Algebraic Restriction Codes and Their Applications

Divesh Aggarwal □

National University of Singapore, Singapore

Nico Döttling **□ 0**

Helmholtz Center for Information Security (CISPA), Saarbrücken, Germany

Jesko Dujmovic ⊠

Helmholtz Center for Information Security (CISPA), Saarbrücken, Germany Saarland University, Saarbrücken, Germany

Mohammad Hajiabadi ⊠

University of Waterloo, ON, Canada

Giulio Malavolta ⊠

Max Planck Institute for Security and Privacy, Bochum, Germany

Maciej Obremski ⊠

National University of Singapore, Singapore

— Abstract

Consider the following problem: You have a device that is supposed to compute a linear combination of its inputs, which are taken from some finite field. However, the device may be faulty and compute arbitrary functions of its inputs. Is it possible to encode the inputs in such a way that only linear functions can be evaluated over the encodings? I.e., learning an arbitrary function of the encodings will not reveal more information about the inputs than a linear combination.

In this work, we introduce the notion of algebraic restriction codes (AR codes), which constrain adversaries who might compute any function to computing a linear function. Our main result is an information-theoretic construction AR codes that restrict any class of function with a bounded number of output bits to linear functions. Our construction relies on a seed which is not provided to the adversary.

While interesting and natural on its own, we show an application of this notion in cryptography. In particular, we show that AR codes lead to the first construction of rate-1 oblivious transfer with statistical sender security from the Decisional Diffie-Hellman assumption, and the first-ever construction that makes black-box use of cryptography. Previously, such protocols were known only from the LWE assumption, using non-black-box cryptographic techniques. We expect our new notion of AR codes to find further applications, e.g., in the context of non-malleability, in the future.

2012 ACM Subject Classification Security and privacy \rightarrow Information-theoretic techniques; Security and privacy \rightarrow Public key (asymmetric) techniques

Keywords and phrases Algebraic Restriction Codes, Oblivious Transfer, Rate 1, Statistically Sender Private, OT, Diffie-Hellman, DDH

Digital Object Identifier 10.4230/LIPIcs.ITCS.2022.2

 $\textbf{Related Version}: Full \ \textit{Version}: \texttt{https://eprint.iacr.org/2021/1177}$

Funding Divesh Aggarwal: This work is supported in part by the Singapore National Research Foundation under NRF RF Award No. NRF-NRFF2013-13, the Ministry of Education, Singapore under the Research Centres of Excellence programme by the Tier-3 grant Grant "Random numbers from quantum processes" No. MOE2012-T3-1-009.

Nico Döttling: This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4).

Jesko Dujmovic: This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4).

 $\it Maciej~Obremski$: This author was supported by MOE2019-T2-1-145 Foundations of quantum-safe cryptography.



© Divesh Aggarwal, Nico Döttling, Jesko Dujmovic, Mohammad Hajiabadi, Giulio Malavolta, and Maciej Obremski;

licensed under Creative Commons License CC-BY 4.0

 $13\mathrm{th}$ Innovations in Theoretical Computer Science Conference (ITCS 2022).

Editor: Mark Braverman; Article No. 2; pp. 2:1–2:15 Leibniz International Proceedings in Informatics

LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In this work we consider leakage problems of the following kind: Assume we have a device which takes an input x and is supposed to compute a function f(x) from a certain class of legitimate functions \mathcal{F} . For concreteness, assume that the class \mathcal{F} consists of functions computing linear combinations, e.g., $f(x_1, x_2) = a_1x_1 + a_2x_2$. However, the device might be faulty and instead of computing f it might compute another function g. We want to find a way to encode x into an \hat{x} such that the following two properties hold:

- If the device correctly implements a linear function f, then we can efficiently decode the output y to f(x).
- If, on the other hand, the device implements a non-linear function g, then the output $g(\hat{x})$ does not reveal more information about x than f(x) for some linear function f.

First, note that this notion is trivially achievable if \mathcal{F} includes the identity function, or in fact any invertible function, as in this case we can simulate $g(\hat{x})$ from f(x) by first recovering x from f(x), encoding x to \hat{x} and finally evaluating g on \hat{x} . For this reason, in this work we will focus on function classes \mathcal{F} whose output-length is smaller than their input-length, such as the linear combination functions mentioned above. In general, we will allow both the encoding and decoding procedure to depend on a secret seed, which is not given to the evaluating device/adversary.

It is worthwhile comparing the type of security this notion provides to tamper-resilient primitives such as non-malleable codes (NM-codes) [14, 13, 1] and non-malleable extractors [11, 10, 26, 9]. Such notions are geared towards prohibiting tampering altogether. Moreover, a central aspect for security for such notions is that the decoder tries to detect if some tampering happened, and indeed the decoder plays a crucial role in modeling security of non-malleable codes. In contrast, AR codes do and should allow manipulation by benign functions from the class \mathcal{F} . Furthermore, we only require a decoder for correctness purposes, whereas security is defined independently of the decoder.

One motivation to study the above problem comes from cryptography, specifically secure computation, where this is in fact a natural scenario. Indeed, a typical blueprint for secure two-party computation [32] in two rounds proceeds as follows: One party, called the receiver, encrypts his input y under a homomorphic encryption scheme [29, 17, 8, 19] obtaining a ciphertext c, and sends both the public key pk and the ciphertext c to the other party, called the sender. The sender, in possession of an input x homomorphically performs a computation f on input x and ciphertext c, obtaining a ciphertext c' which encrypts f(x,y). The ciphertext c' is sent back to the receiver who can then decrypt it to f(x,y).

For the case of a malicious receiver, the security of this blueprint breaks down completely: A malicious receiver can choose both the public key pk and the ciphertext c maliciously, i.e. they are generally not well-formed. Effectively, this means that the sender's homomorphic evaluation will result in some value $\tilde{f}(x)$ (where \tilde{f} will be specified by the receiver's protocol message) instead of an encryption of f(x,y). Critically, the value $\tilde{f}(x)$ might reveal substantially more information about x than f(x,y) and compromise the sender's security.

Generally speaking, in this situation there is no direct way for the sender to enforce which information about x the receiver obtains. A typical cryptographic solution for achieving malicious security involves using zero-knowledge proofs to enforce honest behavior for the receiver. This technique, however, is typically undesirable as it often leads to less efficient protocols (due to these tools using non-black-box techniques) and the need for several rounds of interaction or a trusted setup. Our goal is to upgrade such protocols to achieve security against malicious receivers without additional cryptographic machinery.

To see how algebraic restriction codes will help in this scenario, consider the following. Upon receiving a public key pk and a ciphertext c from the receiver (who potentially generated them in a malicious way) the sender proceeds as follows. First, he encodes his own input x into \hat{x} using a suitable AR code with a fresh seed s. Next, also then sender evaluates the function $f(\hat{x},\cdot)$ homomorphically on the ciphertext c (which encrypts the receiver's input y), resulting in a ciphertext $c' = \text{Eval}(pk, f(\hat{x},\cdot), c)$. For simplicity's sake, assume that the sender now sends c' and the seed s back to the receiver, who decrypts c' to $\hat{z} = f(\hat{x}, y)$ and uses the seed s to decode \hat{z} to his output s using the decoding algorithm of the AR code.

How can we argue that even a malicious receiver cannot learn more than the legitimate output z? Let's take a closer look on the computation which is actually performed on the encoding \hat{x} . The output ciphertext c' is computed via $c' = \mathsf{Eval}(\mathsf{pk}, f(\hat{x}, \cdot), c)$. Thus, if we can assure that the function $g(\hat{x}) = \mathsf{Eval}(\mathsf{pk}, f(\hat{x}, \cdot), c)$ is in the class $\mathcal G$ which is restricted by the AR code, then security of the AR code guarantees that c' does not leak more than z = f(x, y) about x, irrespective of the choice of pk and c.

1.1 Roadmap

In the following we will discuss our results in Subsection 1.2. Then, we will outline our techniques in Section 2 and discuss related work in Section 3. For full proofs and formal statements we refer to the full version [2].

1.2 Our Results

In this work, we formalize the notion of algebraic restriction codes and provide constructions which restrict general function classes to linear functions over finite fields. Let \mathcal{G} and \mathcal{F} be two function classes. Roughly, a \mathcal{G} - \mathcal{F} AR code provides a way to encode an x in the domain of the functions in \mathcal{F} into a codeword \hat{x} in the domain of the functions in \mathcal{G} , in a way that any function $f \in \mathcal{F}$ can still be evaluated on \hat{x} , by evaluating a function $f' \in \mathcal{G}$ on \hat{x} . Furthermore, given $f'(\hat{x})$ we can decode to f(x). Security-wise, we require that for any $g \in \mathcal{G}$ there exists a function $f \in \mathcal{F}$, such that $g(\hat{x})$ can be simulated given only the legitimate output f(x). AR codes provide an information-theoretic interface to limit the capabilities of an unbounded adversary in protocols in which some weak restrictions (characterized by the class \mathcal{G}) are already in place. In this way, AR codes will allow us to harness simple structural restrictions of protocols to implement very strong security guarantees.

In this work we consider *seeded* AR codes, where both the encoding and decoding procedures of the AR code have access to a random seed s, which is not provided to the function g.

Our first construction of AR-codes restricts general linear functions to linear combinations.

▶ Theorem 1 (Formal: Theorem 4, Page 14, Full Version [2]). Let \mathbb{F}_q be a finite field, let \mathcal{F} be the class of functions $\mathbb{F}_q^k \times \mathbb{F}_q^k \to \mathbb{F}_q^k$ of the form $(\mathbf{x}, \mathbf{y}) \mapsto a\mathbf{x} + \mathbf{y}$, and let \mathcal{G} be the class of all linear functions $\mathbb{F}_q^n \times \mathbb{F}_q^n \to \mathbb{F}_q^n$ of the form $(\mathbf{x}, \mathbf{y}) \mapsto A\mathbf{x} + \mathbf{y}$. There exists a seeded AR code AR_1 which restricts \mathcal{G} to \mathcal{F} .

Our main contribution is a construction of seeded AR codes restricting arbitrary functions with bounded output length to linear functions.

▶ Theorem 2 (Formal: Theorem 5, Page 19, Full Version [2]). Let \mathbb{F}_q be a finite field, let \mathcal{F} be the class of functions $\mathbb{F}_q \times \mathbb{F}_q \to \mathbb{F}_q$ of the form $(x,y) \mapsto ax + by$, and let \mathcal{G} be the class of all functions $\mathbb{F}_q^n \times \mathbb{F}_q^n \to \{0,1\}^{1.9 \cdot n \log(q)}$. There exists a seeded AR code AR₂ which restricts \mathcal{G} to \mathcal{F} .

We note that the constant 1.9 in the Theorem is arbitrary and can in fact be replaced with any constant smaller than 2.

The main ingredient of this construction is the following theorem, which may be of independent interest and which we will discuss in some greater detail. The theorem exhibits a new correlation-breaking property of the inner-product extractor.

In essence, it states that for a suitable parameter choice, if $\mathbf{x}_1, \dots, \mathbf{x}_t$ are uniformly random vectors in a finite vector space and \mathbf{s} is a random seed (in the same vector space), then anything that can be inferred about the $\langle \mathbf{x}_1, \mathbf{s} \rangle, \dots, \langle \mathbf{x}_t, \mathbf{s} \rangle$ via a *joint leak* $f(\mathbf{x}_1, \dots, \mathbf{x}_t)$ of bounded length can also be inferred from a linear combination $\sum_i a_i \langle \mathbf{x}_i, \mathbf{s} \rangle$, i.e. $f(\mathbf{x}_1, \dots, \mathbf{x}_t)$ does not leak more than $\sum_i a_i \langle \mathbf{x}_i, \mathbf{s} \rangle$.

▶ Theorem 3 (Formal: Theorem 5, Page 19, Full Version [2]). Let q be a prime power, let t, s be positive integers, and $\varepsilon > 0$ and $n = O(t + s/\log(q) + (\log \frac{1}{\varepsilon})/\log(q))$. Let $\mathbf{x}_1, \ldots, \mathbf{x}_t$ be uniform in \mathbb{F}_q^n and \mathbf{s} is uniform in \mathbb{F}_q^n and independent of the \mathbf{x}_i . For any $f: \mathbb{F}_q^{tn} \to \{0,1\}^{n \log q + s}$, there exists a simulator Sim and random variables $a_1, \ldots, a_t \in \mathbb{F}_q$ such that

$$\begin{split} \mathbf{s}, f(\mathbf{x}_1, \dots, \mathbf{x}_t), \langle \mathbf{x}_1, \mathbf{s} \rangle, \dots, \langle \mathbf{x}_t, \mathbf{s} \rangle, a_1, \dots, a_t \\ \approx_{2\varepsilon} \mathrm{Sim} \left(\mathbf{s}, a_1, \dots, a_t, \sum_{i=1}^t a_i u_i \right), u_1, \dots, u_t, a_1, \dots, a_t \end{split}$$

where u_1, \ldots, u_t are uniform and independent random variables in \mathbb{F}_q , independent of (a_1, \ldots, a_t) .

One way to interpret the theorem is that the inner product extractor breaks all correlations (induced by a leak $f(\mathbf{x}_1, \dots, \mathbf{x}_t)$), except linear ones. Recall that our notion of AR codes it is crucial that linear relations are preserved.

We then demonstrate an application of AR codes in upgrading the security of oblivious transfer (OT) protocols while simultaneously achieving optimal communication, a question that had remained opened due to insurmountable difficulties, explained later. Specifically, we obtain the first rate-1 OT protocol with statistical sender privacy from the decisional Diffie Hellman (DDH) assumption. While our motivation to study AR codes is to construct efficient and high rate statistically sender private OT protocols, we expect AR codes and in particular the ideas used to construct them to be useful in a broader sense.

2 Technical Outline

In what follows, we provide an informal overview of the techniques developed in this work.

2.1 Warmup: Algebraic Restriction Codes for General Linear Functions

Before discussing the ideas leading up to our main result, we will first discuss the instructive case of AR codes restricting general linear functions to *simple* linear functions. Specifically, fix a finite field \mathbb{F}_q and let \mathcal{G} be the class of linear functions $\mathbb{F}_q^{2m} \to \mathbb{F}_q^m$ of the form $g(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2) = \mathbf{A}\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$, where $\mathbf{A} \in \mathbb{F}_q^{m \times m}$ is an arbitrary matrix. We want to restrict \mathcal{G} to the class \mathcal{F} consisting of linear functions $\mathbb{F}_q^{2n} \to \mathbb{F}^n$ of the form $f(\mathbf{x}_1, \mathbf{x}_2) = a \cdot \mathbf{x}_1 + \mathbf{x}_2$, where $a \in \mathbb{F}_q$ is a scalar.

Our construction proceeds as follows. The seed s specifies a random matrix $\mathbf{R} \in \mathbb{F}_q^{n \times m}$, such a matrix has full rank except with probability $\leq 2^{-(m-n)}$. To encode a pair of input vectors $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{F}_q^n$, the encoder samples uniformly random $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2 \stackrel{\$}{\leftarrow} \mathbb{F}_q^m$ such that $\mathbf{R}\hat{\mathbf{x}}_1 = \mathbf{x}_1$

and $\mathbf{R}\hat{\mathbf{x}}_2 = \mathbf{x}_2$, and outputs the codeword $(\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2)$. To evaluate a scalar linear function given by $a \in \mathbb{F}_q$ on such a codeword, we (unsurprisingly) compute $\hat{\mathbf{y}} = a\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$. To decode $\hat{\mathbf{y}}$ we compute $\mathbf{y} = \mathbf{R}\hat{\mathbf{y}}$. Correctness of this AR code construction follows routinely:

$$\mathbf{y} = \mathbf{R}\hat{\mathbf{y}} = \mathbf{R}(a\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2) = \mathbf{R}a\hat{\mathbf{x}}_1 + \mathbf{R}\hat{\mathbf{x}}_2 = a\mathbf{R}\hat{\mathbf{x}} + \mathbf{R}\hat{\mathbf{x}}_2 = a\mathbf{x}_1 + \mathbf{x}_2.$$

i.e. correctness holds as the scalar a commutes with the matrix \mathbf{R} .

In this case it will also be more convenient to look at the problem from the angle of randomness extraction; Specifically, assume that $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2 \stackrel{\$}{\leftarrow} \mathbb{F}_q^m$ are chosen uniformly random. We want to show that for any matrix $\mathbf{A} \in \mathbb{F}_q^{m \times m}$ anything that can be learned about $\mathbf{R}\hat{\mathbf{x}}_1$ and $\mathbf{R}\hat{\mathbf{x}}_2$ from $\mathbf{A}\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$ can also be learned from $a \cdot \mathbf{R}\hat{\mathbf{x}}_1 + \mathbf{R}\hat{\mathbf{x}}_2$ for some $a \in \mathbb{F}_q$.

How can we find such an a for any given \mathbf{A} ? First notice that if $\hat{\mathbf{x}}_1$ happens to be an eigenvector of \mathbf{A} with respect to an eigenvalue a_i , then it indeed holds that $\mathbf{A}\mathbf{x}_1 + \mathbf{x}_2 = a_i\mathbf{x}_1 + \mathbf{x}_2$. Thus, a reasonable approach is to set the extracted scalar $a \in \mathbb{F}_q$ to one of the eigenvalues of \mathbf{A} (or 0 if there are no eigenvalues). If the matrix \mathbf{A} has several distinct eigenvalues a_i , we will set a to be the eigenvalue whose eigenspace V_i has maximal dimension. Note that since the sum of the dimensions of all eigenspaces of \mathbf{A} is at most n, there can be at most one eigenspace whose dimension is larger than m/2. Furthermore, the eigenvalue a_i corresponding to this eigenspace will necessarily be the extracted value a.

Rather than showing how we can simulate $\hat{\mathbf{y}} = \mathbf{A}\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$ in general, in this sketch we will only briefly argue the following special case. Namely, if all the eigenspaces of \mathbf{A} have dimension smaller than or equal to m/2, then with high probability over the choice of the random matrix $\mathbf{R} \stackrel{\$}{\leftarrow} \mathbb{F}_q^{n \times m}$ it holds that $\mathbf{x}_1 = \mathbf{R}\hat{\mathbf{x}}_1$ and $\mathbf{x}_2 = \mathbf{R}\hat{\mathbf{x}}_2$ are uniform and independent of $\hat{\mathbf{y}}$. Thus assume that $\hat{\mathbf{y}} = \mathbf{A}\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$ was not independent of $\mathbf{x}_1 = \mathbf{R}\hat{\mathbf{x}}_1$ and $\mathbf{x}_2 = \mathbf{R}\hat{\mathbf{x}}_2$. Since these three variables are linear functions of the uniformly random $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ there must exist a non-zero linear relation given by vectors $\mathbf{u}, \mathbf{v} \in \mathbb{F}_q^n$ and $\mathbf{w} \in \mathbb{F}_q^m$ such that $\mathbf{u}^{\top}\mathbf{x}_1 + \mathbf{v}^{\top}\mathbf{x}_2 + \mathbf{w}^{\top}\hat{\mathbf{y}} = 0$ for all choices of $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$. But this means that it holds that $\mathbf{u}^{\top}\mathbf{R} + \mathbf{w}^{\top}\mathbf{A} = 0$ and $\mathbf{v}^{\top}\mathbf{R} + \mathbf{w}^{\top} = 0$. Eliminating \mathbf{w}^{\top} , this simplifies to the equation $\mathbf{u}^{\top}\mathbf{R} = \mathbf{v}^{\top}\mathbf{R}\mathbf{A}$.

We will now argue that for any such matrix $\mathbf{A} \in \mathbb{F}_q^{m \times m}$ (whose eigenspaces all have dimension $\leq m/2$) with high probability over the choice of the random matrix \mathbf{R} , such a relation given by $(\mathbf{u}, \mathbf{v}) \neq 0$ does not exist. We will take a union bound over all non-zero \mathbf{u}, \mathbf{v} and distinguish the following cases:

- If \mathbf{u} and \mathbf{v} are linearly independent, then $\mathbf{u}^{\top}\mathbf{R}$ and $\mathbf{v}^{\top}\mathbf{R}$ are uniformly random and independent (over the random choice of \mathbf{R}). Thus the probability that $\mathbf{u}^{\top}\mathbf{R}$ and $\mathbf{v}^{\top}\mathbf{R}\mathbf{A}$ collide is $1/q^m$.
- If **u** and **v** are linearly dependent, then (say) $\mathbf{u} = \alpha \mathbf{v}$. In this case $\mathbf{u}^{\top} \mathbf{R} = \mathbf{v}^{\top} \mathbf{R} \mathbf{A}$ is equivalent to $\alpha \mathbf{v}^{\top} \mathbf{R} = \mathbf{v}^{\top} \mathbf{R} \mathbf{A}$, i.e. the uniformly random vector $\mathbf{v}^{\top} \mathbf{R}$ is an eigenvector of the matrix **A** with respect to the eigenvalue α . However, since all eigenspaces of **A** have dimension at most m/2, the probability that $\mathbf{v}^{\top} \mathbf{R}$ lands in one of the eigenspaces bounded by $m/q^{m/2}$.

Since there are q^{2n} possible choices for the vectors $\mathbf{u}, \mathbf{v} \in \mathbb{F}_q^n$, choosing m sufficiently large (e.g. m > 5n) implies that the probability that such $\mathbf{u}, \mathbf{v} \in \mathbb{F}_q^n$ exist is negligible. The full proof is provided in Section 6 in the full version [2].

2.2 Algebraic Restriction Codes for Bounded Output Functions

We will now turn to algebraic restriction codes for arbitrary functions with bounded output length. Now let \mathbb{F}_q be the finite field of size q, let \mathcal{G} be the class of all functions from $\mathbb{F}_q^{2n} \to \{0,1\}^{1.5n\log(q)}$ and let \mathcal{F} be the class of linear functions $\mathbb{F}_q^2 \to \mathbb{F}_q$, i.e. al; functions

of the form $f(x_1, x_2) = a_1x_1 + a_2x_2$ for some $a_1, a_2 \in \mathbb{F}_q$. Our AR code construction follows naturally from the inner product extractor. The seed s consists of a random vector $\mathbf{s} \stackrel{\$}{\leftarrow} \mathbb{F}_q^n$, to encode $x_1, x_2 \in \mathbb{F}_q$ we choose uniformly random $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{F}_q^n$ with $\langle \mathbf{x}_1, \mathbf{s} \rangle = x_1$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle = x_2$. Likewise, to decode a value \mathbf{y} we compute $y = \langle \mathbf{y}, \mathbf{s} \rangle$, correctness follows immediately as above. To show that this construction restricts \mathcal{G} to \mathcal{F} , we will again take the extractor perspective. Thus, assume that $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{F}_q^n$ are distributed uniformly random and let $g : \mathbb{F}_q^n \times \mathbb{F}_q^n \to \{0, 1\}^{1.5n \log(p)}$ be an arbitrary function.

We need to argue that for any $g \in \mathcal{G}$ there exist exist $a_1, a_2 \in \mathbb{F}_q$ such that $g(\mathbf{x}_1, \mathbf{x}_2)$ can be simulated given $y = a_1 \langle \mathbf{x}_1, \mathbf{s} \rangle + a_2 \langle \mathbf{x}_2, \mathbf{s} \rangle$, but no further information about $\langle \mathbf{x}_1, \mathbf{s} \rangle$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle$. Our analysis distinguishes two cases.

- In the first case, both $\langle \mathbf{x}_1, \mathbf{s} \rangle$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle$ are statistically close to uniform given $g(\mathbf{x}_1, \mathbf{x}_2)$. In other words, it directly holds that $g(\mathbf{x}_1, \mathbf{x}_2)$ contains no information about $\langle \mathbf{x}_1, \mathbf{s} \rangle$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle$. We can simulate $g(\mathbf{x}_1, \mathbf{x}_2)$ by choosing two independent \mathbf{x}'_1 and \mathbf{x}'_2 and computing $g(\mathbf{x}'_1, \mathbf{x}'_2)$.
- In the second case $\langle \mathbf{x}_1, \mathbf{s} \rangle$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle$ are (jointly) statistically far from uniform given $g(\mathbf{x}_1, \mathbf{x}_2)$. In this case we will rely on a variant of the XOR Lemma [31] to conclude that there must exist $a_1, a_2 \in \mathbb{F}_q$ such that $a_1x_1 + a_2x_2$ is also far from uniform given $g(\mathbf{x}_1, \mathbf{x}_2)$. Roughly, the XOR Lemma states that if it holds for two (correlated) random variables z_1, z_2 that for all $a_1, a_2 \in \mathbb{F}_q$ (such that one of them is non-zero) that $a_1z_1 + a_2z_2$ are statistically close to uniform, then (z_1, z_2) must be statistically close to uniform in \mathbb{F}_q^2 . Consequently, the existence of such $a_1, a_2 \in \mathbb{F}_q$ in our setting follows directly from the contrapositive of the XOR Lemma. But this implies that $a_1\mathbf{x}_1 + a_2\mathbf{x}_2$ must have very low min-entropy given $g(\mathbf{x}_1, \mathbf{x}_2)$. Otherwise, the leftover hash lemma would imply that $a_1x_1 + a_2x_2 = \langle a_1\mathbf{x}_1 + a_2\mathbf{x}_2, \mathbf{s} \rangle$ is close to uniform given $g(\mathbf{x}_1, \mathbf{x}_2)$, in contradiction to the conclusion above. But this means that $a_1\mathbf{x}_1 + a_2\mathbf{x}_2$ is essentially fully specified by $g(\mathbf{x}_1, \mathbf{x}_2)$. In other words $g(\mathbf{x}_1, \mathbf{x}_2)$ carries essentially the entire information about $a_1\mathbf{x}_1 + a_2\mathbf{x}_2$. But now recall that the bit size of $g(\mathbf{x}_1, \mathbf{x}_2)$ is $1.5n\log(q)$ bits and the bit size of $a_1\mathbf{x}_1 + a_2\mathbf{x}_2$ is $n\log(q)$ bits. Thus, there is essentially not enough room in $g(\mathbf{x}_1, \mathbf{x}_2)$ to carry significant further information about \mathbf{x}_1 or \mathbf{x}_2 . Again relying on the leftover hash lemma, we then conclude that given $g(\mathbf{x}_1, \mathbf{x}_2)$, $\langle \mathbf{x}_1, \mathbf{s} \rangle$ and $\langle \mathbf{x}_2, \mathbf{s} \rangle$ are statistically close to uniform subject to $a_1\langle \mathbf{x}_1, \mathbf{s} \rangle + a_2\langle \mathbf{x}_2, \mathbf{s} \rangle = y$.

While this sketch captures the very high level ideas of our proof, the actual proof needs to overcome some additional technical challenges and relies on a careful partitioning argument. The proof can be found in Section 7 in the full version [2].

2.3 From AR Codes to Efficient Oblivious Transfer

We display the usefulness of AR codes in cryptography by constructing a new oblivious transfer (OT) [30, 15] protocol. OT is a protocol between two parties, a sender, who has a pair of messages (m_0, m_1) , and a receiver who has a bit b, where at the end, the receiver learns m_b , while the sender should learn nothing. OT is a central primitive of study in the field of secure computation: Any multiparty functionality can be securely computed given a secure OT protocol [33, 24]. In particular, statistically-sender private (SSP) [27, 3] 2-message OT has recently received a lot of attention due to its wide array of applications, such as statistical ZAPs [4, 21] and maliciously circuit-private homomorphic encryption [28]. While the standard security definitions for OT are simulation-based (via efficient simulators), SSP OT settles for a weaker indistinguishability-based security notion for the receiver and an inefficient simulation notion for the sender. On the other hand, SSP OT can be realized in

just two messages, without a setup and from standard assumptions, a regime in which no OT protocols with simulation-based security are known¹. In this work, we obtain the first OT protocol that simultaneously satisfies the following properties:

- (1) It is round-optimal (2 messages) and it does not assume a trusted setup.
- (2) It satisfies the notion of *statistical* sender privacy (and computational receiver privacy). That is, a receiver who may (potentially) choose her first round message maliciously will be statistically oblivious to at least one of the two messages of the sender.
- (3) It achieves optimal rate for information transfer (i.e., it is rate-1).
- (4) It makes only black-box use of cryptographic primitives, in the sense that our protocol does not depend on circuit-level implementations of the underlying primitives.

Prior to our work, we did not know any OT protocol that simultaneously satisfied all of the above properties from any assumption. The only previous construction was based on LWE (using expensive fully-homomorphic encryption techniques), which only satisfies the first three conditions, but not the last one. (See Section 3.) We obtain constructions that satisfy all the above conditions from DDH/LWE. Optimal-rate OT is an indispensable tool in relazing various MPC functionalities with sublinear communication [23]. As direct corollaries, we obtain two-message maliciously secure protocols for keyword search [23] and symmetric private information retrieval (PIR) protocols [25] with statistical server privacy and with asymptotically optimal communication complexity from DDH/LWE. Our scheme is the first that makes only black-box use of cryptography, which we view as an important step towards the practical applicability of these protocols.

Packed ElGamal

Before delving into the description of our scheme, we recall the *vectorized* variant of the ElGamal encryption scheme [16]. Let \mathbb{G} be an Abelian group of prime order p and let g be a generator of \mathbb{G} . In the packed ElGamal scheme, a public key pk consists of a vector $\mathbf{h} = (h_1, \ldots, h_n) \in \mathbb{G}^n$ where $h_i = g^{x_i}$ for random $x_i \stackrel{\$}{\leftarrow} \mathbb{Z}_p$. The secret $\mathbf{s}k$ is the vector $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n$. To encrypt a $\mathbf{m} = (m_1, \ldots, m_n) \in \{0, 1\}^n$, we choose a uniformly random $r \stackrel{\$}{\leftarrow} \mathbb{Z}_p$ and set the ciphertext \mathbf{c} to

$$\mathbf{c} = (d_0, \mathbf{d}) = (q^r, \mathbf{h}^r \cdot q^{\mathbf{m}})$$

where both exponentiations and group operations of vectors are component-wise. We call d_0 the header of the ciphertext and $\mathbf{d} = (d_1, \dots, d_n)$ the payload of \mathbf{c} , we further call d_1, \dots, d_n the slots. To decrypt a ciphertext \mathbf{c} , we compute $\mathbf{m} = \mathsf{dlog}_g(d_0^{-\mathbf{x}} \cdot \mathbf{d})$. If we disregard the need for efficient decryption, we can encrypt arbitrary \mathbb{Z}_p^n vectors rather than just binary vectors. For such full range plaintexts the rate of packed ElGamal, i.e. the ratio between plaintext size and ciphertext size comes down to $(1-1/(n+1))\log(p)/\lambda$, assuming a group element can be described using λ bits. If $\lambda \approx \log(p)$, as is the case for dense groups, the rate approaches 1, for sufficiently large n. Finally, for a matrix $\mathbf{X} \in \{0,1\}^{n \times k}$, we encrypt \mathbf{X} column-wise, to obtain a ciphertext-matrix \mathbf{C} .

¹ In fact, it can be shown that any simulator for such a protocol would need to make non-black-box use of the adversary, as it would immediately imply a two-message zero-knowledge protocol, which was shown black-box impossible in [20]

Homomorphism and Ciphertext Compression

Packed ElGamal supports two types of homomorphism. It is linearly homomorphic with respect to \mathbb{Z}_p -linear combinations. Namely, if \mathbf{c} is an encryption of a vector $\mathbf{m} \in \mathbb{Z}_p^n$ and \mathbf{c}' is an encryption of a vector $\mathbf{m}' \in \mathbb{Z}_p^n$, then for any $\alpha, \beta \in \mathbb{Z}_p$ it holds that $\mathbf{c}'' = \mathbf{c}^{\alpha} \cdot \mathbf{c}'^{\beta}$ is a well-formed encryption of $\alpha \mathbf{m} + \beta \mathbf{m}'$ (again, disregarding the need for efficient decryption for large plaintexts). This routinely generalizes to arbitrary linear combinations, namely we can define a homomorphic evaluation algorithm \mathbf{Eval}_1 which takes as input a public key \mathbf{pk} , a ciphertext matrix \mathbf{C} encrypting a matrix $\mathbf{X} \in \mathbb{Z}_p^{n \times m}$, and two vectors $\mathbf{a} \in \mathbb{Z}_p^m$ and $\mathbf{b} \in \mathbb{Z}_p^n$ and outputs an encryption of $\mathbf{Xa} + \mathbf{b}$. By re-randomizing the resulting ciphertext this can be made function private, i.e. the output ciphertext leaks nothing beyond $\mathbf{Xa} + \mathbf{b}$ about \mathbf{a} and \mathbf{b} .

The second type of homomorphism supported by packed ElGamal is a limited type of homomorphism across the slots. Specifically, let $\mathbf{c} = (d_0, \mathbf{d})$ be an encryption of a message $\mathbf{m} \in \mathbb{Z}_p^n$ and let $\mathbf{M} \in \mathbb{Z}_p^{m \times n}$ be a matrix. Then there is a homomorphic evaluation algorithm Eval_2 which takes the public key pk , the ciphertext \mathbf{c} and a matrix $\mathbf{M} \in \mathbb{Z}_p^{m \times n}$ and outputs a ciphertext \mathbf{c}' , such that \mathbf{c}' encrypts the message $\mathbf{m}' = \mathbf{Mm}$ under a modified public key $\mathsf{pk}' = g^{\mathbf{Mx}}$. Furthermore, if the decrypter knows the matrix \mathbf{M} , it can derive the modified secret $\mathsf{sk}' = \mathbf{Mx}$ and decrypt \mathbf{c}' to \mathbf{m}' (given that $\mathbf{m}' \in \{0,1\}^m$).

Finally, the packed ElGamal scheme supports ciphertext compression for bit encryptions [12]. There is an efficient algorithm Shrink which takes a ciphertext $\mathbf{c} = (d_0, \mathbf{d})$ and produces a compressed ciphertext $\tilde{\mathbf{c}} = (d_0, K, \mathbf{b})$, where K is a (short) key and $\mathbf{b} \in \{0, 1\}^n$ is a binary vector. Consequently, compressed ciphertexts are of size $n + \mathsf{poly}$ bits and therefore have rate $1 - \mathsf{poly}/n$, which approaches 1 for a sufficiently large n (independent of the description size of group elements). Such compressed ciphertexts can then be decrypted using a special algorithm ShrinkDec, using the same secret key sk. Compressed ciphertexts generally do not support any further homomorphic operations, so ciphertext compression is performed after all homomorphic operations.

Semi-Honest Rate-1 OT from Packed ElGamal

The packed ElGamal encryption scheme with ciphertext compression immediately gives rise to a *semi-honestly secure* OT protocol with download rate 1. Specifically, the receiver whose choice-bit is b generates a key-pair pk, sk, encrypts the matrix $b \cdot \mathbf{I}$ to a ciphertext matrix \mathbf{C} , and sends $ot_1 = (pk, \mathbf{C})$ to the sender. The sender, whose input are two strings \mathbf{m}_0 and $\mathbf{m}_1 \in \{0, 1\}^n$ uses Eval_1 to homomorphically evaluate the function

$$f(\mathbf{X}) = \mathbf{X}(\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0$$

on the ciphertext \mathbf{C} , obtaining a ciphertext \mathbf{c} . It then compresses the ciphertext \mathbf{c} to a compressed ciphertext $\tilde{\mathbf{c}}$ and sends $ot_1 = \tilde{\mathbf{c}}$ back to the receiver who can decrypt it to a value \mathbf{m}' using the ShrinkDec algorithm. By homomorphic correctness it holds that $b \cdot \mathbf{I} \cdot (\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0 = \mathbf{m}_b$.

However, note that the sender privacy of this protocol completely breaks down against malicious receivers. Specifically, a malicious receiver is not bound to encrypting the scalar matrix $b \cdot \mathbf{I}$, but could instead encrypt an arbitrary matrix $\mathbf{A} \in \mathbb{Z}_p^{n \times n}$, thereby learning $\mathbf{A}(\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0$ instead of \mathbf{m}_b . By e.g. choosing

$$\mathbf{A} = \begin{pmatrix} 0 & 0 \\ 0 & \mathbf{I} \end{pmatrix}$$

the receiver could learn half of the bits of \mathbf{m}_0 and half of the bits of \mathbf{m}_1 , thus breaking sender privacy.

Malicious Security via AR Codes

Next we show how to make the above protocol statistically sender private against malicious receivers using AR codes. The protocol follows the same outline as above, except that the sender samples a seed $\bf R$ for an AR code and encodes its inputs

$$\hat{\mathbf{x}}_1 = \mathsf{Encode}(\mathbf{R}, \mathbf{m}_1 - \mathbf{m}_0) \text{ and } \hat{\mathbf{x}}_2 = \mathsf{Encode}(\mathbf{R}, \mathbf{m}_0).$$

Then it computes a ciphertext $\mathbf{c} = \mathsf{Eval}_1(\mathsf{pk}, \mathbf{C}, \hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2)$. If the sender were to transmit directly this ciphertext, the rate of the scheme would degrade (due to the size of the encodings) and the decryption would not be efficient, since \mathbf{c} contains an encoding $\hat{\mathbf{y}} \in \mathbb{Z}_p^m$. To deal with this issue, we observe that decoding $\hat{\mathbf{y}}$ to \mathbf{y} via $\mathbf{y} = \mathbf{R}\hat{\mathbf{y}}$ is exactly the type of operation supported by the homomorphic evaluation Eval_2 . Thus, we let the sender further compute $\mathbf{c}' = \mathsf{Eval}_2(\mathsf{pk}, \mathbf{c}, \mathbf{R})$. By homomorphic correctness of Eval_2 , it holds that \mathbf{c}' is an encryption of $\mathbf{R}\hat{\mathbf{y}} = \mathbf{y} = \mathbf{m}_b \in \{0,1\}^n$ under a modified public key pk' (which depends on \mathbf{R}). Since \mathbf{c}' encrypts a binary message, the sender can further use the ciphertext compression algorithm Shrink to shrink \mathbf{c}' into a rate-1 ciphertext $\tilde{\mathbf{c}}$. The sender now sends \mathbf{R} and $\tilde{\mathbf{c}}$ back to the receiver, who derives a key from sk and \mathbf{R} , and uses it to decrypt $\tilde{\mathbf{c}}$ via ShrinkDec.

If we were to do things naively, the protocol would still not achieve rate-1 since we have to also attach to the OT second message a potentially large matrix \mathbf{R} . This can be resolved via a standard trick: By reusing the same matrix \mathbf{R} in several parallel instances of the protocol, we can amortize the cost of sending the matrix \mathbf{R} . Note that \mathbf{R} can be reused as we only need to ensure that the matrix \mathbf{A} does not depend on \mathbf{R} . Thus, we have achieved a rate-1 protocol.

There is one subtle aspect that we need to address before declaring victory: The security of AR codes only guarantees that a malicious receiver may learn $a(\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0$ for some $a \in \mathbb{Z}_p$, rather than $b(\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0 = \mathbf{m}_b$ for $b \in \{0, 1\}$. To address this last issue, we let the sender compute $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ by

$$\hat{\mathbf{x}}_1 = \mathsf{Encode}(\mathbf{R}, \mathbf{x}_1)$$

$$\hat{\mathbf{x}}_2 = \mathsf{Encode}(\mathbf{R}, \mathbf{x}_2),$$

where
$$\mathbf{x}_1 = \begin{pmatrix} \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_0 \\ \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_1 \end{pmatrix}$$
 and $\mathbf{x}_2 = \begin{pmatrix} \mathbf{m}_0 \\ \mathbf{m}_0 - \mathbf{r}_1 \end{pmatrix}$ and $\mathbf{r}_0, \mathbf{r}_1$ are uniformly random.

Consequently, instead of $a(\mathbf{m}_1 - \mathbf{m}_0) + \mathbf{m}_0$ the ciphertext \mathbf{c} now encrypts

$$f(\mathbf{x}_1, \mathbf{x}_2) = a \cdot \begin{pmatrix} \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_0 \\ \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_1 \end{pmatrix} + \begin{pmatrix} \mathbf{m}_0 \\ \mathbf{m}_0 - \mathbf{r}_1 \end{pmatrix},$$

and by the security of the AR code **c** does not leak more information about \mathbf{x}_1 and \mathbf{x}_2 then $f(\mathbf{x}_1, \mathbf{x}_2)$. Now, note that if a = 0, then

$$f(\mathbf{x}_1, \mathbf{x}_2) = \begin{pmatrix} \mathbf{m}_0 \\ \mathbf{m}_0 - \mathbf{r}_1 \end{pmatrix},$$

where we note that $\mathbf{r}'_1 = \mathbf{m}_0 - \mathbf{r}_1$ is uniformly random. On the other hand, if a = 1, then

$$f(\mathbf{x}_1, \mathbf{x}_2) = \begin{pmatrix} \mathbf{m}_1 + \mathbf{r}_0 \\ \mathbf{m}_1 \end{pmatrix},$$

where we note that $\mathbf{r}'_0 = \mathbf{m}_1 + \mathbf{r}_0$ is uniformly random. Finally, if $a \notin \{0, 1\}$, then

$$f(\mathbf{x}_0, \mathbf{x}_1) = a \cdot \begin{pmatrix} \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_0 \\ \mathbf{m}_1 - \mathbf{m}_0 + \mathbf{r}_1 \end{pmatrix} + \begin{pmatrix} \mathbf{m}_0 \\ \mathbf{m}_0 - \mathbf{r}_1 \end{pmatrix} = \begin{pmatrix} a\mathbf{m}_1 + (1-a)\mathbf{m}_0 \\ a\mathbf{m}_1 + (1-a)\mathbf{m}_0 \end{pmatrix} + \begin{pmatrix} a \cdot \mathbf{r}_0 \\ (1-a) \cdot \mathbf{r}_1 \end{pmatrix},$$

which is uniformly random as the last term is uniformly random. I.e. if $a \notin \{0,1\}$ the receiver will learn nothing about \mathbf{m}_0 and \mathbf{m}_1 . Thus, we can conclude that even for a malformed public key pk and ciphertext \mathbf{C} the view of the receiver can be simulated given at most one \mathbf{m}_b , and statistical sender privacy follows.

Back to Rate-1

Note that now the ciphertext \mathbf{c} is twice as long as before, which again ruins the rate of our scheme. However, note that in order to get a correct scheme, if a=0 the receiver only needs to recover the first half \mathbf{z}_0 of the vector $f(\mathbf{x}_1, \mathbf{x}_2) = \begin{pmatrix} \mathbf{z}_0 \\ \mathbf{z}_1 \end{pmatrix}$, whereas if a=1 she needs the second part \mathbf{z}_1 . Our final idea is to facilitate this by additionally using a rate-1 OT protocol $\mathsf{OT}' = (\mathsf{OT}'_1, \mathsf{OT}'_2, \mathsf{OT}'_3)$ with semi-honest security (e.g. as given in [12]). We will further use the fact that the packed ElGamal ciphertext $\tilde{\mathbf{c}}$ can be written as $(h, \tilde{\mathbf{c}}_0, \tilde{\mathbf{c}}_1)$, where h is the ciphertext header, $\tilde{\mathbf{c}}_0$ is a rate-1 ciphertext encrypting \mathbf{z}_0 and $\tilde{\mathbf{c}}_1$ is a rate-1 ciphertext encrypts \mathbf{z}_1 (both with respect to the header h).

We modify the above protocol such that the receiver additionally includes a first message ot'_1 computed using his choice bit b. Instead of sending both $\tilde{\mathbf{c}}_0$ and $\tilde{\mathbf{c}}_1$ to the receiver (which would ruin the rate), we compute the sender message ot'_2 for OT' as $ot'_2 \leftarrow \mathsf{OT}_2(ot'_1, \tilde{\mathbf{c}}_0, \tilde{\mathbf{c}}_1)$ and send (h, ot'_2) to the receiver. The receiver can now recover $\tilde{\mathbf{c}}_b$ from ot'_2 and decrypt the ciphertext $(h, \tilde{\mathbf{c}}_b)$ as above. Note that now the communication rate from sender to receiver is 1. Note that we do not require any form of sender security from the rate-1 OT. Finally, note that as discussed above the protocol can be made overall rate-1 by amortizing for the size of the receiver's message (i.e. repeating the protocol in parallel for the same receiver message but independent blocks of the sender message).

Certified vs Uncertified Groups

We conclude this overview by discussing two variants of groups where we can implement the OT as specified above. In *certified groups*, we can assume that \mathbb{G} in fact implements a group of prime order p, even if maliciously chosen. In these settings, our simpler variant of AR codes suffices, since we are warranted that a malicious receiver can only obtain information of the form $\mathbf{A}\hat{\mathbf{x}}_1 + \hat{\mathbf{x}}_2$ (for an arbitrarily chosen matrix \mathbf{A}). In *non-certified groups*, the linearity of the group is no longer checkable by just looking at its description \mathbb{G} . Here we can only appeal to the fact that have a bound on the size of the output learned by the receiver, enforced by the fact that our OT achieves rate-1: The second OT message is too short to encode both $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$. In these settings, we need the full power of bounded-output AR codes, in order to show the statistical privacy of the above protocol.

3 Related Work

A recent line of works [12] proposed a new approach to constructing semi-honest OT with a rate approaching 1. This framework can be instantiated from a wide range of standard assumptions, such as the DDH, QR and LWE problems. The core idea of this approach is to construct OT from a special type of packed linearly homomorphic encryption scheme which allows compressing ciphertexts after homomorphic evaluation. Pre-evaluation ciphertexts in

D. Aggarwal et al. 2:11

such packed encryption schemes typically need to encrypt a *structured plaintext* containing redundant information to guarantee correctness of homomorphic evaluation. In the context of statistical sender privacy, this presents an issue as a malicious receiver may deviate from the structure required by the protocol to (potentially) learn correlated information about m_0 and m_1 .

Regarding the construction of SSP OT, all current schemes roughly follow one of three approaches sketched below.

The Two Keys Approach [27, 3, 22, 6]

In this construction blueprint, the receiver message ot_1 specifies two (correlated) public keys pk_0 and pk_1' under potentially different public key encryption schemes. The sender's message ot_2 now consists of two ciphertexts $c_0 = \mathsf{Enc}(\mathsf{pk}_0, m_0)$ and $c_1 = \mathsf{Enc}'(\mathsf{pk}_1', m_1)$. Statistical sender privacy is established by choosing the correlation between the keys pk_0 and pk_1' in such a way that one of these keys must be lossy, and that this is either directly enforced by the underlying structure or checkable by the sender. Here, lossiness means that either c_0 or c_1 loses information about their respective encrypted message. In group-based constructions following this paradigm [27, 3, 22], the sender must trust that the structure on which the encryption schemes are defined actually implements a group in order to be convinced that either pk_0 or pk_1' is lossy. We say that the group $\mathbb G$ must be a certified group. This is problematic if the group $\mathbb G$ is chosen by the receiver, as the group $\mathbb G$ could e.g. have non-trivial subgroups which prevent lossiness.

Furthermore, note that since the sender's message ot_2 contains two ciphertexts, each of which should, from the sender's perspective be potentially decryptable, this approach is inherently limited to rates below 1/2.

The Compactness Approach [5]

The second approach to construct SSP OT is based on high rate OT. Specifically, assume we are starting with any two round OT protocol with a (download) rate greater than 1/2, say for the sake of simplicity with rate close to 1. This means that the sender's message ot_2 is shorter than the concatenation of m_0 and m_1 . But this means that, from an information theoretic perspective ot_2 must lose information about either m_0 or m_1 . This lossiness can now be used to bootstrap statistical sender privacy as follows. The sender chooses two random messages r_0 and r_1 and uses them as his input to the OT. Moreover, he uses a randomness extractor to derive a key k_0 from r_0 and k_1 from r_1 respectively. Now the sender provides two one-time pad encrypted ciphertexts $c_0 = k_0 \oplus m_0$ and $c_1 = k_1 \oplus m_1$ to the receiver. A receiver with choice bit b can then recover r_b from the OT, derive the key k_b via the randomness extractor and obtain m_b by decrypting c_b .

To argue statistical sender privacy using this approach, we need to ensure that one of the keys k_0 or k_1 is uniformly random from a malicious receivers perspective. Roughly speaking, due to the discussion above the second OT message ot_2 needs to lose either half of the information in r_0 or r_1 . Thus, in the worst case, the receiver could learn half of the information in each r_0 and r_1 from ot_2 . Consequently, we need a randomness extractor which produces a uniformly random output as long as its input has n/2 bits of min-entropy. Thus, we can prove statistical sender privacy for messages of length smaller than n/2.

But in terms of communication efficiency, this means that we used a high rate n-bit string OT to implement a string OT of length $\leq n/2$, which means that the rate of the SSP OT we've constructed is less than 1/2. This is true without even taking into account the addition

communication cost required to transmit the ciphertexts c_0 and c_1 . Thus, this approach effectively trades high rate for statistical sender privacy at the expense of falling back to a lower rate. We conclude that this approach is also fundamentally stuck at rate 1/2.

The Non Black-Box Approach [7, 18]

While the above discussion seems to imply that there might be an inherent barrier in achieving SSP OT with rate > 1/2, there is in fact a way to convert any SSP OT protocol into a rate-1 SSP OT protocol using sufficiently powerful tools. Specifically, using a rate-1 fully-homomorphic encryption (FHE) scheme [7, 18], the receiver can delegate the decryption of ot_2 to the sender. In more detail, assume that $OT_3(st, ot_2)$ is the decryption operation which is performed by the receiver at the end of the SSP OT protocol. By providing an FHE encryption FHE. Enc(st) of the OT receiver state st along with the first message ot_1 , the receiver enables the sender to perform $OT_3(st, ot_2)$ homomorphically, resulting in an FHE encryption c of the receivers output m_b . Now the receiver merely has to decrypt c to recover m_b . In terms of rate, note that the OT sender message now merely consists of c, which is rate-1 as the FHE scheme is rate-1. Further note that this transformation does not harm SSP security, as from the sender's view the critical part of the protocol is over once ot_2 has been computed. I.e. for the sender performing the homomorphic decryption is merely a post-processing operation. On the downside, this transformation uses quite heavy tools. In particular, this transformation needs to make non black-box use of the underlying SSP OT protocol by performing the OT_3 operation homomorphically.

In summary, to the best of our knowledge, all previous approaches to construct SSP OT are either fundamentally stuck at rate 1/2 or make non black-box usage of the underlying cryptographic machinery, making it prohibitively expensive to run such a protocol in practice.

Finally, we mention that if one wishes to settle on a computational instead of statistical privacy for the sender, it is possible to build rate-1 OT using existing techniques by relying on super-polynomial hardness assumptions. The idea is that the parties will first engage in a (low-rate) OT protocol OT₁, so that the receiver will learn one of the two random PRG seeds (s_0, s_1) sampled by the sender. In parallel, the sender prepares two ciphertexts $(\mathsf{ct}_0 := \mathsf{PRG}(s_0) \oplus m_0, \mathsf{ct}_1 := \mathsf{PRG}(s_1) \oplus m_1)$ for his two input messages (m_0, m_1) , and communicates one of them to the receiver using a semi-honest rate-1 OT protocol. Even given both (ct_0, ct_1) the receiver cannot recover both m_0 and m_1 , because OT_1 will guarantee at least one of the seeds remains computationally hidden to the receiver. The above protocol is rate-1 because the added communication of obliviously transferring (s_0, s_1) is independent of the size of m_0 . The main drawback of this above protocol is that, since we do not rely on a trusted setup, we cannot extract the choice bit in polynomial time from the receiver, and hence we will have to rely on complexity leveraging to establish sender security. In particular, the best we can guarantee is that a malicious computationally-bounded receiver cannot compute both messages of the sender. This notion will fall short in replacing rate-1 SSP OT in the aforementioned applications.

References

Divesh Aggarwal, Yevgeniy Dodis, and Shachar Lovett. Non-malleable codes from additive combinatorics. In David B. Shmoys, editor, Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014, pages 774–783. ACM, 2014. doi: 10.1145/2591796.2591804.

D. Aggarwal et al. 2:13

2 Divesh Aggarwal, Nico Döttling, Jesko Dujmovic, Mohammad Hajiabadi, Giulio Malavolta, and Maciej Obremski. Algebraic restriction codes and their applications. *IACR Cryptol. ePrint Arch.*, page 1177, 2021. URL: https://eprint.iacr.org/2021/1177.

- William Aiello, Yuval Ishai, and Omer Reingold. Priced oblivious transfer: How to sell digital goods. In Birgit Pfitzmann, editor, Advances in Cryptology EUROCRYPT 2001, International Conference on the Theory and Application of Cryptographic Techniques, Innsbruck, Austria, May 6-10, 2001, Proceeding, volume 2045 of Lecture Notes in Computer Science, pages 119–135. Springer, 2001. doi:10.1007/3-540-44987-6_8.
- 4 Saikrishna Badrinarayanan, Rex Fernando, Aayush Jain, Dakshita Khurana, and Amit Sahai. Statistical ZAP arguments. In Anne Canteaut and Yuval Ishai, editors, Advances in Cryptology EUROCRYPT 2020 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III, volume 12107 of Lecture Notes in Computer Science, pages 642–667. Springer, 2020. doi:10.1007/978-3-030-45727-3 22.
- 5 Saikrishna Badrinarayanan, Sanjam Garg, Yuval Ishai, Amit Sahai, and Akshay Wadia. Two-message witness indistinguishability and secure computation in the plain model from new assumptions. In Tsuyoshi Takagi and Thomas Peyrin, editors, Advances in Cryptology ASIACRYPT 2017 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part III, volume 10626 of Lecture Notes in Computer Science, pages 275–303. Springer, 2017. doi:10.1007/978-3-319-70700-6_10.
- 6 Zvika Brakerski and Nico Döttling. Two-message statistically sender-private OT from LWE. In Amos Beimel and Stefan Dziembowski, editors, Theory of Cryptography 16th International Conference, TCC 2018, Panaji, India, November 11-14, 2018, Proceedings, Part II, volume 11240 of Lecture Notes in Computer Science, pages 370-390. Springer, 2018. doi:10.1007/978-3-030-03810-6_14.
- 7 Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. Leveraging linear decryption: Rate-1 fully-homomorphic encryption and time-lock puzzles. In Dennis Hofheinz and Alon Rosen, editors, Theory of Cryptography 17th International Conference, TCC 2019, Nuremberg, Germany, December 1-5, 2019, Proceedings, Part II, volume 11892 of Lecture Notes in Computer Science, pages 407–437. Springer, 2019. doi:10.1007/978-3-030-36033-7_16.
- 8 Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In Rafail Ostrovsky, editor, *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 97–106. IEEE Computer Society, 2011. doi:10.1109/FOCS.2011.12.
- 9 Mahdi Cheraghchi and Venkatesan Guruswami. Non-malleable coding against bit-wise and split-state tampering. In Yehuda Lindell, editor, *Theory of Cryptography 11th Theory of Cryptography Conference, TCC 2014, San Diego, CA, USA, February 24-26, 2014. Proceedings*, volume 8349 of *Lecture Notes in Computer Science*, pages 440–464. Springer, 2014. doi: 10.1007/978-3-642-54242-8_19.
- Yevgeniy Dodis, Xin Li, Trevor D. Wooley, and David Zuckerman. Privacy amplification and non-malleable extractors via character sums. In Rafail Ostrovsky, editor, IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011, pages 668-677. IEEE Computer Society, 2011. doi:10.1109/FOCS. 2011.67.
- 11 Yevgeniy Dodis and Daniel Wichs. Non-malleable extractors and symmetric key cryptography from weak secrets. In Michael Mitzenmacher, editor, *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 June 2, 2009*, pages 601–610. ACM, 2009. doi:10.1145/1536414.1536496.
- Nico Döttling, Sanjam Garg, Yuval Ishai, Giulio Malavolta, Tamer Mour, and Rafail Ostrovsky. Trapdoor hash functions and their applications. In Alexandra Boldyreva and Daniele Micciancio, editors, Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology

- Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part III, volume 11694 of Lecture Notes in Computer Science, pages 3-32. Springer, 2019. doi:10.1007/978-3-030-26954-8_1.
- Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski. Non-malleable codes from two-source extractors. In Ran Canetti and Juan A. Garay, editors, Advances in Cryptology CRYPTO 2013 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II, volume 8043 of Lecture Notes in Computer Science, pages 239-257. Springer, 2013. doi:10.1007/978-3-642-40084-1_14.
- Stefan Dziembowski, Krzysztof Pietrzak, and Daniel Wichs. Non-malleable codes. J. ACM, 65(4):20:1-20:32, 2018. doi:10.1145/3178432.
- Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. In David Chaum, Ronald L. Rivest, and Alan T. Sherman, editors, Advances in Cryptology: Proceedings of CRYPTO '82, Santa Barbara, California, USA, August 23-25, 1982, pages 205-210. Plenum Press, New York, 1982. doi:10.1007/978-1-4757-0602-4_19.
- Taher El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. R. Blakley and David Chaum, editors, Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings, volume 196 of Lecture Notes in Computer Science, pages 10-18. Springer, 1984. doi:10.1007/3-540-39568-7_2.
- 17 Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 June 2, 2009, pages 169–178. ACM, 2009. doi: 10.1145/1536414.1536440.
- 18 Craig Gentry and Shai Halevi. Compressible FHE with applications to PIR. In Dennis Hofheinz and Alon Rosen, editors, Theory of Cryptography 17th International Conference, TCC 2019, Nuremberg, Germany, December 1-5, 2019, Proceedings, Part II, volume 11892 of Lecture Notes in Computer Science, pages 438–464. Springer, 2019. doi:10.1007/978-3-030-36033-7_17.
- 19 Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In Ran Canetti and Juan A. Garay, editors, Advances in Cryptology CRYPTO 2013 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I, volume 8042 of Lecture Notes in Computer Science, pages 75-92. Springer, 2013. doi:10.1007/978-3-642-40041-4_5.
- 20 Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. J. Cryptol., 7(1):1–32, 1994.
- Vipul Goyal, Abhishek Jain, Zhengzhong Jin, and Giulio Malavolta. Statistical zaps and new oblivious transfer protocols. In Anne Canteaut and Yuval Ishai, editors, Advances in Cryptology EUROCRYPT 2020 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III, volume 12107 of Lecture Notes in Computer Science, pages 668–699. Springer, 2020. doi:10.1007/978-3-030-45727-3_23.
- Shai Halevi and Yael Tauman Kalai. Smooth projective hashing and two-message oblivious transfer. J. Cryptol., 25(1):158–193, 2012. doi:10.1007/s00145-010-9092-8.
- Yuval Ishai and Anat Paskin. Evaluating branching programs on encrypted data. In Salil P. Vadhan, editor, Theory of Cryptography, 4th Theory of Cryptography Conference, TCC 2007, Amsterdam, The Netherlands, February 21-24, 2007, Proceedings, volume 4392 of Lecture Notes in Computer Science, pages 575-594. Springer, 2007. doi:10.1007/978-3-540-70936-7_31.
- Joe Kilian. Founding cryptography on oblivious transfer. In Janos Simon, editor, *Proceedings* of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA, pages 20–31. ACM, 1988. doi:10.1145/62212.62215.
- 25 Eyal Kushilevitz and Rafail Ostrovsky. Replication is NOT needed: SINGLE database, computationally-private information retrieval. In 38th Annual Symposium on Foundations

D. Aggarwal et al.

of Computer Science, FOCS '97, Miami Beach, Florida, USA, October 19-22, 1997, pages 364–373. IEEE Computer Society, 1997. doi:10.1109/SFCS.1997.646125.

- 26 Xin Li. Non-malleable extractors, two-source extractors and privacy amplification. In 53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012, pages 688-697. IEEE Computer Society, 2012. doi:10.1109/FOCS.2012.26.
- 27 Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In S. Rao Kosaraju, editor, *Proceedings of the Twelfth Annual Symposium on Discrete Algorithms, January 7-9, 2001, Washington, DC, USA*, pages 448-457. ACM/SIAM, 2001. URL: http://dl.acm.org/citation.cfm?id=365411.365502.
- Rafail Ostrovsky, Anat Paskin-Cherniavsky, and Beni Paskin-Cherniavsky. Maliciously circuit-private FHE. In Juan A. Garay and Rosario Gennaro, editors, Advances in Cryptology CRYPTO 2014 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I, volume 8616 of Lecture Notes in Computer Science, pages 536-553. Springer, 2014. doi:10.1007/978-3-662-44371-2_30.
- 29 Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, Advances in Cryptology EUROCRYPT '99, International Conference on the Theory and Application of Cryptographic Techniques, Prague, Czech Republic, May 2-6, 1999, Proceeding, volume 1592 of Lecture Notes in Computer Science, pages 223–238. Springer, 1999. doi:10.1007/3-540-48910-X_16.
- Michael O. Rabin. How to exchange secrets with oblivious transfer. *IACR Cryptol. ePrint Arch.*, page 187, 2005. URL: http://eprint.iacr.org/2005/187.
- 31 Umesh Vazirani. *Randomness, Adversaries and Computation*. PhD thesis, EECS, UC Berkeley, 1986. Ph.D. Thesis.
- 32 Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In 23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982, pages 80–91. IEEE Computer Society, 1982. doi:10.1109/SFCS.1982.45.
- Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In 27th Annual Symposium on Foundations of Computer Science, Toronto, Canada, 27-29 October 1986, pages 162–167. IEEE Computer Society, 1986. doi:10.1109/SFCS.1986.25.