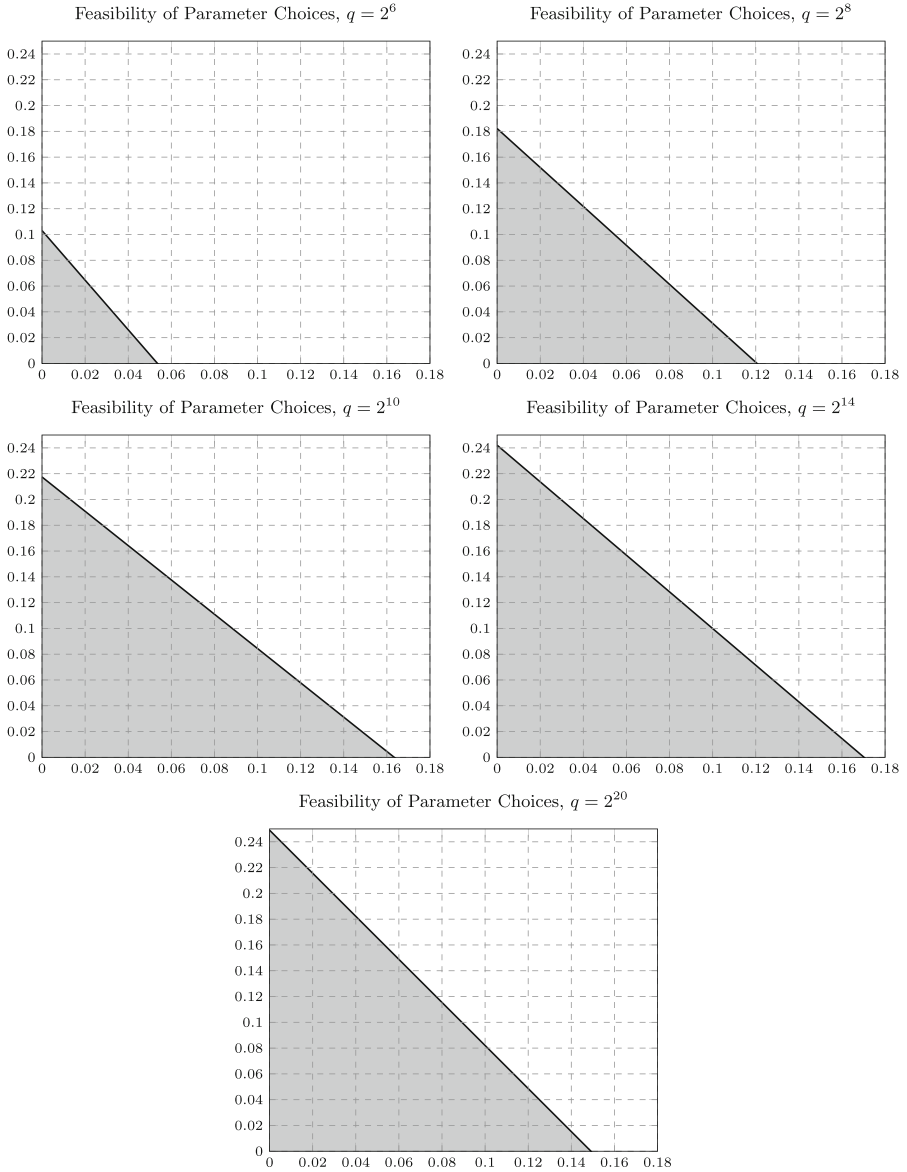


Fig. 9. A comparison of the feasibility regions for our correlation extractors for $\text{ROLE}(\mathbb{F})$ for various finite fields \mathbb{F} of characteristic 2. For each plot, the X-axis represents the relative production rate $\alpha = m/n$ and the Y-axis represents the fractional leakage resilience $\beta = t/n$.

GIMS Production. The GIMS correlation extractor for ROT [22] trades-off simulation error to achieve higher production by sub-sampling the precomputed ROTs. For $\beta = (t/n) = 1\%$ fractional leakage, the GIMS correlation extractor achieves (roughly) $m = n/4p$ production with $\varepsilon = m \cdot 2^{-p/4}$ simulation error. To achieve negligible simulation error, suppose $p = \log^2(n)$. For this setting, at $n = 10^3$, $n = 10^6$, and $n = 10^9$, the GIMS correlation extractor obtains $m = 3$, $m = 625$, and $m = 277,777$, respectively. These numbers are significantly lower than what our construction achieves.

Our Production. We use a bilinear multiplication algorithm to realize one $\text{ROLE}(\mathbb{F})$ by performing several ROT. For example, we use $\mu_2(s) = 15$ ROTs to implement one $\text{ROLE}(\mathbb{GF}[2^s])$, where $s = 6$. Thus, our original n -bit share changes into n' -bit share, where $n' = (6/15)n$ while preserving the leakage $t = \beta n$. So, the fractional leakage now becomes $t = \beta' n'$, where $\beta' = (15/6)\beta$. Now, we can compute the production $m' = \alpha' n' = \alpha n$.

The highest rate is achieved for $s = 10$, i.e., constructing the correlation extractor for ROT via the correlation extractor for $\text{ROLE}(\mathbb{GF}[2^{10}])$. For this choice, our correlation extractor achieves production rate $\alpha = (m/n) = 4.20\%$, if the fractional leakage is $\beta = (t/n) = 1\%$. For $n = 10^3$, $n = 10^6$, and $n = 10^9$, our construction obtains $m = 42$, $m = 42,000$, and $m = 42,000,000$, respectively.

5.3 Close to Optimal Resilience

An interesting facet of our correlation extractor for $\text{ROLE}(\mathbb{F})$ is the following. As $q = |\mathbb{F}|$ increases, the maximum fractional resilience, i.e., the intercept of the feasibility curve on the Y -axis, tends to $1/4$. Ishai et al. [27] showed that any correlation extractor cannot be resilient to fractional leakage $\beta = (t/n) = 25\%$. For every $g \in (0, 1/4]$, we show that, by choosing sufficiently large q , we can achieve positive production rate $\alpha = (m/n)$ for $\beta = (1/4 - g)$. Thus, our family of correlation extractors (for larger, albeit constant-size, finite fields) achieve near optimal fractional resilience. Figure 9 (Sect. 6) demonstrates this phenomenon for a few values of q . The proof of this result, which proves Theorem 4, can be found in the full version of our work [8].

6 Parameter Comparison Graphs

In this section we highlight the feasibility of parameters for our ROT to ROT correlation extractor (Theorem 2) for a few representative values of $q = |\mathbb{F}|$.

The shaded regions in the graphs in Fig. 9 represent the feasible parameter choices. In particular, the X -axis represents the production rate m/n and the Y -axis represents the leakage rate t/n given our parameter choices. The full version of the paper [8] details the calculation of the feasible parameters.

Note that, as the size of the field \mathbb{F} increases, the quality of the algebraic geometric code used in our construction increases. This observation translates

into higher possible production values and leakage resilience, which is illustrated by increasing $q = 2^6$ to $q = 2^{14}$. However, as the size of the field \mathbb{F} increases, the efficiency of the BMN embedding [9] reduces, potentially reducing the overall production rate (for example, increasing $q = 2^{14}$ to $q = 2^{20}$).

Finally, as noted earlier, the feasibility graphs demonstrate that our family of correlation extractors achieve near optimal fractional resilience. That is, as the size of the field \mathbb{F} increases, the fractional leakage resilience approaches $1/4$, which is optimal [27].

References

1. Alon, N., Goldreich, O., Håstad, J., Peralta, R.: Simple constructions of almost k -wise independent random variables. In: 31st FOCS, pp. 544–553. IEEE Computer Society Press, October 1990
2. Alon, N., Roichman, Y.: Random cayley graphs and expanders. *Random Struct. Algorithms* **5**(2), 271–284 (1994)
3. Applebaum, B.: Garbled circuits as randomized encodings of functions: a primer. *Cryptology ePrint Archive*, Report 2017/385 (2017). <http://eprint.iacr.org/2017/385>
4. Applebaum, B., Damgård, I., Ishai, Y., Nielsen, M., Zichron, L.: Secure arithmetic computation with constant computational overhead. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017. LNCS, vol. 10401, pp. 223–254. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63688-7_8
5. Applebaum, B., Ishai, Y., Kushilevitz, E.: Cryptography in NC^0 . In: 45th FOCS, pp. 166–175. IEEE Computer Society Press, October 2004
6. Beaver, D.: Efficient multiparty protocols using circuit randomization. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 420–432. Springer, Heidelberg (1992). https://doi.org/10.1007/3-540-46766-1_34
7. Ben-David, A., Nisan, N., Pinkas, B.: FairplayMP: a system for secure multi-party computation. In: Ning, P., Syverson, P.F., Jha, S. (eds.) ACM CCS 2008, pp. 257–266. ACM Press, October 2008
8. Block, A.R., Gupta, D., Maji, H.K., Nguyen, H.H.: Secure computation using leaky correlations (asymptotically optimal constructions). *Cryptology ePrint Archive*, Report 2018/372 (2018). <https://eprint.iacr.org/2018/372>
9. Block, A.R., Maji, H.K., Nguyen, H.H.: Secure computation based on leaky correlations: high resilience setting. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017. LNCS, vol. 10402, pp. 3–32. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63715-0_1
10. Canetti, R.: Security and composition of multiparty cryptographic protocols. *J. Cryptology* **13**(1), 143–202 (2000)
11. Cascudo, I., Damgård, I., Farràs, O., Ranellucci, S.: Resource-efficient OT combiners with active security. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017. LNCS, vol. 10678, pp. 461–486. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70503-3_15
12. Cenk, M., Özbudak, F.: On multiplication in finite fields. *J. Complex.* **26**(2), 172–186 (2010)
13. Chen, H., Cramer, R.: Algebraic geometric secret sharing schemes and secure multi-party computations over small fields. In: Dwork, C. (ed.) CRYPTO 2006. LNCS, vol. 4117, pp. 521–536. Springer, Heidelberg (2006). https://doi.org/10.1007/11818175_31

14. Chudnovsky, D.V., Chudnovsky, G.V.: Algebraic complexities and algebraic curves over finite fields. *Proc. Natl. Acad. Sci.* **84**(7), 1739–1743 (1987)
15. Damgård, I., Pastro, V., Smart, N., Zakarias, S.: Multiparty computation from somewhat homomorphic encryption. In: Safavi-Naini, R., Canetti, R. (eds.) *CRYPTO 2012*. LNCS, vol. 7417, pp. 643–662. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32009-5_38
16. 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>
17. Dodis, Y., Smith, A.: Correcting errors without leaking partial information. In: Gabow, H.N., Fagin, R. (eds.) *37th ACM STOC*, pp. 654–663. ACM Press, May 2005
18. Döttling, N., Ghosh, S., Nielsen, J.B., Nilges, T., Trifiletti, R.: TinyOLE: Efficient actively secure two-party computation from oblivious linear function evaluation. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) *ACM CCS 17*, pp. 2263–2276. ACM Press, October/November 2017
19. Garcia, A., Stichtenoth, H.: On the asymptotic behaviour of some towers of function fields over finite fields. *J. Number Theory* **61**(2), 248–273 (1996)
20. 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). [https://doi.org/10.1002/\(SICI\)1098-2418\(199712\)11:4<315::AID-RSA3>3.0.CO;2-1](https://doi.org/10.1002/(SICI)1098-2418(199712)11:4<315::AID-RSA3>3.0.CO;2-1)
21. Goppa, V.D.: Codes on algebraic curves. In: *Soviet Math. Dokl.*, pp. 170–172 (1981)
22. Gupta, D., Ishai, Y., Maji, H.K., Sahai, A.: Secure computation from leaky correlated randomness. In: Gennaro, R., Robshaw, M. (eds.) *CRYPTO 2015*. LNCS, vol. 9216, pp. 701–720. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-48000-7_34
23. 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). https://doi.org/10.1007/978-3-540-78524-8_22
24. 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). https://doi.org/10.1007/11426639_6
25. Ishai, Y., Kushilevitz, E.: Perfect constant-round secure computation via perfect randomizing polynomials. In: Widmayer, P., Ruiz, F.T., Bueno, R.M., Hennessy, M., Eidenbenz, S., Conejo, R. (eds.) *ICALP 2002*. LNCS, vol. 2380, pp. 244–256. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45465-9_22
26. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Extracting correlations. In: *50th FOCS*, pp. 261–270. IEEE Computer Society Press, October 2009
27. Ishai, Y., Maji, H.K., Sahai, A., Wullschleger, J.: Single-use OT combiners with near-optimal resilience. In: *2014 IEEE International Symposium on Information Theory*, Honolulu, HI, USA, 29 June–4 July 2014, pp. 1544–1548 (2014)
28. 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). https://doi.org/10.1007/978-3-540-85174-5_32
29. Keller, M., Orsini, E., Scholl, P.: MASCOT: Faster malicious arithmetic secure computation with oblivious transfer. In: Weippl, E.R., Katzenbeisser, S., Kruegel, C., Myers, A.C., Halevi, S. (eds.) *ACM CCS 2016*, pp. 830–842. ACM Press, October 2016

30. Kilian, J.: A general completeness theorem for two-party games. In: 23rd ACM STOC, pp. 553–560. ACM Press, May 1991
31. Malkhi, D., Nisan, N., Pinkas, B., Sella, Y.: Fairplay - secure two-party computation system. In: Proceedings of the 13th USENIX Security Symposium, 9–13 August 2004, San Diego, pp. 287–302 (2004)
32. Massey, J.L.: Some applications of coding theory in cryptography. In: Codes and Ciphers: Cryptography and Coding IV, pp. 33–47 (1995)
33. 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). https://doi.org/10.1007/11818175_33
34. 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). https://doi.org/10.1007/978-3-540-70936-7_22
35. Naor, J., Naor, M.: Small-bias probability spaces: Efficient constructions and applications. In: 22nd ACM STOC, pp. 213–223. ACM Press, May 1990
36. Naor, M., Pinkas, B.: Computationally secure oblivious transfer. *J. Cryptology* **18**(1), 1–35 (2005)
37. 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). https://doi.org/10.1007/978-3-642-32009-5_40
38. Pless, V.: Introduction to the Theory of Error-Correcting Codes, vol. 48. Wiley (2011)
39. Przydatek, B., Wullschleger, J.: Error-tolerant combiners for oblivious primitives. In: Aceto, L., Damgård, I., Goldberg, L.A., Halldórsson, M.M., Ingólfssdóttir, A., Walukiewicz, I. (eds.) ICALP 2008. LNCS, vol. 5126, pp. 461–472. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70583-3_38
40. Rao, A.: An exposition of bourgain’s 2-source extractor. ECCCTR: Electronic Colloquium on Computational Complexity Technical reports (2007)
41. Vladut, S., Nogin, D., Tsfasman, M.: Algebraic Geometric Codes: Basic Notions. American Mathematical Society, Boston (2007)
42. Wolf, S., Wullschleger, J.: Oblivious transfer is symmetric. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 222–232. Springer, Heidelberg (2006). https://doi.org/10.1007/11761679_14