Information-Theoretic Distributed Point Functions

Elette Boyle
IDC Herzliya, Israel

Niv Gilboa
Ben-Gurion University of the Negev, Beer-Sheva, Israel

Yuval Ishai
Technion, Haifa, Israel

Victor I. Kolobov
Technion, Haifa, Israel

Abstract

A *distributed point function* (DPF) (Gilboa-Ishai, Eurocrypt 2014) is a cryptographic primitive that enables compressed additive secret-sharing of a secret weight-1 vector across two or more servers. DPFs support a wide range of cryptographic applications, including efficient private information retrieval, secure aggregation, and more. Up to now, the study of DPFs was restricted to the *computational* security setting, relying on one-way functions. This assumption is necessary in the case of a dishonest majority.

We present the first statistically private 3-server DPF for domain size N with subpolynomial key size $N^{o(1)}$. We also present a similar *perfectly* private 4-server DPF. Our constructions offer benefits over their computationally secure counterparts, beyond the superior security guarantee, including better computational complexity and better protocols for distributed key generation, all while having comparable communication complexity for moderate-sized parameters.

2012 ACM Subject Classification Theory of computation → Cryptographic primitives

Keywords and phrases Information-theoretic cryptography, homomorphic secret sharing, private information retrieval, secure multiparty computation

Digital Object Identifier 10.4230/LIPIcs.ITC.2022.17

Funding *Elette Boyle*: Supported by AFOSR Award FA9550-21-1-0046, ERC Project HSS (852952), ERC Project NTSC (742754), and a Google Research Scholar Award.
Niv Gilboa: Supported by ISF grant 2951/20, ERC grant 876110, and a grant by the BGU Cyber Center.
Yuval Ishai: Supported by ERC Project NTSC (742754), BSF grant 2018393, and ISF grant 2774/20.
Victor I. Kolobov: Supported by ERC Project NTSC (742754) and ISF grant 2774/20.

1 Introduction

A *Distributed Point Function* (DPF) [28, 13] enables splitting any secret *point function* $f_{\alpha,\beta}$ (i.e., for which $f_{\alpha,\beta}(x) = \beta$ if $x = \alpha$, and 0 otherwise) into m succinctly described function shares f_i, that individually hide $f_{\alpha,\beta}$, and which support a simple additive per-input reconstruction $f_{\alpha,\beta}(x) = \sum_i f_i(x)$ over some fixed Abelian group. More concretely, each function share f_i is described by a *key* k_i such that with an appropriate evaluation algorithm Eval it holds that $\mathsf{Eval}(k_i, x) = f_i(x)$. In effect, this provides a compressed additive secret-sharing of a secret weight-1 vector across servers.

DPFs have a wide range of cryptographic applications, including Private Information Retrieval (PIR) [17, 16, 28], anonymous messaging systems [18, 35], secure aggregation and statistical analysis [13, 7], private set intersection [39, 22], secure computation for

RAM programs [23, 15] and programs with mixed-mode operations [14, 8], and recently pseudorandom correlation generators [9, 10, 11], with applications to secure computation and beyond.

As with many cryptographic notions, the security property of DPFs can be either *computational* (based on computational hardness assumptions), or *information theoretic*. The vast majority of attention to date has been placed in the two-server regime, where it is known that nontrivial DPFs require the existence of one-way functions [28, 12]. In turn, from one-way functions, efficient two-server DPF constructions have been demonstrated with small key size, which grows logarithmically with the domain size of $f_{\alpha,\beta}$ [28, 13].

However, as soon as one steps beyond two servers to an honest majority, the impossibility no longer holds, and the question of minimizing the key size of information-theoretic DPFs becomes wide open. Despite its neglect up to now, the regime of information-theoretically secure DPFs offers potential for application scenarios where information-theoretic security is desired (or required), as well as appealing potential for simplicity of constructions. Another important motivation for information-theoretic constructions is the possibility to avoid the limitations of current techniques for distributed key generation of computationally secure DPF [23].

1.1 Our Contribution

We initiate an investigation of information-theoretically secure DPFs (IT DPFs for short), focusing on the case of non-colluding servers (i.e., security threshold $t = 1$). While simple constructions based on Reed-Muller codes are implicit in the PIR literature [17], these have polynomial key size of $O(N^{1/(m-1)})$, where N is the domain size and m is the number of servers. In contrast, the new generation of PIR schemes [40, 25, 24, 4], which achieve sub-polynomial communication, do not directly give rise to standard DPFs. Instead, they imply a relaxed form of DPF in which the output is not shared additively. While this suffices for the PIR application, it does not suffice for most other applications of DPFs. Even in the PIR context, an additive representation is helpful for maximizing the download rate [26].

Our primary technical contribution is in bridging this gap. We obtain the first statistically private 3-server DPF for domain size N with subpolynomial key size $N^{o(1)}$. We also present a similar *perfectly* private 4-server DPF. Our constructions offer benefits over their computationally secure counterparts, beyond the superior security guarantee, including better computational complexity and potential for "MPC friendliness" in the sense of efficient distributed key generation, all while having comparable key size[1] for moderate-sized parameters.

We obtain the following main results:

▶ **Theorem 1** (4-server perfectly secure IT DPF, informal). *Let $p \geq 3$ be a prime and $s \geq 1$ an integer. There exists a perfectly secure 4-server DPF, for point functions with output group $\mathbb{Z}_{p^s}$ and key size $O\left(s \log(p) \cdot 2^{2p\sqrt{\log N \log\log N}}\right)$.*

[1] This assumes that β is taken from a small output group, such as $\mathbb{Z}_2$, which suffices for many applications of DPF. While in this work we focus on asymptotic efficiency and do not attempt to optimize concrete efficiency, our techniques can be applied to concretely efficient variants of the "matching vector" based PIR schemes on which we rely (see Table 2 in the full version of [31]). Unlike the Reed-Muller based 3-server PIR, these variants can be practical even for (sparse, virtual) database of size $\approx 2^{60}$, which arise in private keyword search applications.

▶ **Theorem 2** (3-server statistically-secure IT DPF, informal). *Let $p \geq 2$ be a prime. There exists a $2^{-\lambda}$-statistically secure 3-server DPF, for point functions with output group $\mathbb{Z}_p$ and key size $O\left(\lambda \log(p) \cdot 2^{k(p)\sqrt{\log N \log\log N}}\right)$ where $k(2) = 6$, $k(3) = 10$, and $k(p) = 2p$ if $p \geq 5$.*

Due to the prime p appearing in the exponent in the key size, Theorem 1 permits only groups of the form $\mathbb{Z}_{p^s}$ for small prime p (or products of such groups via CRT). The same is true for Theorem 2, except that there we further have the restriction of $s = 1$. However, for many applications of DPF (including both "reading" and "writing") an output group of $\mathbb{Z}_2$ suffices, in which case Theorem 2 gives an efficient construction. Moreover, in applications of DPF that require a group of a large characteristic (e.g., for aggregation [7] or weighted private set intersection [22]), Theorem 1 gives an efficient construction over $\mathbb{Z}_{3^s}$ for a sufficiently large s.

We further explore advantages of our IT DPF constructions over existing computationally secure constructions, beyond their stronger security guarantees. We explicitly demonstrate one such benefit: simplicity of distributed key generation. This relates to the procedure of two or more clients jointly executing the DPF key generation algorithm for an input point function that is *secret shared* across servers (i.e., where no client individually knows the secret $f_{\alpha,\beta}$). This "Distributed Gen" procedure is a crucial and costly part of important DPF-based applications. Distributed Gen protocols in the computational setting currently fall into one of two categories. They use either generic MPC machinery, which requires non-black box secure computation of cryptographic primitives such as PRGs, or tailored protocols [23] requiring computation that is proportional to the size of the input domain and a number of communication rounds that is logarithmic in that size. Moreover, there is no known approach for distributing the key generation of 2-server DPF in the *malicious* security setting that makes black-box use of a PRG, regardless of round complexity.

In contrast, the simpler structure of keys in IT DPFs implies the following:

▶ **Theorem 3** (Distributed key generation, informal). *There exist protocols for distributed generation of the keys required in Theorems 1 and 2 that are information-theoretically secure for $m \geq 3$ servers with one malicious corruption, or for 2PC in the OT-hybrid model, have computation and communication cost $\tilde{O}(h)$ for required key size h, and $O(\log h)$ rounds. Alternatively, settling for computational security, there are such* constant-round *protocols that only make a* black-box *use of a PRG.*

1.2 Overview of Techniques

Our information-theoretic (IT) DPF constructions are based on a related primitive, IT *private information retrieval* (PIR) [17]. A PIR scheme allows a client to retrieve a single bit from a database D of N bits, by communicating with $m \geq 2$ servers, such that no server learns the client's bit index. Multi-server PIR served as an original driving motivation behind the introduction of DPFs, as an m-server DPF directly yields an m-server PIR protocol. In this work, however, we study this connection in the other direction: building DPF from PIR.

Assuming the m-server IT PIR scheme satisfies that each server responds with a single bit to the client query, and that the client's reconstruction is additive, then in fact we obtain an IT DPF for the point function $f_{\alpha,1}$, by having the client query for index α, and the servers considering the database D_x which has the value 1 at index x, and 0 at all other indices. As we will see, some existing classes of IT PIR schemes fit into this framework, and thus yield IT DPF with similar communication. However, other categories of IT PIR constructions will require more work.

Known IT PIR schemes can be roughly classified into three generations. The first-generation schemes, originating in the work of [17], are based on Reed-Muller codes, and achieve communication complexity $N^{1/\Theta(m)}$. As it turns out, for $m \geq 3$ these schemes imply IT DPF schemes with similar communication complexity. Hence, these constructions serve as our baseline.

▶ **Theorem 4** (Reed-Muller IT DPF - Informal, implicit in [17, 3]). *Let $p \geq 2$ be a prime and $m \geq 2$ an integer. There exists a perfectly secure m-server DPF, for point functions with output group $\mathbb{Z}_p$ and key size $O_m(\log(p) \cdot N^{1/(m-1)})$.*

Note that the above theorem is stated for prime sized cyclic groups, as the usual Reed-Muller code based constructions are based on extension fields. However, by employing a similar construction over extension rings, which are used in the MPC literature on secure computation over rings [20, 19], it is possible to extend the above result to any prime power sized cyclic groups, and hence to any Abelian group.

In the second-generation PIR scheme of [5] the exponent of N vanishes super-linearly with m (but is still a constant for any fixed m), and corresponding IT DPF constructions can be derived as well.

Finally, third-generation PIR schemes [40, 25, 24, 4] achieve $N^{o(1)}$ communication complexity with as low as 3 servers, or even 2 servers if we allow the servers to respond with $N^{o(1)}$-bit messages. These schemes are based on a nontrivial combinatorial object called a *matching vectors* (MV) family, based on the work of [30]. In addition to their superior asymptotic communication complexity, as was discussed in [4, 31], for moderate size parameters, these schemes can achieve superior concrete complexity as well, by employing an MV family based on the work of Frankl [27].

Unfortunately, unlike the first and second generation PIR schemes, MV-based PIR schemes do not readily imply a DPF. Indeed, for some specific output ring R, the schemes imply a form of "quasi-additive" DPF, where β can either be chosen to be zero or *some* invertible element ζ of R (which depends on the choice of α and the randomness of the key generation). Note that given such a quasi-additive DPF for m servers, it can be converted to a true DPF with $2m$ servers by replicating each quasi-additive DPF share among two servers, as well as secret sharing $\zeta = \zeta_1 + \zeta_2$ among them. Indeed, this principle can also be applied to *balanced* PIR schemes, where the output message of each server is a vector instead of a single element. By applying this to the 2-server "quasi-additive" DPF implicit in the 2-server PIR work of Dvir and Gopi [24] we obtain Theorem 1.

The above discussion leaves open the question of obtaining a 3-server IT DPF with communication complexity $N^{o(1)}$. We are able to construct such a DPF with statistical security. One subtle difficulty is that even though the nonzero payload β generated by the quasi-additive DPF depends on the randomness of the key generation, this entropy is eliminated when we condition on the view of a server. Our strategy is to repeat the quasi-additive DPF σ times, for point functions $f_{\alpha,\beta_1}, \ldots, f_{\alpha,\beta_\sigma}$, such that with probability $1/2$ we take $\beta_i = 0$ and take a nonzero β_i otherwise. This ensures that even when fixing α and conditioning on the view of a single server, the payload β has some entropy. Then, we provide each server with its respective σ keys. In addition, denoting by B the vector of payloads, such that each coordinate i takes the value β_i, we provide the servers with a random vector r satisfying $\langle r, B \rangle = \beta$ for the desired output value β.

By the perfect security of the PIR, the σ keys alone do not reveal any information. However, since r is correlated with them and with β, some information on the relation between α and β is revealed. To argue that the amount of information is a negligible function of σ, we first invoke the leftover hash lemma to argue that for a uniformly random r', the

distribution of $(r', \langle r', B\rangle)$ is statistically close to uniform. We then argue that if we condition this joint distribution on different values of $\langle r', B\rangle$, the distribution of r' cannot change much. By applying this principle to the PIR scheme of [4], we obtain Theorem 2.

2 Preliminaries

Notation. For $N \in \mathbb{N}$ we let $[N] = \{1, \ldots, N\}$. We denote the inner product of two vectors u and v of the same length by $\langle u, v\rangle = \sum_i u_i v_i$.

Probability. For two distributions D_1, D_2 we denote by $d(D_1, D_2) = \frac{1}{2}\sum_\omega |\Pr_{D_1}[\omega] - \Pr_{D_2}[\omega]|$ their total variation distance. We denote by U_ℓ uniformly distributed random strings of length ℓ.

Groups. We represent an Abelian group $\mathbb{G}$ of the form $\mathbb{G} = \mathbb{Z}_{q_1} \times \cdots \times \mathbb{Z}_{q_\ell}$, for prime powers $q_1, \ldots, q_\ell$ by $\hat{\mathbb{G}} = (q_1, \ldots, q_\ell)$ and represent a group element of $\mathbb{G}$ by a sequence of ℓ non-negative integers.

Point functions. Given a domain size N and Abelian group $\mathbb{G}$, a *point function* $f_{\alpha,\beta} : [N] \to \mathbb{G}$ for $\alpha \in [N]$ and $\beta \in \mathbb{G}$ evaluates to β on input α and to $0 \in \mathbb{G}$ on all other inputs. We denote by $\hat{f}_{\alpha,\beta} = (N, \hat{\mathbb{G}}, \alpha, \beta)$ the representation of such a point function.

2.1 Distributed Point Functions

We begin with a formal definition of the cryptographic primitive of distributed point functions (DPFs).

▶ **Definition 5** (DPF [28, 13]). *A (1-private) m-server* distributed point function*, or m-DPF for short, is a tuple of algorithms $\Pi = (\mathsf{Gen}, \mathsf{Eval}_0, \ldots, \mathsf{Eval}_{m-1})$ with the following syntax:*

- $\mathsf{Gen}(1^\lambda, \hat{f}_{\alpha,\beta}) \to (k_0, \ldots, k_{m-1})$*: On input security parameter $\lambda \in \mathbb{N}$ and point function description $\hat{f}_{\alpha,\beta} = (N, \hat{\mathbb{G}}, \alpha, \beta)$, the (randomized) key generation algorithm* Gen *returns an m-tuple of keys $k_0, \ldots, k_{m-1} \in \{0,1\}^*$. We assume that N and $\mathbb{G}$ are determined by each key.*
- $\mathsf{Eval}_i(k_i, x) \to y_i$*: On input key $k_i \in \{0,1\}^*$ and input $x \in [N]$ the (deterministic) evaluation algorithm of server i,* Eval_i*, returns a group element $y_i \in \mathbb{G}$.*

We require Π to satisfy the following requirements:

- ***Correctness:*** *For every λ, $\hat{f}_{\alpha,\beta} = (N, \hat{\mathbb{G}}, \alpha, \beta)$ and $x \in [N]$, if $(k_0, \ldots, k_{m-1}) \leftarrow \mathsf{Gen}(1^\lambda, \hat{f}_{\alpha,\beta})$, then $\Pr\left[\sum_{i=0}^{m-1} \mathsf{Eval}_i(k_i, x) = f_{\alpha,\beta}(x)\right] = 1$.*
- ***Security:*** *Consider the following semantic security challenge experiment for a corrupted server $T \in \{0, \ldots, m-1\}$:*
  1. *The adversary gives challenge point function descriptions $(\hat{f}^1 = (N_1, \hat{\mathbb{G}}_1, \alpha_1, \beta_1), \hat{f}^2 = (N_2, \hat{\mathbb{G}}_2, \alpha_2, \beta_2)) \leftarrow \mathcal{A}(1^\lambda)$ with $N_1 = N_2$ and $\hat{\mathbb{G}}_1 = \hat{\mathbb{G}}_2$.*
  2. *The challenger samples $b \xleftarrow{\$} \{0,1\}$ and $(k_0, \ldots, k_{m-1}) \leftarrow \mathsf{Gen}(1^\lambda, \hat{f}^b)$.*
  3. *The adversary outputs a guess $b' \leftarrow \mathcal{A}(k_T)$.*

  Denote by $\mathrm{Adv}(1^\lambda, \mathcal{A}, T) := \Pr[b = b'] - 1/2$ the advantage of $\mathcal{A}$ in guessing b in the above experiment. For circuit size bound $S = S(\lambda)$ and advantage bound $\epsilon(\lambda)$, we say that Π is (S, ϵ)-secure if for all T, and all non-uniform adversaries $\mathcal{A}$ of size $S(\lambda)$, we have $\mathrm{Adv}(1^\lambda, \mathcal{A}, T) \le \epsilon(\lambda)$. We say that Π is:

- Computationally ϵ-secure *if it is* (S, ϵ)*-secure for all polynomials* S*.*
- Computationally secure *if it is* $(S, 1/S)$*-secure for all polynomials* S*.*
- Statistically ϵ-secure *if it is* (S, ϵ)*-secure for all* S*. When* ϵ *is omitted it is understood to be negligible in* λ*.*
- Perfectly secure *if it is statistically* 0*-secure.*

3 Constructions

In this section we give two main constructions of information-theoretic DPF. The first is perfectly secure but requires 4 servers. The second requires just 3 servers but only offers statistical security.

3.1 4-server MV-based DPF

Our first result is the following, based on 2-server "quasi-additive" DPF, implicit in [24]. In that quasi-additive DPF the response of each server is a vector, such that the result is constructed by taking an inner product of the sum of the servers' vectors with a reconstruction vector that is a function of the point α and therefore must not be part of the DPF key. However, the reconstruction vector can be readily secret-shared between two keys. Using four servers such that each pair of servers receives one of the original keys of the two-server DPF of [24] and secret shares of the reconstruction vector, results in a scheme in which each server returns a single element.

▶ **Theorem 6.** *Let* $p \geq 3$ *be a prime and* $s \geq 1$ *an integer. There exists a perfectly secure* 4*-DPF, for point functions with output group* $\mathbb{Z}_{p^s}$*,* $\beta \in \{0, 1\}$*, domain size* N*, and key size* $|k_i| = O\left(s \log(p) \cdot 2^{2p\sqrt{\log N \log\log N}}\right)$*,* $i \in \{0, 1, 2, 3\}$*.*

We prove the theorem in several steps. The following theorem is a generalization of the construction implicit in [24] to the case of matching vector families over moduli $m = 2p^s$, for a prime $p \geq 3$ and an integer s. Here, Share is a randomized algorithm that shares an input $\alpha \in [N]$ between two servers, Conv is a share conversion algorithm employed by the servers that maps the shares of α to shares of $f_{\alpha,\zeta}(x)$, for some nonzero ζ, and Rec is an algorithm that allows, given α, to recover $f_{\alpha,\zeta}(x)$.

▶ **Theorem 7** (Dvir Gopi share conversion [24], generalized)**.** *Let* $p \geq 3$ *be a prime,* $s \geq 1$ *an integer, and denote* $q = 2p^s$*. For every integer* $N \geq 1$ *there exist a randomized mapping* $\mathsf{Share} : [N] \to \mathbb{Z}_q^h \times \mathbb{Z}_q^h$*,* $h = O\left(\log(q) \cdot 2^{2p\sqrt{\log N \log\log N}}\right)$*, and deterministic mappings* $\mathsf{Conv} : \{0, 1\} \times \mathbb{Z}_q^h \times [N] \to \mathbb{Z}_{p^s}^h$ *and* $\mathsf{Rec} : [N] \to \mathbb{Z}_{p^s}^h$*, such that*

- *For every* $\alpha, x \in [N]$*,*

$$\Pr\left[(c_0, c_1) \leftarrow \mathsf{Share}(\alpha) : \left\langle \mathsf{Rec}(\alpha), \sum_{i=0}^{1} \mathsf{Conv}(i, c_i, x) \right\rangle \begin{cases} \in \{2, -2\}, & x = \alpha \\ = 0, & x \neq \alpha \end{cases}\right] = 1.$$

- *For every* $\alpha, \alpha' \in [N]$ *and* $i \in \{0, 1\}$*,*

$$[(c_0, c_1) \leftarrow \mathsf{Share}(\alpha); \mathit{Output}\ c_i] \equiv [(c_0, c_1) \leftarrow \mathsf{Share}(\alpha'); \mathit{Output}\ c_i]$$

- $\mathsf{Share}, \mathsf{Conv}, \mathsf{Rec}$ *are computable in time polynomial in their input and output size.*

We will first need the following result from [30].

Notation: Let $\mathsf{Share}, \mathsf{Conv}, \mathsf{Rec}$ be as in Theorem 7.

$\mathsf{Gen}(\hat{f}_{\alpha,\beta} = (N, \hat{\mathbb{G}} = \widehat{\mathbb{Z}_{p^s}}, \alpha, \beta \in \{0,1\}))$:
- Compute $(c_0, c_1) \leftarrow \mathsf{Share}(\alpha)$.
- Compute $r = \left[\left\langle \mathsf{Rec}(\alpha), \sum_{i=0}^{1} \mathsf{Conv}(i, c_i, \alpha)\right\rangle\right]^{-1} \mathsf{Rec}(\alpha)\beta$, and share it additively $r = r_0 + r_1$.
- Output $k_0 = (c_0, r_0), k_1 = (c_0, r_1), k_2 = (c_1, r_0), k_3 = (c_1, r_1)$.

$\mathsf{Eval}_i(k_i = (c_{j_1}, r_{j_2}), x)$:
- Compute and output $\langle r_{j_2}, \mathsf{Conv}(j_1, c_{j_1}, x)\rangle$.

Figure 1 4-server MV-based DPF.

▶ **Theorem 8** (Matching Vectors [30]). *For every integers N, s and prime p, there is $h = O\left(\log(q) 2^{2p\sqrt{\log N \log\log N}}\right)$ and a collection of vectors $(u_i, v_i)_{i\in[N]}$ in $\mathbb{Z}_q^h$ (called* matching vectors*), $q = 2p^s$, such that*
- *For every $i \in [N]$, $\langle u_i, v_i\rangle = 0$.*
- *For every $i \neq j$, $\langle u_i, v_j\rangle \in \{1, p^s, p^s + 1\}$.*

We are now ready to prove Theorem 7.

Proof of Theorem 7. Let h and (u_i, v_i) be as in Theorem 8. $\mathsf{Share}(\alpha)$ draws a random vector $w \xleftarrow{\$} \mathbb{Z}_q^h$ and outputs $(w, w + u_\alpha)$. $\mathsf{Conv}(i, w', x)$ outputs

$$((-1)^{\langle w', v_x\rangle}, (-1)^{\langle w', v_x\rangle} v_x) \mod p^s.$$

$\mathsf{Rec}(\alpha)$ outputs $(1, -u_\alpha) \mod p^s$. Efficiency and security are obvious. Correctness follows because the expression

$$\left\langle (1, -u_\alpha), ((-1)^{\langle w, v_x\rangle}, (-1)^{\langle w, v_x\rangle} v_x) + ((-1)^{\langle w+u_\alpha, v_x\rangle}, (-1)^{\langle w+u_\alpha, v_x\rangle} v_x)\right\rangle \mod p^s$$

equals $(-1)^{\langle w, v_x\rangle} \cdot (1 - \langle u_\alpha, v_x\rangle) \cdot \left(1 + (-1)^{\langle u_\alpha, v_x\rangle}\right) \mod p^s$ which is in $\{-2, 2\}$ if $x = \alpha$ and equals 0 if $x \neq \alpha$, because then $\langle u_\alpha, v_x\rangle \in \{1, p^s, p^s + 1\}$. ◀

Using the above result, we can construct a 4-server IT DPF. Below is a construction for the output group $\mathbb{Z}_{p^s}$ and $\beta \in \{0, 1\}$. An extension to general finite Abelian group $\mathbb{G}$ and any $\beta \in \mathbb{G}$ can be done by bit decomposition, which will incur a multiplicative factor of $\log|\mathbb{Z}_q| = \log q$ in privacy loss, computational cost, and key length. However, some groups might have large key size, due to p appearing as an exponent in the key size in Theorem 6.

Proof of Theorem 6. The construction is given in Figure 1.

Security and efficiency are obvious. Correctness follows because

$$\begin{aligned}
\sum_{j_1=0}^{1}\sum_{j_2=0}^{1} \langle r_{j_2}, \mathsf{Conv}(j_1, c_{j_1}, x)\rangle &= \left\langle \sum_{j_2=0}^{1} r_{j_2}, \sum_{j_1=0}^{1} \mathsf{Conv}(j_1, c_{j_1}, x)\right\rangle \\
&= \left\langle \left[\left\langle \mathsf{Rec}(\alpha), \sum_{i=0}^{1} \mathsf{Conv}(i, c_i, \alpha)\right\rangle\right]^{-1} \mathsf{Rec}(\alpha)\beta, \sum_{j_1=0}^{1} \mathsf{Conv}(j_1, c_{j_1}, x)\right\rangle \\
&= \beta \left[\left\langle \mathsf{Rec}(\alpha), \sum_{i=0}^{1} \mathsf{Conv}(i, c_i, \alpha)\right\rangle\right]^{-1} \left\langle \mathsf{Rec}(\alpha), \sum_{i=0}^{1} \mathsf{Conv}(i, c_i, x)\right\rangle,
\end{aligned}$$

which is either β or 0 depending on whether $x = \alpha$ or $x \neq \alpha$, respectively. ◀

3.2 3-server statistically-secure MV-based DPF

To construct a 3-server statistically-secure DPF, we need the following result from [4].

▶ **Theorem 9** ([4, Theorem 3.5]). *For every domain size $N \geq 1$ there exist a randomized mapping* $\mathsf{Share} : [N] \to \mathbb{Z}_6^h \times \mathbb{Z}_6^h \times \mathbb{Z}_6^h$, $h = O\left(2^{6\sqrt{\log N \log\log N}}\right)$, *and a deterministic mapping* $\mathsf{Conv} : \mathbb{Z}_6^h \times \mathbb{Z}_6^h \times [N] \to \mathbb{Z}_2^2$, *such that*

1. *For every $\alpha, x \in [N]$,*

$$\Pr\left[(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha) : \sum_{i=0}^{2} \mathsf{Conv}(c_i, c_{(i+1) \bmod 3}, x) \begin{cases} \neq \mathbf{0}, & x = \alpha \\ = \mathbf{0}, & x \neq \alpha \end{cases}\right] = 1.$$

2. *For every $\alpha, \alpha' \in [N]$ and $i, j \in \{0, 1, 2\}$,*

$$[(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha); \mathit{Output}\ (c_i, c_j)] \equiv [(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha'); \mathit{Output}\ (c_i, c_j)]$$

3. *Share, Conv are computable in time polynomial in their input and output size.*

Below is the our main theorem for this section. Using the results of [37, 36], we also show how to extend this theorem to bigger payloads.

▶ **Theorem 10.** *Fix an integer $\lambda > 0$. The construction in Figure 2 is a statistically $\left(41 \cdot 2^{\frac{2-\lambda}{2}}\right)$-secure 3-DPF, for point functions with output group $\mathbb{G} = \mathbb{Z}_2^2$, domain size N, and key size $|k_i| = O\left(\lambda \cdot 2^{6\sqrt{\log N \log\log N}}\right)$, $i \in \{0, 1, 2\}$.*

Next, we will need an additional result.

▶ **Definition 11.** *Let X be a random variable. Then the* min-entropy *of X is*

$$H_\infty(X) = \min_x \log \frac{1}{\Pr[X = x]}$$

▶ **Lemma 12** (Leftover Hash Lemma). *Let $\mathbb{F}$ be a finite field. If X is a random variable over $\mathbb{F}^n$ with $H_\infty(X) \geq R$ and $Y \xleftarrow{\$} \mathbb{F}^n$ is drawn independently, then it holds that $d\left((Y, \langle Y, X\rangle), U_{(n+1)\log|\mathbb{F}|}\right) \leq 2^{\frac{\log|\mathbb{F}| - R}{2}}$.*

Proof of Theorem 10.

Efficiency. Follows by construction and Theorem 9.

Correctness. In fact, the Gen algorithm may not be defined if $y = 0$, as there might not be r such that $\langle r, y\rangle = \beta$. In that case we can just let Gen reveal α and β for a negligible privacy loss. When this does not happen, we need to show that $\sum_{i=1}^3 \mathsf{Eval}_i(k_i, x) = \langle r, \sum_{i=1}^3 y_i\rangle = f_{\alpha,\beta}(x)$. Indeed, when $x \neq \alpha$ we have that $\sum_{i=1}^3 y_i = 0$ because every $(c_0^\ell, c_1^\ell, c_2^\ell)$ was produced by computing either $\mathsf{Share}(\alpha)$ or $\mathsf{Share}(N+1)$. When $x = \alpha$ we have that $\sum_{i=1}^3 y_i^\ell = y^\ell$, which implies that $\langle r, \sum_{i=1}^3 y_i\rangle = \langle r, y\rangle = \beta$.

Notation: Let $\mathsf{Share}, \mathsf{Conv}$ be as in Theorem 9 with domain size $N+1$.

$\mathsf{Gen}(1^\lambda, \hat{f}_{\alpha,\beta} = (N, \hat{\mathbb{G}} = \widehat{\mathbb{Z}_2^2}, \alpha, \beta))$:

- For $\ell = 1, \ldots, \lambda$ draw $\alpha_\ell^* \xleftarrow{\$} \{\alpha, N+1\}$ and compute $(c_0^\ell, c_1^\ell, c_2^\ell) \leftarrow \mathsf{Share}(\alpha_\ell^*)$.
- For $\ell = 1, \ldots, \lambda$ set

$$y^\ell = \begin{cases} \mathsf{Conv}(c_1^\ell, c_2^\ell, \alpha) + \mathsf{Conv}(c_2^\ell, c_0^\ell, \alpha) + \mathsf{Conv}(c_0^\ell, c_1^\ell, \alpha), & \alpha_\ell^* = \alpha \\ 0, & \alpha_\ell^* = N+1 \end{cases}$$

  Denote by $y \in \mathbb{F}_4^\lambda$ the vector of all y^ℓ values concatenated, where we naturally associate elements of $\mathbb{Z}_2^2$ with these of $\mathbb{F}_4$ (mapping zero to zero).
- Choose $r \in \mathbb{F}_4^\lambda$ at random under the constraint that $\langle r, y \rangle = \beta$.
- Output $k_0 = ((c_1^\ell, c_2^\ell)_{\ell=1}^\lambda, r), k_1 = ((c_2^\ell, c_0^\ell)_{\ell=1}^\lambda, r), k_2 = ((c_0^\ell, c_1^\ell)_{\ell=1}^\lambda, r)$.

$\mathsf{Eval}_i(k_i = ((c_a^\ell, c_b^\ell)_{\ell=1}^\lambda, r), x)$:

- For $\ell = 1, \ldots, \lambda$ set

$$y_i^\ell := \mathsf{Conv}(c_a^\ell, c_b^\ell, x),$$

  and denote by $y_i \in \mathbb{F}_4^\lambda$ the vector of all y_i^ℓ values concatenated.
- Compute and output $\langle r, y_i \rangle$.

Figure 2 3-server statistically secure MV-based DPF.

Security. Denote by $D_{\alpha,\beta}$ the distribution of k_0 as outputted by Gen on input λ and $\hat{f}_{\alpha,\beta}$. We will show that for $\alpha_1 \neq \alpha_2$ and β_1, β_2 the distributions $D_1 = D_{\alpha_1,\beta_1}$ and $D_2 = D_{\alpha_2,\beta_2}$ have statistical distance negligible in λ. The claim for k_1 and k_2 follows without loss of generality. It holds by part 2 of Theorem 9 that

$$\begin{aligned} d(D_1, D_2) &= \frac{1}{2} \sum_{c'} \sum_{r'} \left| \Pr_{D_1}[k_0 = c', r = r'] - \Pr_{D_2}[k_0 = c', r = r'] \right| \\ &= \frac{1}{2} \sum_{c'} \Pr_{D_1}[k_0 = c'] \sum_{r'} \left| \Pr_{D_1}[r = r' | k_0 = c'] - \Pr_{D_2}[r = r' | k_0 = c'] \right| \\ &\leq \frac{1}{2} \max_{c'} \sum_{r'} \left| \Pr_{D_1}[r = r' | k_0 = c'] - \Pr_{D_2}[r = r' | k_0 = c'] \right| \\ &\leq \max_{c'} d(D_1|_{k_0=c'}, D_2|_{k_0=c'}). \end{aligned}$$

Therefore, it is sufficient to upper bound the distance between the distributions D_1 and D_2 conditioned on $k_0 = c'$.

Let y be the vector depending on c' in the distribution D_i conditioned on $k_0 = c'$, which is a distribution over the set $\{(c', r') : r' \in \mathbb{F}_4^\lambda\}$. Then, by part 1 of Theorem 9, in this distribution, every y^ℓ attains two possible values with equal probability, either some nonzero value (depending on c' and α) if $\alpha_\ell^* = \alpha$ or zero if $\alpha_\ell^* = N+1$. Therefore, $H_\infty(y) = \lambda$. By applying Lemma 12 we deduce that when $\hat{r} \xleftarrow{\$} \mathbb{F}_4^\lambda$, the joint distribution $(\hat{r}, \langle \hat{r}, y \rangle)$ is $\epsilon := 2^{\frac{2-\lambda}{2}}$-close to $U_{2(\lambda+1)}$. In particular, $\left| \Pr[\langle \hat{r}, y \rangle = \beta_i] - \frac{1}{4} \right| \leq \epsilon$.

Conditioned on $k_0 = c'$, the distribution of r is exactly $\hat{r}|_{\langle \hat{r}, y\rangle = \beta_i}$. Hence, for a value $w := \Pr[\langle \hat{r}, y\rangle = \beta_i] - \frac{1}{4}$, $-\epsilon \leq w \leq \epsilon$, we arrive at

$$\begin{aligned}
d(\hat{r}|_{\langle \hat{r}, y\rangle = \beta_i}, U_{2\lambda}) &= \sup_{E \subseteq \{(r', \beta_i) : r' \in \mathbb{F}_4^\lambda\}} \left| \frac{\Pr[(\hat{r}, \langle \hat{r}, y\rangle) \in E]}{\frac{1}{4} + w} - \frac{|E|}{4^\lambda} \right| \\
&\leq 4 \sup_{E \subseteq \mathbb{F}_4^{\lambda+1}} \left| \frac{\Pr[(\hat{r}, \langle \hat{r}, y\rangle) \in E]}{1 + 4w} - \frac{|E|}{4^{\lambda+1}} \right| \\
&\leq 4 \sup_{E \subseteq \mathbb{F}_4^{\lambda+1}} \left| \Pr[(\hat{r}, \langle \hat{r}, y\rangle) \in E] - \frac{|E|}{4^{\lambda+1}} \right| + 16|w| + O(|w|^2) \\
&= 4d((\hat{r}, \langle \hat{r}, y\rangle), U_{2(\lambda+1)}) + 16\epsilon + O(\epsilon^2) \\
&= 20\epsilon + O(\epsilon^2),
\end{aligned}$$

which concludes the proof, because if $d(\hat{r}|_{\langle \hat{r}, y\rangle = \beta_i}, U_{2\lambda}) \leq 20\epsilon + O(\epsilon^2)$ in both distributions, then also $d(D_1|_{k_0=c'}, D_2|_{k_0=c'}) \leq 40\epsilon + O(\epsilon^2) \leq 41\epsilon$ by the triangle inequality, and by choosing $\lambda \geq 10$. ◀

The construction from Theorem 10 can be generalized to any prime characteristic, due to the results of [37, 36], from which we get the following.

▶ **Theorem 13** ([37, 36]). *Let p and $p_1 < p_2$ be primes such that either*

- *$p_1, p_2 \neq 2$ and $p \in \{p_1, p_2\}$;*
- *$2 \in \{p_1, p_2\}$ and $p = 2$.*

Then, for $q = p_1 p_2$, there exists a randomized mapping $\mathsf{Share} : [N] \to \mathbb{Z}_q^h \times \mathbb{Z}_q^h \times \mathbb{Z}_q^h$, $h = O\left(\log(q) 2^{2p_2\sqrt{\log N \log\log N}}\right)$, and a deterministic mapping $\mathsf{Conv} : \mathbb{Z}_{2p_2}^h \times \mathbb{Z}_q^h \times [N] \to \mathbb{Z}_p^\ell$, for some constant $\ell = O(q^2)$, such that

1. *For every $\alpha, x \in [N]$,*

$$\Pr\left[(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha) : \sum_{i=0}^{2} \mathsf{Conv}(c_i, c_{(i+1) \bmod 3}, x) \begin{cases} \neq \mathbf{0}, & x = \alpha \\ = \mathbf{0}, & x \neq \alpha \end{cases}\right] = 1.$$

2. *For every $\alpha, \alpha' \in [N]$ and $i, j \in \{0, 1, 2\}$, the distributions of (c_i, c_j), produced by either $(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha)$ or $(c_0, c_1, c_2) \leftarrow \mathsf{Share}(\alpha')$, are identical.*
3. *Share, Conv are computable in time polynomial in their input and output length.*

Utilizing Theorem 13 in similar fashion to how Theorem 9 is used in the proof of Theorem 10, we deduce the following. Note that we require the group size p to be rather small, due to p_2 appearing in the exponent in the expression for h.

▶ **Theorem 14.** *Fix an integer $\lambda > 0$. There exists a statistically $2^{-\Omega(\lambda)}$-secure 3-DPF, for point functions with output group $\mathbb{G} = \mathbb{Z}_p$, where p is a prime, domain size N, and key size $|k_i| = O\left(\lambda \log(p) \cdot 2^{k(p)\sqrt{\log N \log\log N}}\right)$, $i \in \{0, 1, 2\}$, where $k(2) = 6$, $k(3) = 10$, and $k(p) = 2p$ if $p \geq 5$.*

4 Distributed Key Generation

In the standard model for DPF, a client accepts a point (α, β) as input and generates appropriate DPF keys. However, in certain applications of DPF, such as distributed computation of RAM program or MPC with preprocessing for mixed-mode computations, one needs to

accommodate an input (α, β) that is secret-shared among parties that jointly act as client to generate DPF keys, which can then be either locally evaluated or provided to external servers.

We discuss two different settings for distributed key generation: either two clients sharing an input and then jointly generating keys for $m \geq 3$ servers, or all m parties together secret-sharing the input with threshold $t = 1$ and then generating the keys. In either setting it is natural to consider both semi-honest and malicious adversaries.

The point α can be shared in the input in different ways, e.g. secret-sharing each bit separately, or sharing α as an integer value modulo a $N' \geq N$ for domain size N. Generic MPC protocols can be used to switch between these representations with security against malicious adversaries and in time and communication that is linear in the size of the input (for a constant number of servers). Since the input size is negligible in the key length and in the overall communication and computation for distributed key generation we ignore this cost in the rest of the section.

DPF schemes that are based on a family of Matching Vectors such as the schemes in Figures 1, 2 or the scheme in [4] that is based on Frankl's MV family [27] have Gen algorithms that use the following template. Associate the points in the input domain with subsets of a given size w out of a universe of k items. Each point x in the input domain is therefore associated with a binary vector v_x of length k and Hamming weight w, and it must hold that $\binom{k}{w} \geq N$. The vector v_x determines a second binary vector u_x in which each coordinate is a product of a fixed subset of the coordinates of v_x. On input point α the Gen algorithm returns as output u_α. By adapting the discussion in Appendix C of [2] we have that:

▶ **Proposition 15.** *Let $f_{\alpha,\beta} : [N] \to \mathbb{Z}_2$ be a point function and let Share and $h = O\left(2^{6\sqrt{\log N \log\log N}}\right)$ be as in Theorem 9. Choose w to be the smallest integer such that $\binom{w^2}{w} \geq N$, and set $k = w^2$. Then, there exists a Boolean circuit that computes Share with $O(\binom{w^2}{3\sqrt{w}} \cdot 3\sqrt{w} \cdot w^2) = \tilde{O}(h)$ gates and depth $O(\log h)$.*

A circuit to compute the mapping Share can be readily transformed into a circuit that computes Gen for the IT-DPF schemes that we presented. In the 3-server quasi-additive DPF from [4], Gen is identical to Share. In the 3-server statistically secure DPF scheme from Section 3.2, Share is repeated λ times for a statistical security parameter λ and each key is of twice the size of the key from [4] due to CNF sharing of each coordinate in the vector. Therefore, the circuit for Gen is 2λ times the size of the circuit for Share. Finally, in the 4-server scheme of Section 3.1 the circuit size is identical to the circuit size of the 3-server quasi-additive DPF from [4].

The next theorem describes the asymptotic features of using general MPC protocols to securely and distributively generate the keys in the presence of an adversary that corrupts at most one of the servers.

▶ **Theorem 16.** *Let $f_{\alpha,\beta} : [N] \to \mathbb{Z}_2$ be a point function and $h = O\left(2^{6\sqrt{\log N \log\log N}}\right)$. If α and β are secret-shared between $m \geq 2$ servers, for constant m, and the adversary controls at most one server then there exist protocols for distributed key generation for the protocols in Figure 1 and Figure 2 that have the following features:*

- *If $m \geq 3$ then the protocol has information-theoretic security against a malicious adversary using only secure point-to-point channels.*
- *If $m = 2$ then the protocol has information-theoretic security against a malicious adversary in the OT-hybrid model.*
- *The communication and computation costs of the protocol are $\tilde{O}(h)$.*

- *The round complexity is $O(\log h)$; alternatively, the protocols can have constant round complexity if we settle for computational security, while making only a* black-box *use of a pseudorandom generator.*

Proof. General MPC protocols for $m \geq 3$ parties communicating only by point-to-point channels that are information-theoretically secure against a semi-honest adversary that controls at most one party have first been proposed by [6]. Protocols in the same setting that are secure against a malicious adversary were given in [38]. Two-party protocols in the OT-hybrid model, i.e. that are information-theoretically secure in the OT-hybrid model were given in [29, 33, 32].

All of the above protocols have communication and computation cost $\tilde{O}(|C|)$ if the computed function can be realized by a circuit C with $|C|$ gates and their round complexity scales linearly with the circuit depth. Combining these results with the result of Proposition 15 on the size and depth of the circuit to compute key generation gives the information-theoretic variant of the protocol. For the computational case, we can use constant-round protocols based on garbled circuits that make a black-box use of a PRG [1, 21, 34]. ◀

5 Open Questions

We leave open the question of extending our results to general output groups. In particular:

1. Is there a *perfectly secure* 3-server DPF with key size $N^{o(1)}$?
2. Can our results be extended to general Abelian output groups? For the case of $\mathbb{Z}_p$ with an s-bit prime p, we do not know how to construct a DPF with key size $\mathrm{poly}(s) \cdot N^{o(1)}$, even if we allow an arbitrary constant number of servers and settle for statistical security.

We briefly explain the relevant barriers. For the first question, it is not clear how to construct a share conversion that improves upon the one in Theorem 9 by satisfying $\sum_{i=0}^{2} \mathsf{Conv}(c_i, c_{(i+1) \bmod 3}, x) = 1$ whenever $x = \alpha$, instead of just being nonzero. For the second question, the obstacle to obtaining a DPF over $\mathbb{Z}_p$ for a large prime p is that this necessitates the underlying share conversion to operate over characteristic p. For existing share conversion schemes, this requires matching vectors whose length grows super-polynomially with the bit-length of p.

References

1 Donald Beaver, Silvio Micali, and Phillip Rogaway. The round complexity of secure protocols (extended abstract). In Harriet Ortiz, editor, *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*, pages 503–513. ACM, 1990. doi:10.1145/100216.100287.

2 Amos Beimel, Yuval Ishai, Ranjit Kumaresan, and Eyal Kushilevitz. On the cryptographic complexity of the worst functions. In *Theory of Cryptography Conference*, pages 317–342. Springer, 2014.

3 Amos Beimel, Yuval Ishai, and Eyal Kushilevitz. General constructions for information-theoretic private information retrieval. *J. Comput. Syst. Sci.*, 71(2):213–247, 2005.

4 Amos Beimel, Yuval Ishai, Eyal Kushilevitz, and Ilan Orlov. Share conversion and private information retrieval. In *2012 IEEE 27th Conference on Computational Complexity*, pages 258–268. IEEE, 2012.

5 Amos Beimel, Yuval Ishai, Eyal Kushilevitz, and J-F Raymond. Breaking the $O(n^{1/(2k-1)})$ barrier for information-theoretic private information retrieval. In *FOCS 2002*, pages 261–270, 2002.

6 Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In Oded Goldreich, editor, *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pages 351–371. ACM, 2019. doi:10.1145/3335741.3335756.

7 Dan Boneh, Elette Boyle, Henry Corrigan-Gibbs, Niv Gilboa, and Yuval Ishai. Lightweight techniques for private heavy hitters. In *42nd IEEE Symposium on Security and Privacy, SP 2021*, pages 762–776. IEEE, 2021.

8 Elette Boyle, Nishanth Chandran, Niv Gilboa, Divya Gupta, Yuval Ishai, Nishant Kumar, and Mayank Rathee. Function secret sharing for mixed-mode and fixed-point secure computation. In *EUROCRYPT 2021, Part II*, pages 871–900, 2021.

9 Elette Boyle, Geoffroy Couteau, Niv Gilboa, and Yuval Ishai. Compressing vector OLE. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 896–912, 2018.

10 Elette Boyle, Geoffroy Couteau, Niv Gilboa, Yuval Ishai, Lisa Kohl, and Peter Scholl. Efficient pseudorandom correlation generators: Silent OT extension and more. In *CRYPTO 2019*, 2019.

11 Elette Boyle, Geoffroy Couteau, Niv Gilboa, Yuval Ishai, Lisa Kohl, and Peter Scholl. Efficient pseudorandom correlation generators from ring-LPN. In *Annual International Cryptology Conference*, pages 387–416. Springer, 2020.

12 Elette Boyle, Niv Gilboa, and Yuval Ishai. Function secret sharing. In *Eurocrypt 2015*, pages 337–367, 2015.

13 Elette Boyle, Niv Gilboa, and Yuval Ishai. Function secret sharing: Improvements and extensions. In *CCS*, 2016.

14 Elette Boyle, Niv Gilboa, and Yuval Ishai. Secure computation with preprocessing via function secret sharing. In *Theory of Cryptography Conference*, pages 341–371, 2019.

15 Paul Bunn, Jonathan Katz, Eyal Kushilevitz, and Rafail Ostrovsky. Efficient 3-party distributed oram. In *International Conference on Security and Cryptography for Networks*, pages 215–232. Springer, 2020.

16 Benny Chor and Niv Gilboa. Computationally private information retrieval. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 304–313, 1997.

17 Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan. Private information retrieval. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 41–50. IEEE, 1995.

18 Henry Corrigan-Gibbs, Dan Boneh, and David Mazières. Riposte: An anonymous messaging system handling millions of users. In *2015 IEEE Symposium on Security and Privacy*, pages 321–338. IEEE, 2015.

19 Ronald Cramer, Ivan Damgård, Daniel Escudero, Peter Scholl, and Chaoping Xing. Spd$\mathbb{Z}_{2^k}$: Efficient mpc mod 2^k for dishonest majority. In *Annual International Cryptology Conference*, pages 769–798. Springer, 2018.

20 Ronald Cramer, Serge Fehr, Yuval Ishai, and Eyal Kushilevitz. Efficient multi-party computation over rings. In *EUROCRYPT 2003*, pages 596–613, 2003.

21 Ivan Damgård and Yuval Ishai. Constant-round multiparty computation using a black-box pseudorandom generator. In Victor Shoup, editor, *Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, volume 3621 of *Lecture Notes in Computer Science*, pages 378–394. Springer, 2005. doi:10.1007/11535218_23.

22 Samuel Dittmer, Yuval Ishai, Steve Lu, Rafail Ostrovsky, Mohamed Elsabagh, Nikolaos Kiourtis, Brian Schulte, and Angelos Stavrou. Function secret sharing for psi-ca: With applications to private contact tracing. *arXiv preprint arXiv:2012.13053*, 2020.

23 Jack Doerner and Abhi Shelat. Scaling ORAM for secure computation. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 523–535, 2017.

24 Zeev Dvir and Sivakanth Gopi. 2-server PIR with sub-polynomial communication. In Rocco A. Servedio and Ronitt Rubinfeld, editors, *STOC 2015*, pages 577–584. ACM, 2015.

25 Klim Efremenko. 3-query locally decodable codes of subexponential length. *SIAM Journal on Computing*, 41(6):1694–1703, 2012.

26 Ingerid Fosli, Yuval Ishai, Victor I. Kolobov, and Mary Wootters. On the download rate of homomorphic secret sharing. In *ITCS*, 2022.

27 Peter Frankl. Constructing finite sets with given intersections. *Combinatorial mathematics (Marseille-Luminy, 1981)*, pages 289–291, 1983.

28 Niv Gilboa and Yuval Ishai. Distributed point functions and their applications. In *EUROCRYPT*, 2014.

29 Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game, or a completeness theorem for protocols with honest majority. In Oded Goldreich, editor, *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pages 307–328. ACM, 2019. doi:10.1145/3335741.3335755.

30 Vince Grolmusz. Superpolynomial size set-systems with restricted intersections mod 6 and explicit ramsey graphs. *Combinatorica*, 20(1):71–86, 2000.

31 Matthew M Hong, Yuval Ishai, Victor I Kolobov, and Russell WF Lai. On computational shortcuts for information-theoretic pir. In *Theory of Cryptography Conference*, pages 504–534. Springer, 2020.

32 Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding cryptography on oblivious transfer–efficiently. In *Annual international cryptology conference*, pages 572–591. Springer, 2008.

33 Joe Kilian. Founding cryptography on oblivious transfer. In Janos Simon, editor, *STOC 1988*, pages 20–31, 1988.

34 Yehuda Lindell and Benny Pinkas. An efficient protocol for secure two-party computation in the presence of malicious adversaries. In Moni Naor, editor, *Advances in Cryptology - EUROCRYPT 2007, 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain, May 20-24, 2007, Proceedings*, volume 4515 of *Lecture Notes in Computer Science*, pages 52–78. Springer, 2007. doi:10.1007/978-3-540-72540-4_4.

35 Zachary Newman, Sacha Servan-Schreiber, and Srinivas Devadas. Spectrum: High-bandwidth anonymous broadcast. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*, pages 229–248, 2022.

36 Anat Paskin-Cherniavsky and Olga Nissenbaum. New bounds and a generalization for share conversion for 3-server PIR. *Entropy*, 24(4), 2022. doi:10.3390/e24040497.

37 Anat Paskin-Cherniavsky and Leora Schmerler. On share conversions for private information retrieval. *Entropy*, 21(9):826, 2019.

38 Tal Rabin and Michael Ben-Or. Verifiable secret sharing and multiparty protocols with honest majority. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 73–85, 1989.

39 Ni Trieu, Kareem Shehata, Prateek Saxena, Reza Shokri, and Dawn Song. Epione: Lightweight contact tracing with strong privacy. *arXiv preprint arXiv:2004.13293*, 2020.

40 Sergey Yekhanin. Towards 3-query locally decodable codes of subexponential length. *Journal of the ACM (JACM)*, 55(1):1–16, 2008.