Electronic Colloquium on Computational Complexity, Report No. 189 (2017)

Conditional Disclosure of Secrets and d-Uniform Secret Sharing with Constant Information Rate*

Benny Applebaum†, Barak Arkis†

December 25, 2017

Abstract

Consider the following secret-sharing problem. Your goal is to distribute a long file s between n servers such that $(d-1)$-subsets cannot recover the file, $(d+1)$-subsets can recover the file, and d-subsets should be able to recover s if and only if they appear in some predefined list L. How small can the information ratio (i.e., the number of bits stored on a server per each bit of the secret) be?

We initiate the study of such d-uniform access structures, and view them as a useful scaled-down version of general access structures. Our main result shows that, for constant d, any d-uniform access structure admits a secret sharing scheme with a *constant* asymptotic information ratio of c_d that does not grow with the number of servers n. This result is based on a new construction of d-party Conditional Disclosure of Secrets (Gertner et al., JCSS '00) for arbitrary predicates over n-size domain in which each party communicates at most four bits per secret bit.

In both settings, previous results achieved non-constant information ratio which grows asymptotically with n, even for the simpler (and widely studied) special case of $d = 2$. Moreover, our results provide a unique example for a natural class of access structures $\mathcal{F}$ that can be realized with information rate smaller than its bit-representation length $\log |\mathcal{F}|$ (i.e., $\Omega(d \log n)$ for d-uniform access structures) showing that *amortization can beat the representation size barrier*.

Our main result applies to exponentially long secrets, and so it should be mainly viewed as a barrier against amortizable lower-bound techniques. We also show that in some natural simple cases (e.g., low-degree predicates), amortization kicks in even for quasi-polynomially long secrets. Finally, we prove some limited lower-bounds, point out some limitations of existing lower-bound techniques, and describe some applications to the setting of private simultaneous messages.

1 Introduction

Secret sharing schemes (SS), introduced by [Sha79, Bla79], are a central cryptographic tool with a wide range of applications (see [Bei11] and references therein). In its general form, an n-party secret sharing scheme for a family of authorized sets $\mathcal{A} \subseteq 2^{[n]}$ (referred to as *access structure*) allows to distribute a secret $s \in \mathcal{S}$ into n shares, $s_1, \ldots, s_n$, one for each party, such that: (1) every authorized set of parties, $A \in \mathcal{A}$, can reconstruct s from its shares; and (2) every unauthorized set

*Research supported by the European Union's Horizon 2020 Programme (ERC-StG-2014-2020) under grant agreement no. 639813 ERC-CLC, and the Check Point Institute for Information Security.

†Tel-Aviv University, bennyap@post.tau.ac.il, barakark@mail.tau.ac.il

ISSN 1433-8092

of parties A not in $\mathcal{A}$ cannot reveal any partial information on the secret even if the parties are computationally unbounded. A canonical example is the case of threshold secret-sharing in which $\mathcal{A}$ contains all the sets whose cardinality is at least a certain threshold. For this case, Shamir's scheme [Sha79] provides an optimal solution since each party gets a share whose length equals to the length of the secret s which is the best that one can hope for.

It is known that any monotone access structure $\mathcal{A}$ admits a secret sharing scheme [ISN87].[1] However, the communication complexity of general access structures has remained wide open. It is known that the *information ratio*, $\max_i |s_i|/|s|$, of an access structure is at most polynomial in the representation size of $\mathcal{A}$ as a monotone formula [BL88] or as a monotone span program [KW93]. This leads to an exponential upper-bound of $2^{\Omega(n)}$ for any $\mathcal{A}$. On the other hand, despite much efforts, the best known lower-bound on the information ratio of an n-party access structure is $\Omega(n/\log n)$ due to [Csi97]. It is widely believed that some access structures require exponential information rate [Bei11]; however, proving any super-linear lower-bound (even for a non-explicit access structure) has remained an intriguing open problem.

Given this state of affairs, it makes sense to focus on less general access structures. In this paper, we consider two such (related) settings: the family of *d-uniform* access structures and access structures that correspond to *Conditional Disclosure of Secrets*.

1.1 Uniform Access Structures

A d-uniform access structure $\mathcal{A}$ is represented by a d-uniform hypergraph G over $[n]$ and has the following semantics:

- All sets of $d+1$ parties (or more) are authorized.
- All sets of $d-1$ parties (or less) are unauthorized.
- A set of size d is authorized if it appears as an hyperedge in G.

The family of *d-uniform* access structures is rich enough to capture an arbitrary relation on d-size sets. By focusing on a constant d (which does not grow with the number of parties n), we get a scaled-down "toy" version of the more general problem of arbitrary access structures.

Previous works. The case of $d = 2$ was presented by Sun and Shieh [SS97] under the terminology of graph forbidden access structure and was further studied in several works. For single-bit secrets and linear schemes (in which the secret is viewed as a field element and each share can be written as a linear combination of the secret and several independent random field elements), we know that an information ratio of $\Theta(\sqrt{n})$ is both sufficient [BIKK14, GKW15] and necessary [BFMP17, Min12] for 2-uniform access structures. Very recently, it was shown in [LVW17] that a non-linear scheme can achieve a sub-polynomial information ratio of $2^{O(\sqrt{\log n \log\log n})}$. If the secret is sufficiently long (exponential in n), then the information ratio can be further reduced to $O(\log n)$ as follows from [AARV17]. At the same paper, it was shown that some (non-explicit) 2-uniform access structures require an information ratio of $\Omega(\log n)$ for a single-bit secret. We further note the logarithmic bound matches (up to a constant factor) the information-theoretic representation length

[1] Monotonicity here means that for any $A \subset B$ it holds that $A \in \mathcal{A} \Rightarrow B \in \mathcal{A}$. It is not hard to see that a non-monotone access structure does not admit an SS, and therefore this requirement is necessary.

of 2-uniform access structure (since any such access structure can be represented by $2 \log n$ bits), which may be viewed as a natural candidate for information ratio lower-bound.

Our contribution. We show that the asymptotic information ratio (for sufficiently long secrets) of any d-uniform access structure can be reduced to a constant.

Theorem 1.1. *Any n-party d-uniform access structure $\mathcal{A}$ can be realized by a secret sharing scheme that achieves a constant information ratio of $c_d \leq 6\frac{d^d+1}{d!} \leq O(e^d)$ for sufficiently long secrets.*

To the best of our knowledge, this is the first result that realizes a natural class of access structures $\mathcal{F}$ with information rate smaller than its bit-representation length $\log |\mathcal{F}|$ (i.e., $\Omega(d \log n)$ for d-uniform access structures).

The theorem (whose proof appears in Section 4) yields a *multilinear* SS, namely, the secret is viewed as a vector of field elements and each share can be written as a linear combination of the secret and several independent random field elements. Although we did not try to optimize the constant c_d, we mention that, for the special case of $d = 2$, we get an information ratio c_d of at most 12.5.

Unfortunately, amortization kicks in very late, only for secrets of length exponential in n^d. As a result, the scheme seems hardly useful for positive applications. Indeed, we view Theorem 1.1 mainly as a *barrier* for proving secret-sharing lower-bounds: One cannot prove a super-constant lower-bound on the information ratio of d-uniform access structure by techniques that "fail to distinguish" between short secrets and very long secrets.

For example, information theoretic based arguments typically apply to amortized complexity as well, and therefore seem hard to employ in this context. In Section 5, we further show that a standard information-theoretic method [CSGV93, KGH83] based on Shannon's information inequalities cannot prove a lower-bound better than d for d-uniform access structures. We mention that non-amortized lower-bounds can be proven for linear schemes (based on dimensionality arguments). Indeed, we observe that the lower-bound of [BFMP17, Min12] for 2-uniform linear schemes yields a lower-bound of $\Omega_d(n^{(d-1)/2})$ for linear SS over 1 bit secrets. This result also implies that the amortization point of any multilinear scheme (like in Theorem 1.1) must be at least polynomial in n. (See Section 5 for details.)

1.2 Conditional Disclosure of Secrets

The proof of Theorem 1.1 is based on a new construction of Conditional Disclosure of Secrets (CDS) [GIKM00]. In this model, Alice and Bob hold a shared secret s and private inputs x and y, respectively, and they wish to let Carol learn the secret s if and only if the inputs (x, y) satisfy some predefined predicate $f : X \times Y \to \{0, 1\}$. The inputs x, y are known to Carol, and, in addition, she gets a single message, a, from Alice and a single message, b, from Bob. These messages depend on the party's input, on the secret s, and on a random string r which is shared between Alice and Bob but is hidden from Carol. Given (a, b, x, y), Carol should be able to recover s if $f(x, y) = 1$ but should learn nothing on the secret otherwise. The parties are assumed to be computationally unbounded, and the goal is to minimize the communication complexity of Alice and Bob. (See Section 2 for a formal definition.)

CDS schemes have found useful applications in various contexts such as information-theoretically private information retrieval protocols [CKGS98], priced oblivious transfer [AIR01], and attribute

based encryption [GPSW06, SW05]. Focusing on the last application, it turns out that the communication complexity of CDS for natural predicates is tightly connected to the parameters (private-key/ciphertext length) achievable by natural constructions of attribute based encryption. (See the discussion in [GKW15].) As a result, the communication complexity of CDS has recently attracted a noticeable amount of research.

CDS as a secret sharing. CDS can be viewed as a (simpler) variant of 2-uniform access structure. Specifically, consider an access structure over the set of players $X \times Y$ in which every pair of parties $(x, y) \in X \times Y$ should be able to recover the secret s if and only if $f(x, y) = 1$. We further assume that singletons are not authorized, but other than that we do not require any privacy/correctness condition for other subsets of parties. Then, we can represent the secret-sharing problem as the problem of realizing a CDS for the predicate f and vice-versa by setting the share of the x-th player (resp., y-th player) to be the message $a(x, s; r)$ (resp., $b(y, s; r)$). The communication complexity of the CDS protocol therefore corresponds to the maximal size of the shares.

The worst-case complexity of CDS (over all predicates $f : [n] \times [n] \to \{0, 1\}$) matches, up to a constant multiplicative factor, the complexity of the worst-case 2-uniform SS over $2n$ players (as shown implicitly in [BIKK14]).[2] In particular, for single bit secrets, the best known communication complexity is sub-polynomial in the domain size [LVW17], and for exponentially long secrets the best upper-bound on the information ratio (i.e., communication divided by the length of the secret) is logarithmic in n [AARV17]. (In fact, these results were first established for the CDS setting and then were exported to the more general 2-uniform setting via [BIKK14].)

Our contribution. We prove that any predicate admits a CDS with asymptotic information ratio of 4. Moreover, this result applies to multiparty CDS where Alice and Bob are replaced with k parties. (See Section 2 for formal definitions.)

Theorem 1.2. *Any k-party predicate $f : X_1 \times \ldots \times X_k \to \{0, 1\}$ admits a k-party* CDS *in which, for sufficiently large secrets, each party communicates at most 4 bits per each bit of the secret. For the special case of $k = 2$, the information ratio can be improved to 3.*

Given Theorem 1.2 we derive Theorem 1.1 by extending the transformation of [BIKK14] to the multiparty setting. (See Section 4.)

Theorem 1.2 is proved in Section 3 by strengthening the amortization techniques of [AARV17]. In particular, Applebaum et al. reduce the problem of amortizing the complexity of two-party CDS to the problem of constructing a two-party *batch*-CDS scheme. In the latter setting Alice holds a single input x, Bob holds a single input y, and both parties hold 2^{2n} secrets, one for each predicate in $\mathcal{F} = \{f : [n] \times [n] \to \{0, 1\}\}$. The scheme releases the secret s_f if and only if f evaluates to 1 on (x, y). In [AARV17] such a scheme is realized by recursing over the inputs (x, y) in a bit-by-bit manner. Loosely speaking, once Alice knows that the last bit of x is, say, zero, she can complete the task by invoking a batch-CDS for the residual functions $\mathcal{G} = \{g : [n/2] \times [n] \to \{0, 1\}\}$ with random secrets r_g and release $s_f \oplus r_g$. In fact, many functions f will be simplified to the same $g \in \mathcal{G}$, and therefore, in order to deliver the secret s_f for each such f, Alice will have to use many copies of g with a different secret $r_{g,i}$ for each copy. The crucial point is that each $g \in \mathcal{G}$ accounts

[2] The reader should note that CDS complexity is sometimes measured in terms of the bit-length of the x and y (i.e., $\log |X| + \log |Y|$). In our context it is more natural to use the cardinality of the alphabet as the main parameter.

for the same number $D = |\mathcal{F}|/|\mathcal{G}|$ of $f \in \mathcal{F}$, and so we can use D copies of batch-CDS over $\mathcal{G}$. This bit-by-bit recursion leads to a batch-CDS with communication complexity of $O(|\mathcal{F}| \log n)$, and the logarithmic overhead is carried over to the setting of amortized CDS for long secrets.

In order to get rid of this overhead, we modify the construction of batch-CDS, and instead of treating Alice's inputs in a bit-by-bit manner, we treat it as a single element from $[n]$. Abstracting the above argument, the transformation works as long as each residual function g over Bob's inputs accounts for the same number of original functions in $\mathcal{F}$. We further abstract this property of $\mathcal{F}$ and extend the argument to k parties (recursing over the parties instead of the bits of the inputs). This allows us to shave the logarithmic factor and to obtain a constant overhead for any function family $\mathcal{F}$ that satisfies some regularity and closure conditions. (See Section 3.1 for details.)

These results are used to obtain multilinear CDS for any predicate f in $\mathcal{F}$ with information ratio of at most 4 as long as the secret is larger than $|\mathcal{F}|$. Taking $\mathcal{F}$ to be the class of all predicates (which is shown to satisfy the required conditions) we derive Theorem 1.2. In this case, amortization kicks in only when the secret is exponential in the domain size of f. This can be significantly improved when f is taken from a small family $\mathcal{F}$ of predicates that satisfies our conditions. For example, we show that when f is a low-degree multivariate polynomial amortization kicks in even for secrets of length quasi-polynomial in the size of the domain. (See Section 3 for details.)

From CDS to partial PSM. Finally, we ask whether highly efficient CDS protocols can be used to improve the complexity of more challenging tasks such as *Private Simultaneous Message Protocols* [FKN94]. This setting is similar to the CDS setting except that here, the inputs x, y are treated as private data (not known to Carol), and the goal is to let Carol learn the function $f(x, y)$ without learning any additional information. (The communication pattern is one-way just as the case of CDS.) This setting is much more challenging (just like functional encryption is more challenging than attribute based encryption). For an arbitrary function $f : [n] \times [n] \to \{0, 1\}$, the best upper-bound is $O(\sqrt{n})$ [BIKK14] and no amortization results are known.

Following [IW14], we consider a hybrid model (partial PSM) in which Alice's input x is partitioned into a public part x_1 which is known to Carol (but not to Bob) and to a private part x_2, and similarly Bob's input, y, is partitioned into a public part y_1 (known to Carol but not to Alice) and a private part y_2. Trivially, partial PSM complexity is upper bounded by PSM complexity in the sense that one can apply a PSM protocol to hide all of Alice's and Bob's input (both the private and public parts). Adapting known PSM protocols to the partial PSM model in a way that communication complexity is reduced, does not seem like an easy task. As explained in Section 6, CDS turns out to be a natural tool for accomplishing this task. In Section 6 we reduce partial PSM to CDS with an overhead which is roughly linear in the domain of the private input. (We obtain better results for families of predicates which can be computed by small/shallow Boolean circuits.) Our results improve upon the reduction of [AARV17] whose overhead is exponential in the domain of the private parts.

Acknowledgements

We thank Amos Beimel for helpful discussions.

2 Definitions

In this section we define *Secret-Sharing*, *multiparty* CDS, and *partial*-PSM. In all of our definitions, we consider only perfect correctness and perfect privacy. (Relaxations to the case of imperfect privacy and imperfect correctness can be obtained in a natural manner.)

2.1 Secret-sharing

The following definitions are based on [Bei11].

Access structures and distribution schemes. Let $p_1, ..., p_n$ be a set of parties. A collection $\mathcal{A} \subset 2^{\{p_1,...,p_n\}}$ is monotone if $B \in A$ and $B \subset C$ imply that $C \in \mathcal{A}$. An access structure is a monotone collection $\mathcal{A} \subset 2^{\{p_1,...,p_n\}}$ of non-empty subsets of $\{p_1, ..., p_n\}$. Sets in $\mathcal{A}$ are called authorized, and sets not in $\mathcal{A}$ are called unauthorized. A distribution scheme $\Sigma = (\Pi, \mu)$ with domain of secrets $\mathcal{S}$ is a pair, where μ is a probability distribution on some finite set $\mathcal{R}$ called the set of random strings and Π is a mapping from $\mathcal{S} \times \mathcal{R}$ to a set of n-tuples $\mathcal{Z}_1 \times \mathcal{Z}_2 \times \ldots \times \mathcal{Z}_n$, where $\mathcal{Z}_j$ is called the domain of shares of p_j. A dealer distributes a secret $s \in S$ according to Σ by first sampling a random string $r \in \mathcal{R}$ according to μ, computing a vector of shares $\Pi(s, r) = (z_1, ..., z_n)$, and privately communicating each share z_j to party p_j. For a set $A \subset \{p_1, \ldots, p_n\}$, we denote $\Pi(s, r)_A$ as the restriction of $\Pi(s, r)$ to its A-entries. The *information ratio* of a distribution scheme is $max_{1 \le j \le n} \frac{\log |\mathcal{Z}j|}{\log |\mathcal{S}|}$.

Definition 2.1 (Secret Sharing)**.** *Let $\mathcal{S}$ be a finite set of secrets, where $|\mathcal{S}| \geq 2$. A distribution scheme (Π, μ) with domain of secrets $\mathcal{S}$ is a secret-sharing scheme realizing an access structure $\mathcal{A}$ if the following two requirements hold:*

- ***Correctness.*** *For every authorized set $B \in \mathcal{A}$ (where $B = \{p_{i_1}, \ldots, p_{i_{|B|}}\}$), there exists a reconstruction function $\mathsf{Rec}_B : \mathcal{Z}_{i_1} \times \ldots \times \mathcal{Z}_{i_{|B|}} \to \mathcal{S}$ such that for every $s \in \mathcal{S}$,*

$$\Pr[Recon_B(\Pi(s, r)_B) = s] = 1.$$

- ***Privacy.*** *For any unauthorized set $T \notin \mathcal{A}$, every two secrets $a, b \in \mathcal{S}$, the random variables*

$$\Pi(a, r)_T \qquad \text{and} \qquad \Pi(b, r)_T,$$

induced by sampling r according to μ, are identically distributed.

A secret sharing scheme is *linear* (resp., *multilinear*) over a finite field $\mathbb{F}$, if the secret domain $\mathcal{S}$ is $\mathbb{F}$ (resp., $\mathbb{F}^i$ for some $i \geq 1$), the randomness domain $\mathcal{R}$ is $\mathbb{F}^j$ for some $j \geq 1$, and the mapping Π is linear over $\mathbb{F}$. By default, we always assume that the domain $\mathcal{S}$ can be associated with some finite field.

Uniform access structures. Our main focus will be on *Uniform Access Structures*. Formally, an access structure $\mathcal{A}$ is *d-uniform* if every authorized set of $\mathcal{A}$ is of size at least d, and every set of size at least $d+1$ is authorized. A secret-sharing scheme for a d-uniform access structure is referred to as a d-uniform secret sharing scheme.

2.2 Conditional disclosure of secrets

Definition 2.2 (multiparty CDS)**.** *Let $f : \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \{0,1\}$ be a predicate. For $1 \leq i \leq k$ let $F_i : \mathcal{X}_i \times \mathcal{S} \times \mathcal{R} \to \mathcal{Z}_i$ be deterministic encoding algorithms ($\mathcal{S}$ is the secret domain and $\mathcal{R}$ is the shared randomness domain). We say that the tuple $(F_1, \ldots, F_k)$ is a k-party* CDS *for f, if the function $F(x_1, \ldots, x_k, s, r) = (F_1(x_1, s, r), \ldots, F_k(x_k, s, r))$ satisfies the following conditions:*

- ***Correctness.*** *There exists a deterministic algorithm* Dec*, called the decoder, such that for every input $(x_1, \ldots, x_k)$ such that $f(x_1, \ldots, x_k) = 1$, every secret $s \in \mathcal{S}$, and every random string $r \in \mathcal{R}$ we have that*

$$\mathsf{Dec}(x_1, \ldots, x_k, F(x_1, \ldots, x_k, s, r)) = s.$$

- ***Privacy.*** *There exists a randomized simulator* Sim *such that for every $(x_1, \ldots, x_k)$ such that $f(x_1, \ldots, x_k) = 0$ and any secret $s \in \mathcal{S}$ the random variables*

$$F(x_1, \ldots, x_k, s, r) \qquad \text{and} \qquad \mathsf{Sim}(x_1, \ldots, x_k),$$

  induced by a random choice of $r \in \mathcal{R}$ and a uniform choice of the internal randomness of the simulator, are identically distributed.

The communication complexity *of party i is $\log(|\mathcal{Z}_i|)$ and its* amortized communication *complexity (or* information ratio*) is $\frac{\log(|\mathcal{Z}_i|)}{\log(|\mathcal{S}|)}$. The information ratio of the protocol is the maximum information ratio of all parties.*

A important property of CDS is whether or not it is linear. We distinguish between linear CDS and multilinear CDS. A multiparty CDS is *multilinear* over a finite field $\mathbb{F}$ if:

1. The secret and the randomness domains are both vectors over $\mathbb{F}$.
2. The encoding functions F_i are linear in the secret and randomness. That is, fixing the input x_i, F_i's output is a vector over $\mathbb{F}$ in which every coordinate is a linear combination of the secret and the random field elements.

A multilinear CDS is *linear* if the secret is a *single* field element (i.e., $\mathcal{S} = \mathbb{F}$). By default, we always assume that the domain $\mathcal{S}$ can be associated with some finite field. To simplify notation, we will use the term CDS instead of multiparty CDS when the number of parties is clear from the context.

Remark 2.3. *It is sometimes useful to consider a variant of* CDS *in which only a single party (say the last one) holds the secret. Formally, this means that F_k depends on the secret (and randomness) and $F_1, \ldots, F_{k-1}$ depend only in the randomness. Being a special case of the original definition, any construction of this variant of* CDS*, also satisfies the general notion of* CDS*. We mention that all the constructions in this paper natively admit a* CDS *in which only the last party holds the secret. More generally, it is not hard to turn any standard* CDS *into a single-party-holds-the-secret type with a minor loss of $|s|$ in the total communication complexity. Indeed, one can just run the standard* CDS *with a random secret s', and let the last party send, in addition, the value $s + s'$.*

2.3 Partial simultaneous message protocols

Lastly, we define a variant of PSM called *partial*-PSM which adopts the notion of partial garbling [IW14] to the three-party setting of [FKN94].

Definition 2.4 (partial-PSM)**.** *Let $f : (\mathcal{X} \times \mathcal{W}) \times (\mathcal{Y} \times \mathcal{T}) \to \{0,1\}$ be a function. We say that a pair of deterministic encoding algorithms $F_1 : (\mathcal{X} \times \mathcal{W}) \times \mathcal{R} \to \mathcal{Z}_1$ and $F_2 : (\mathcal{Y} \times \mathcal{T}) \times \mathcal{R} \to \mathcal{Z}_2$ are partial-*PSM *for f if the function $F(x,w,y,t,r) = (F_1(x,w,r), F_2(y,t,r))$ that corresponds to the joint computation of F_1 and F_2 on a common r, satisfies the following properties:*

- ***Correctness.*** *There exists a deterministic algorithm* Dec*, called the decoder, such that for every input (x,w,y,t) and every $r \in \mathcal{R}$ we have that*

$$\mathsf{Dec}(w,t,F(x,w,y,t,r)) = f(x,w,y,t).$$

- ***Privacy.*** *There exists a randomized algorithm (simulator)* Sim *such that for any input (x,w,y,t) the random variables*

$$F(x,w,y,t,r) \qquad \text{and} \qquad \mathsf{Sim}(w,t,f(x,w,y,t)),$$

  induced by a random choice of $r \in \mathcal{R}$ and a uniform choice of the internal randomness of the simulator, are identically distributed.

We refer to $\mathcal{X}$ and $\mathcal{Y}$ as the private domain *of f, and to $\mathcal{W}$ and $\mathcal{T}$ as the* public domain *of f. When the public domain is empty, we get the standard definition for* PSM *(as all input is required to be hidden). The* communication complexity *of the protocol is defined as the total encoding length $(\log|\mathcal{Z}_1| + \log|\mathcal{Z}_2|)$, and the* randomness complexity *is defined as the length $\log|\mathcal{R}|$ of the common randomness.*

Remark 2.5 (PSM as randomized encoding of functions)**.** *A* PSM *protocol for f can be alternatively viewed as a special type of* randomized encoding *[IK00, AIK06] of f, where the output of f is encoded by the output of a randomized function $F((x,y),r)$ such that F can be written as $F((x,y),r) = (F_1(x,r), F_2(y,r))$. This is referred to as a "2-decomposable" encoding in [Ish13]. Similarly, the notion of partial* PSM *can be derived by considering 2-decomposable* partial encoding *(or garbling).*

3 Constant information ratio for CDS

In this section we show that, for sufficiently long secrets, any d-ary predicate f admits a d-party CDS with constant information ratio. Following [AARV17], we begin (in Section 3.1) by constructing a highly efficient *batch* version of CDS (that simultaneously handles a class of different predicates) and then show (in Section 3.2) how to transform it into a standard CDS with low amortized complexity.

3.1 Batch-CDS and Regular Function Families

A k-party *batch*-CDS for a class of predicates $\mathcal{F}$ takes as an input a vector of secrets $(s_f)_{f\in\mathcal{F}}$ and a single input tuple $x = (x_1, \ldots, x_k)$ where x_i belongs to the i-th party, and delivers to Carol all the secrets s_f for which $f(x) = 1$.

Definition 3.1 (batch-CDS [AARV17]). *Let $\mathcal{F} = (f_1, \ldots, f_m)$ be an m-tuple of predicates over the domain $\mathcal{X}_1 \times \ldots \times \mathcal{X}_k$. For $i \in [k]$ let $F_i : \mathcal{X}_i \times \mathcal{S}^m \times \mathcal{R} \to \mathcal{Z}_i$ be deterministic encoding algorithms, where $\mathcal{S}$ is the secret domain. Then, $(F_1, \ldots, F_k)$ is a k-party batch-CDS scheme for $\mathcal{F}$ if the function $F(x, y, s, r) = (F_1(x_1, s, r), \ldots, F_k(x_k, s, r))$, where $s \in \mathcal{S}^m$, satisfies the following properties:*

1. ***Correctness.*** *There exists a deterministic algorithm Dec, called a decoder, such that for every $i \in [m]$, every input $x = (x_1, \ldots, x_k)$ which satisfies f_i and every vector of secrets $s \in \mathcal{S}^m$, we have that:*
$$\Pr_{r \xleftarrow{R} \mathcal{R}} [\mathsf{Dec}(i, x, y, F(x, y, s, r)) = s_i] = 1.$$
2. ***Privacy.*** *There exists a simulator Sim such that for every input $x = (x_1, \ldots, x_k)$ and every vector of secrets $s \in \mathcal{S}^m$, the following distributions are identical*
$$\mathsf{Sim}(x, \hat{s}) \qquad \text{and} \qquad F(x, s, r),$$
where $r \xleftarrow{R} \mathcal{R}$ and $\hat{s}$ is an m-long vector whose i-th component equals to s_i if $f_i(x, y) = 1$, and $\perp$ otherwise.

The communication complexity *of the party i is* $\log |\mathcal{Z}_i|$.

We generalize the ideas of [AARV17] and show that every family of functions that satisfy some closure properties (detailed in Definition 3.2) admits a highly efficient batch-CDS.

Definition 3.2 (regular function family). *Let $\mathcal{X}_1, \ldots, \mathcal{X}_k$ be a tuple of input domains and let $\mathcal{F} = (\mathcal{F}_1, \ldots, \mathcal{F}_k)$ be a sequence of function families where, for every i, the family $\mathcal{F}_i$ contains functions of the form $f : \mathcal{X}_1 \times \ldots \times \mathcal{X}_i \to \{0, 1\}$. We say that $\mathcal{F}$ is* regular *if it satisfies the following conditions:*

1. *$\mathcal{F}$ is closed under addition. That is, for every $i \in [k]$ and $f_1, f_2 \in \mathcal{F}_i$, we have that $f_1 + f_2 \in \mathcal{F}_i$ (addition is over the binary field).*
2. *For every $i \in [k]$, $\mathcal{F}_i$ contains the constant function 1*
3. *For every $i \in [k-1]$ and every function $g \in \mathcal{F}_i$ and $a \in \mathcal{X}_{i+1}$, let $R(g, a)$ be the set of functions $f \in \mathcal{F}_{i+1}$ which simplify to g when their last input is substituted by a. (That is, $f(x_1, \ldots, x_i, a) = g(x_1, \ldots, x_i)$ for every $(x_1, \ldots, x_i) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_i$). Then the size of $R(g, a)$ is independent of g and a, and depends only on the arity i. We let R_i denote this size.*

Remark 3.3. *It is useful to think of the last property of Definition 3.2 in graph-theoretic terms. Consider a k-layered graph in which the i-th layer contains a node for every function $f \in \mathcal{F}_i$, and add an edge, labeled by $a \in \mathcal{X}_{i+1}$, from $f \in \mathcal{F}_{i+1}$ to $g \in \mathcal{F}_i$ if $f(\cdots, a)$ simplifies to g. Then, each layer i should be regular in the sense that, for every edge label $a \in \mathcal{X}_{i+1}$, every node $f \in \mathcal{F}_i$ has exactly R_i incoming edges which are labeled by a. (This, in particular, implies that $|\mathcal{F}_{i+1}| = R_i |\mathcal{F}_i|$.)*

An important example of a regular function family is the family of all functions.

Proposition 3.4. *Let $\mathcal{X}_1, \ldots, \mathcal{X}_k$ be a sequence of finite sets, and let $\mathcal{F}_i$ denote the family of all predicates over $\mathcal{X}_1 \times \ldots \times \mathcal{X}_i$. Then the family $\mathcal{F} = (\mathcal{F}_i)_{i \in [k]}$ is regular.*

Proof. Properties (1) and (2) clearly hold. To prove the third property, fix $i \in [k-1]$, $g \in \mathcal{F}_i$ and $a \in \mathcal{X}_{i+1}$, and observe that there is a 1-1 correspondence between functions in $R(g,a)$ to the set of all binary functions over the domain $\mathcal{X}_1 \times \ldots \times \mathcal{X}_i \times (\mathcal{X}_{i+1} \setminus \{a\})$. Since the number of such functions is independent of a and g, the proposition follows. □

Another regular function family is polynomials of degree at most D over the binary field.

Proposition 3.5. *Let $(\ell_1, \ldots, \ell_k)$ be a k-tuple of positive integers and let $\mathcal{X}_i = \{0,1\}^{\ell_i}$. For an integer D let $\mathcal{P}_i$ be the family of all functions over $\mathcal{X}_1 \times \ldots \times \mathcal{X}_i$ that can be expressed as multivariate polynomials over the binary field with $\sum_{j=1}^{i} \ell_j$ variables and total degree of at most D. Then the family $\mathcal{P}_{\ell,D} = (\mathcal{P}_i)_{i \in [k]}$ is regular.*

Proof. Conditions (1) and (2) clearly hold. We prove the third property. Fix $i \in [k-1]$. Let $q \in \mathcal{P}_i$ be a polynomial over $L = \sum_{j \le i} \ell_i$ variables $x_{[L]} = (x_1, \ldots, x_L)$ whose (total) degree is at most D, and fix some vector $a \in \{0,1\}^{\ell_{i+1}}$ of binary field elements. We prove that the size of $R(q,a)$ is independent of q and a, by showing that there is a bijection φ from $R(q,a)$ to the set of polynomials over $L' = L + \ell_{i+1}$ variables $x_{[L']} = (x_1, \ldots, x_{L'})$ and degree at most $D' = D - \ell_{i+1}$.

Indeed, any polynomial $p \in R(q,a)$ is a multivariate polynomial over L' variables and degree at most D such that $p(\cdot, a) = q(\cdot)$, and therefore we can write

$$p(x_{[L']}) - q(x_{[L]}) = g(x_{[L']}) \cdot \prod_{j=1}^{\ell_{i+1}} (x_{L+j} - a_j),$$

for some polynomial $g(x_{[L']})$. Since the degree of the LHS is at most D, the degree of g is at most D'. We let $\varphi(p) = g$. To see that this is a bijection, observe that for any polynomial g over $x_{[L']} = (x_1, \ldots, x_{L'})$ and degree at most D', the polynomial $\varphi^{-1}(g)$ defined by $q(x_{[L]}) + g(x_{[L']}) \prod_{j=1}^{\ell_{i+1}} (x_{L+j} - a_j)$ is in $R(q,a)$. The proposition follows. □

We continue by showing that every regular function family has an efficient batch-CDS. From now on, we work with secrets (and randomness) that are taken from some arbitrary finite field $\mathbb{F}$ (e.g., the binary field).

Lemma 3.6. *Let $\mathcal{F} = \{\mathcal{F}_i\}_{i=1}^{k}$ be a regular function family over the input domains $\mathcal{X}_1, \ldots, \mathcal{X}_k$. There is a batch-CDS for $\mathcal{F}_k$ such that the communication of each party consists of at most $|\mathcal{F}_k|$ field elements. Moreover, one of the parties (e.g., the first) communicates only $|\mathcal{F}_k|/2$ field elements.*

Proof. Denote by s_f the secret field element associated with some function $f \in \mathcal{F}_k$. We show (inductively) how to construct a batch-CDS for $\mathcal{F}_k$. For $k = 1$ a single party holds the entire input and can send s_f for every f which satisfies $f(x_1) = 1$, using communication at most $|\mathcal{F}_1|$ field elements. In fact, the regularity conditions (1 and 2) guarantee that exactly half of the functions are satisfied by x_1, and therefore only $|\mathcal{F}_1|/2$ field elements will be sent by the first party.

Let us assume that the claim holds for $k-1$. To extend the protocol to k parties we make use of the following family of mappings. For every $a \in \mathcal{X}_k$ let T_a be an injective mapping that maps a function $f \in \mathcal{F}_k$ to $(g,i) \in \mathcal{F}_{k-1} \times [R_{k-1}]$, such that f is the i-th function in $R(g,a)$ according to some fixed predefined order. (Recall that $f \in R(g,a)$ if $f(\cdot, a) = g(\cdot)$.) By the third regularity condition, $|R(g,a)| = R_{k-1}$ for every g, a, and therefore T_a is well defined. The existence of such mappings T_a gives us the ability to use the batch-CDS inductively:

1. Players $1, \ldots, k-1$ run the batch-CDS for $\mathcal{F}_{k-1}$, R_{k-1} times with random field elements $r_{g,i}$ for $(g,i) \in \mathcal{F}_{k-1} \times [R_{k-1}]$ to release $r_{g,i}$ if and only if $g(x_1, \ldots, x_{k-1}) = 1$.

2. For every function $f \in \mathcal{F}_k$ player k computes $(g,i) = T_{x_k}(f)$ and releases $s_f + r_{g,i}$.

The decoding procedure is simple. If the input $(x_1, \ldots, x_k)$ satisfies $f \in \mathcal{F}_k$, the decoder does the following: (1) Computes $(g,i) = T_{x_k}(f)$ and retrieves the value of $r_{g,i}$ which is released by the batch-CDS since $g(x_1, \ldots, x_{k-1}) = f(x_1, \ldots, x_k) = 1$; (2) Collects the values $s_f + r_{g,i}$ sent during the second step, and recovers the value of s_f.

It is not hard to verify that perfect privacy holds. Indeed, suppose that $(x_1, \ldots, x_k)$ does not satisfy f. Then, the only s_f-dependent value which is released is $s_f + r_{g,i}$ where g is the restriction of f to x_k. However, since $(x_1, \ldots, x_k)$ fails to satisfy f, its prefix does not satisfy g and therefore $r_{g,i}$ remains hidden from the receiver.

We complete the proof by analyzing the communication complexity. The last party sends exactly $|\mathcal{F}_k|$ field elements. By the induction hypothesis, each of the other parties sends at most $R_{k-1} \cdot |\mathcal{F}_{k-1}| = |\mathcal{F}_k|$ field elements, and the first party sends $R_{k-1} \cdot |\mathcal{F}_{k-1}|/2 = |\mathcal{F}_k|/2$ field elements, as required. □

3.2 Amortization for CDS

We use the above lemma to amortize the complexity of CDS over long secrets.

Theorem 3.7 (Theorem 3.7 restated)**.** *Let $\mathcal{F} = \{\mathcal{F}_i\}_{i=1}^k$ be a regular family of functions, and let $f \in \mathcal{F}_k$. Then for $m = |\mathcal{F}_k|/2$ there exists a multilinear (k-party) CDS which supports m field element secrets with information ratio of 4. Moreover, one of the parties has information ratio of 2.*

Proof. Given a secret vector $s \in \mathbb{F}^m$, we duplicate each secret twice and index the secrets by predicates $p \in \mathcal{F}_m$ such that $s_p = s_{\bar{p}}$ (i.e., a predicate and its complement index the same secret). Note that properties (1) and (2) guarantee that $\mathcal{F}_k$ is closed under complement. On inputs $x_1, \ldots, x_k$, the parties make two calls to $\mathcal{F}_k$-batch CDS. In the first call the secret associated with a predicate $p \in \mathcal{F}_k$ is a random value $r_p \in \mathbb{F}$. In the second call, for every predicate $f + p + 1 \in \mathcal{F}_k$, we release $s_p + r_p$. Since the mapping $p \mapsto p + f + 1$ is a bijection, the second call associates exactly one secret to each function.

Correctness. Suppose that $f(x_1, \ldots, x_k) = 1$. Recall that each of the original secrets s_i appears in two copies $(s_p, s_{\bar{p}})$ for some predicate p. Since one of these copies is satisfied by $x = (x_1, \ldots, x_k)$, it suffices to show that, whenever $p(x) = 1$, the secret s_p can be recovered. Indeed, for such a predicate p, the value r_p is released by the first batch-CDS, and the value $s_p + r_p$ is released by the second batch-CDS. The latter follows by noting that x satisfies the predicate $p + f + 1$ (since it satisfies both f and p). It follows that s_p can be recovered for every p which is satisfied by x, as required.

Privacy. Suppose that $f(x) = 0$. We show that all the "virtual secrets" s_p remain perfectly hidden in this case. Indeed, for every $p \in \mathcal{F}_k$, it holds that whenever $f(x) = 0$, either $(f+p+1)(x) = 0$ or $p(x) = 0$, and therefore, for any p, either r_p or $s_p + r_p$ are released, but never both.

Finally, using Lemma 1.5, the total communication complexity of each party is $2|\mathcal{F}_k| = 4m$ and the first party has communication complexity of $2|\mathcal{F}_k|/2 = 2m$, as claimed. Also note that our protocol is multilinear. Indeed, our construction uses batch-CDS on "virtual" secrets that are

linear in the original secrets and the randomness. In addition, batch-CDS itself is multilinear in the sense that the output of every player is a vector with coordinates of the form $s + r$ or r for some secret s and random element r. □

Plugging in the regular family of all functions, we get the following corollary.

Corollary 3.8 (Theorem 1.2 restated)**.** *Every function $f : [N]^k \to \{0,1\}$ has a multilinear k-party* CDS *protocol that supports secrets of length 2^{N^k-1} with information ratio of 4. Moreover, for secrets of length $k2^{N^k-1}$, one can get an information ratio of $4 - \frac{2}{k}$ (i.e., 3 for the case of $k = 2$).*

Proof. The first part follows directly from Theorem 3.7. To prove the "Moreover" part, we exploit the fact that in Theorem 3.7 one of the parties (say the first) has information ratio of 2. In particular, partition the $k2^{N^k-1}$-long secret to k blocks of length $B = 2^{N^k-1}$ and run the protocol k times (one for each block) where in each invocation a different party plays the role of the first party. This way each party communicates $4(k-1)B + 2B$ elements for a secret of length kB, and the information ratio is $4 - \frac{2}{k}$. □

Applying Theorem 3.7 to the class of all degree-D multivariate polynomials (which was shown to be regular in Proposition 3.5), we conclude:

Corollary 3.9. *Every multivariate polynomial $p : \{0,1\}^{\ell_1} \times \cdots \times \{0,1\}^{\ell_k} \to \{0,1\}$ over $\ell = \sum_i \ell_i$ variables with total degree of at most D admits a k-party* CDS *protocol with information ratio of 4 for secrets of length $P(\ell, D)/2$ where $P(\ell, D)$ denotes the number of multivariate polynomials with ℓ variables and total degree of at most D over the binary field.*

Note that $P(\ell, D) \leq 2^{D \cdot \ell^D}$ which, for constant D, is quasipolynomial in the size of the total domain $L = 2^\ell$ (as opposed to exponential in the size of the domain as in Corollary 3.8). Overall, in order to construct an amortized CDS for a target function f, it is beneficial to employ Theorem 3.7 with the smallest regular family of functions that constrains f. Smaller families can significantly improve the amortization starting point.

4 From multiparty CDS to d-uniform secret-sharing

As shown by [BIKK14] CDS is closely related to secret-sharing. We further extend this relation by using our multiparty CDS to construct efficient secret-sharing for d-uniform access structures (here, efficiency is measured by the information ratio of the scheme).

Hypergraph representation. Every access structure $\mathcal{A}$ can be represented as a hypergraph $\mathcal{H} = (V, E)$ whose vertices correspond to parties of $\mathcal{A}$ and hyperedges correspond to **minimal** authorized sets of $\mathcal{A}$ (a minimal authorized set is a set for which no subset is authorized). In the case of d-uniform access structure $\mathcal{A}$, it is convenient to restrict the attention to minimal authorized sets of size exactly d while keeping in mind that all larger sets are always authorized. Under this convention, we represent d-uniform access structures by d-uniform hypergraphs.

Hypergraph decomposition. A sub-hypergraph $\mathcal{G} = (V', E')$ of a hypergraph $\mathcal{H} = (V, E)$ is a hypergraph such that $V' \subset V$ and $E' \subset E$. Decomposing a "complicated" hypergraph into a set of "simple" sub-hypergraphs is a common way to achieve secret-sharing schemes for the former. For that matter, Stinson's theorem [Sti94] is commonly used. In this paper, a "complicated" hypergraph is a d-uniform hypergraph, and a "simple" hypergraph is a *d-partite hypergraph* - a hypergraph whose vertices can be partitioned into d parts $V_1, \ldots, V_d$ such that every hyperedge is an element of $V_1 \times \ldots \times V_d$. The following fact follows from Stinson's theorem. (We sketch a proof for completeness)

Fact 4.1. *Let $\mathcal{H}$ be a hypergraph, and let $\mathcal{H}_1, \ldots, \mathcal{H}_t$ be sub-hypergraphs of $\mathcal{H}$ such that for some $0 < c \leq 1$ every edge $e \in E$ appears in at least $c \cdot t$ different sub-hypergraphs. Assume in addition that every sub-hypergraph $\mathcal{H}_i$ has a secret-sharing scheme with information ratio of at most r for secrets whose domain $\mathcal{S}$ is of size at least t.[3] Then $\mathcal{H}$ has secret-sharing scheme with information ratio at most $\frac{r}{c}$ for secrets taken from $\mathcal{S}^{ct}$. In addition, if the schemes for $\mathcal{H}_i$ are multilinear, the new scheme is multilinear as well.*

Proof. We use a maximum distance separable (MDS) linear error correcting code $L : \mathcal{S}^{ct} \to \mathcal{S}^t$ that handles $1 - c$ fraction of erasures with rate c. (Since $|\mathcal{S}|$ is of size at least t such as code exists, e.g., Reed-Solomon code). Encode a vector s of ct secrets using the code to get a codeword of length t, and distribute the i-th coordinate of the codeword using a secret sharing scheme for $\mathcal{H}_i$. Since the original secret vector can be reconstructed by observing a c fraction of the entries of the codeword, every hyperedge of $\mathcal{H}$ (authorized set) can recover the vector s. On the other hand, an unauthorized set e does not appear as an hyperedge in any of the hypergraphs and so such a coalition does not learn even a single coordinate of $L(s)$. □

4.1 Secret-Sharing for d-partite hypergraphs

For a d-partite hypergraph $\mathcal{H} = (V = (V_1, \ldots, V_d), E)$ we define $f_{\mathcal{H}} : V_1 \times \ldots \times V_d \to \{0, 1\}$ to be the function that outputs 1 on an input $e = (v_1, \ldots, v_d)$ if and only if $e \in E$.

Lemma 4.2. *Suppose that $f_{\mathcal{H}}$ has a d-party* CDS *scheme $(F_1, \ldots, F_d)$ with information ratio w for secrets whose domain $\mathcal{S}$ is of size at least n where n is the number of nodes in $\mathcal{H}$. Then, there is a secret sharing scheme for $\mathcal{H}$ with information ratio $w + 2$ for secrets in $\mathcal{S}$. Moreover, if the* CDS *scheme is linear (resp., multilinear) then the secret sharing scheme is also linear (resp., multilinear).*

Proof. Let $\mathcal{S}$ be the secret domain of the CDS for $f_{\mathcal{H}}$ and let $|V| = n$. Given a secret $s \in \mathcal{S}$ we share it as follows. First, we use $(d+1)$-out-of-$(d+1)$ secret sharing to share s into $(s_0, \ldots, s_d)$. Next, we sample randomness r for the CDS and distribute the secret s_0; That is, for each vertex $v \in V_i$, we generate the share $a_v = F_i(v, s_0, r)$. Finally, we use $(d+1)$-out-of-n Shamir's secret sharing to share the secret s into n shares $(b_v)_{v \in V}$. (For this we view $\mathcal{S}$ as a field and use the fact that $|\mathcal{S}| \geq n$.) Overall, the share of the vertex $v \in V_i$ is the triplet (s_i, a_v, b_v). Observe that the information ratio is $w + 2$ (since threshold access structures can be realized with information ratio of 1).

Correctness: Consider an authorized coalition parties $e \subset V$. If e contains more than d parties then the secret can be recovered based on the b parts. Otherwise, $e \in E$. In this case, the CDS

[3] This condition can be completely waived at the expense of losing a constant factor in the final rate.

allows the coalition to recover s_0. Moreover, since e must contain exactly one vertex from each part V_i of the graph the parties also have the shares $s_1, \ldots, s_d$ and they can recover s.

Privacy: Consider an unauthorized coalition of parties $e \subset V$. In any case e is smaller than $d+1$ and so the b parts reveal no information. If the size of e is smaller than d then e does not contain a vertex from V_i for some $i \in [d]$, and so s_i remains hidden and no information is revealed about s. If e is of size d then $e \notin E$ and so the CDS keeps s_0 hidden, and no information is revealed about s. □

Corollary 4.3. *Every d-partite hypergraph has a d-uniform, multilinear secret-sharing scheme with information ratio of 6 for secrets of domain size 2^{n^d-1}, where n is the number of nodes in $\mathcal{H}$.*

Proof. Let $\mathcal{H}$ be a d-partite hypergraph with n vertices $V = (V_1, \ldots, V_d)$. Since each V_i contains at most n vertices, the function $f_{\mathcal{H}}$ can be viewed as a binary function over $[n]^d$. We construct a d-party CDS for $f_{\mathcal{H}}$ using Corollary 3.8, and then use Lemma 4.2 to get the required secret-sharing scheme. □

4.2 Secret-Sharing for d-uniform hypergraphs

Recall that Fact 4.1 shows that the case of general d-uniform hypergraphs reduces to the case of d-partite hypergraphs provided that we have a "good" covering of hypergraphs by d-partite hypergraphs. The following lemma uses a probabilistic argument to establish the existence of such a good covering.

Lemma 4.4. *Let $\mathcal{H} = (V, E)$ be a d-uniform hypergraph with n vertices. Let $t = 3\frac{d^d(d^d+1)^2}{d!} \cdot \ln(n^d)$. There exists a set of sub-hypergraphs of $\mathcal{H}$ denoted by $\{\mathcal{H}_1, \ldots, \mathcal{H}_t\}$ such that every $\mathcal{H}_i$ is d-partite and every edge of $\mathcal{H}$ appears in at least $\frac{d!}{d^d+1} \cdot t$ sub-hypergraphs.*

The constant $\frac{d!}{d^d+1}$ can be replaced with any constant strictly smaller than $\frac{d!}{d^d}$.

Proof. Every mapping $\varphi : V \to [d]$ naturally induces a partition of V to d subsets $V_1, \ldots, V_d$ where V_j contains all nodes v for which $\varphi(v) = j$. Let us denote by $\mathcal{H}(\varphi)$ the d-partite graph obtained by keeping all the hyperedges of $\mathcal{H}$ that contain a single vertex from every subset V_i. We will use the probabilistic method to show that there exists a t-tuple of mappings $\varphi = (\varphi_1, \ldots, \varphi_t)$ that induce t hypergraphs $(\mathcal{H}_1, \ldots, \mathcal{H}_t)$, $\mathcal{H}_i = \mathcal{H}(\varphi_i)$ that satisfy the lemma.

For every $i \in [t]$ choose the mapping $\varphi_i : V \to \{1, \ldots, d\}$ uniformly at random. We show that, with non-zero probability, for any hyperedge $e = (v_1, \ldots, v_d)$ of H there are at least $\frac{d!}{d^d+1} \cdot t$ indices $i_1, \ldots, i_\ell$ for which e appears as an hyperedge in $\mathcal{H}_i$. The claim will then follow by applying a union bound over all $|E| < n^d$ hyperedges.

Fix $e = (v_1, \ldots, v_d)$, and call an index i good if the set $\{\varphi_i(v_1), \ldots, \varphi_i(v_d)\}$ covers $[d]$. (This guarantees that e appears as an hyperedge in H_i). Let w_i be a random variable that indicates whether i is good, and let $w = \sum_{i=1}^{t} w_i$. We get that $\mathbb{E}[w_i] = \Pr[i \text{ is good}] = \frac{d!}{d^d}$ and conclude, by the linearity of expectation, that $\mathbb{E}[w] = t\frac{d!}{d^d}$. We use the following form of Chernoff bound:

$$\Pr[X \leq (1-\delta)\mu] \leq e^{-\frac{\delta^2\mu}{2}},$$

where X is a sum of independent indicator random variables, $\mu = \mathbb{E}[X]$ and $0 \leq \delta \leq 1$. Since the w_i's are statistically independent (due to the independent choice of φ_i), we can apply Chernoff

bound with $\delta = 1 - \frac{d^d}{d^d+1}$ and get:

$$\Pr\left[w \leq t\frac{d!}{d^d+1}\right] \leq e^{-1.5\cdot\ln(n^d)} = n^{-1.5\cdot d}.$$

By applying the union-bound over all (at most n^d) hyperedges, we conclude that all hyperedges e have at least $t\frac{d!}{d^d+1}$ good indices except with probability

$$n^d \cdot \Pr\left[w \leq t\frac{d!}{d^d+1}\right] \leq n^{-\frac{d}{2}} < 1.$$

Therefore there exists a t-tuple of mappings φ with the required property. □

We can now prove Theorem 1.1 (restated here for convenience).

Theorem 4.5. *Every d-uniform hypergraph $\mathcal{H}$ has a multilinear d-uniform secret-sharing scheme with information ratio $6 \cdot \frac{d^d+1}{d!}$ for secrets of length $\exp(O(n^d \cdot \log n \cdot d^{2d+1}))$ where n is the number of nodes in $\mathcal{H}$.*

Proof. First, we use Lemma 4.4 to decompose $\mathcal{H}$ into $t = 3\frac{d^d(d^d+1)^2}{d!}\cdot\ln(n^d)$ sub-hypergraphs that are d-partite, such that every edge of $\mathcal{H}$ appears in at least $c\cdot t$ different sub-hypergraphs where $c = \frac{d!}{d^d+1}$. Following Corollary 4.3, every sub-hypergraph in the decomposition has a multilinear d-uniform secret-sharing scheme with information ratio of 6 for secrets of domain size 2^{n^d-1}. Finally, we use Fact 4.1 to establish a multilinear d-uniform secret-sharing scheme for $\mathcal{H}$ with information ratio $\frac{6}{c} = 6 \cdot \frac{d^d+1}{d!}$ for secrets domain of size $\left(2^{n^d-1}\right)^{ct} = 2^{(n^d-1)3d^d(d^d+1)\ln(n^d)} = \exp(O(n^d \cdot \log n \cdot d^{2d+1}))$. □

For the special case of $d = 2$ (i.e., forbidden graph access structure) we get the following corollary.

Corollary 4.6. *Every forbidden graph access structure has a multilinear secret-sharing scheme with information ratio of* 12.5*.*

Proof. As explained in Corollary 3.8 there exists a multilinear 2-party CDS with information ratio of 3. □

Remark 4.7. *There are some tweaks that can be applied to our secret-sharing construction to get (minor) improvements in the information ratio. Since these modifications complicate the statements and their proofs, we briefly describe them here instead:*

1. *In our construction of secret-sharing for d-partite hypergraphs, as described in Lemma 4.2, each party is given a $(d+1)$-out-of-n share of Shamir's secret sharing. This is done to promise that any $d+1$ parties can reconstruct the secret. As we use the construction from Lemma 4.2 multiple times in our final construction for d-uniform hypergraphs, this creates a redundancy. Instead, we can drop this step at Lemma 4.2, apply Lemma 4.4, and add a Shamir secret sharing for $d+1$ sets at the end. This gives us an overall information ratio of $5 \cdot \frac{d^d+1}{d!} + 1$.*
2. *In Lemma 4.4 we used Chernoff bound to show the existence of our desired decomposition. We chose a value for δ that is $1 - \frac{d^d}{d^d+1}$. In general, every value of δ smaller than 1 would suffice. Hence, the information ratio can be arbitrarily close to $5 \cdot \frac{d^d}{d!} + 1$. (Naturally, when the information ratio gets closer to $5 \cdot \frac{d^d}{d!} + 1$, longer secrets are required in order to achieve amortization).*

3. *An additional improvement can be obtained by plugging-in the optimized* $4 - \frac{2}{k}$ *bound on the information ratio of* k*-party* CDS *(Corollary 3.8). This yields a secret-sharing scheme for* d*-uniform hypergraphs with an information ratio* $(5 - \frac{2}{d}) \cdot \frac{d^d}{d!} + 1 + \epsilon$ *for every* $\epsilon > 0$.

5 Lower bounds for d-uniform secret sharing

In this section we discuss the possibility of proving lower-bounds against d-uniform secret sharing.

5.1 Lower bound for the share size of d-uniform linear SS

We start by showing a lower bound on the *share size* (in bits) of linear d-uniform secret sharing. This immediately implies a similar lower-bound on the share size of multilinear schemes. (Since one can turn a multilinear scheme into a linear scheme by fixing all but a single secret). The following definitions are needed:

Definition 5.1. *Let* $\mathcal{A}$ *be an access structure and* q *be a prime power. Define* $\rho_q(\mathcal{A})$ *to be the minimal information ratio of all* ***linear*** *secret sharing schemes realizing* $\mathcal{A}$ *over the field* $\mathbb{F}_q$ *(the finite field over* q *elements) .*

Definition 5.2. *For an access structure* $\mathcal{A}$*, we say that* $\mathcal{A}$ *has* rank r*, if every minimal authorized set of* $\mathcal{A}$ *is of size at most* r*.*

The following theorem is proved in [BFM16]:

Theorem 5.3. *Let* q *be a prime power, and* s, r, n *be integers such that* $s > \log(n)$*. Denote by* $T(q, s, r, n)$ *the number of access structures with* n *parties, rank* r *and* $\rho_q(\mathcal{A}) \leq s$*. Then* $T(q, s, r, n) \leq 2^{2rns^2 \log(q)}$

From this theorem, it is easy to get a lower bound for the maximum share size of linear d-uniform secret sharing schemes. The following corollary is presented by [BFM16] for the case of forbidden graphs. We generalize this result to d-uniform access structures:

Corollary 5.4. *For every* n*, there exists a* d*-uniform access structure* $\mathcal{A}$ *such that the maximal share size of every linear secret sharing scheme realizing it (and therefore of every multilinear scheme as well), is at least* $(\sqrt{\frac{n^{d-1}}{2d^d(d+1)}})$

Proof. As we are interested in the share size, as opposed to the information ratio, we denote $z = s\log(q)$. Every d-uniform access structure, is a rank $d+1$ access structure. Therefore we get that on one hand the number of d uniform access structures such that $\rho_q(\mathcal{A}) < s$ is at most $T(q, s, d+1, n) \leq 2^{2(d+1)nz^2}$. On the other hand, the number of d-uniform access structures is $2^{\binom{n}{d}}$. Therefore, $2^{2(d+1)nz^2} \geq 2^{\binom{n}{d}}$ which in turn means that $z \geq (\sqrt{\frac{n^{d-1}}{2d^d(d+1)}})$. □

For a constant d, we conclude that the share size of d-uniform linear (or multilinear) SS must be at least $\Omega_d(n^{\frac{d-1}{2}})$. We conclude that multilinear SS (like the one from Theorem 1.1) cannot achieve constant information rate for secrets shorter than $\Omega_d(n^{\frac{d-1}{2}})$. Note that in our scheme amortization begins only for exponentially long secrets. Narrowing this gap, even for multilinear schemes, remains an interesting open problem.

5.2 Limitations of Shannon's Inequalities based Lower-bounds

A commonly used technique for proving secret sharing lower bounds is by analyzing the entropy of the shares (induced by a uniform choice of the secret). In particular, one typically relies on the following claim. (Below H denotes Shannon's entropy).

Claim 5.5. *Let $\mathcal{A}$ be an access structure and let Σ be a (perfect) secret sharing scheme for $\mathcal{A}$ with secret domain of $\mathcal{S}$. For a set of parties A, denote by S_A the joint distribution of the shares of parties in A induced by a uniformly chosen secret $S \xleftarrow{R} \mathcal{S}$, and by the internal randomness of Σ. Define $f(A) = \frac{H(S_A)}{H(S)}$. Then the following holds:*

1. *Monotonicity. If $A \subset B$, then $f(B) \geq f(A) \geq f(\emptyset) = 0$.*
2. *Submodularity. $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$.*
3. *Strong Monotonicity. If $A \notin \mathcal{A}, B \in \mathcal{A}$, and $A \subset B$, then $f(B) \geq f(A) + 1$.*
4. *Strong Submodularity. If $A, B \in \mathcal{A}$ and $A \cap B \notin \mathcal{A}$, then $f(A)+f(B) \geq f(A \cup B)+f(A \cap B)+1$.*

These inequalities are called Shannon inequalities, and a proof of the claim is given by Csirmaz [Csi97]. The claim is typically used to lower-bound, for some party a, the value of $f(a)$ and conclude a lower-bound on the (normalized) entropy value of a's share which implies a lower-bound on the share size. Indeed, this technique was used by Csirmaz to prove the best known lower-bound $(\frac{n}{\log n})$ on the information ratio of some n-party access structure. Csirmaz also showed that this method cannot prove superlinear lower-bounds since there is a "semi-entropy" function g that satisfies the conditions of Claim 5.5 but assign to each singleton a value of $O(n)$. We use the same idea to show a barrier of d for the case of d-uniform access structures.

Theorem 5.6. *Let $d \geq 2$. Then Shannon inequalities cannot give a better lower bound than d for the information ratio of d-uniform secret sharing.*

Proof. Let $\mathcal{A}$ be a d-uniform access structure, and let A be a non-empty set of parties. For $t = min\{|A|, d+1\}$ we define

$$g(A) = \left(\sum_{i=0}^{t-1} (d + 1 - i) \right) - 1$$

For the empty set, we define $g(\emptyset) = 0$. Note that $g(\{p\}) = d$ for every party p. Thus, showing that g satisfies the Shannon inequalities will prove the theorem. Clearly g is monotone and non-negative, so (1) is satisfied. For (3), we assume $A \notin \mathcal{A}, B \in \mathcal{A}$, and $A \subset B$. The set A contains at most d parties (since it is unauthorized), and the set B contains more parties than A, therefore (3) follows.

For (2) and (4), we first ignore the -1 at the definition of g and consider the following cases:

1. $|A| \geq d+1$. In this case, $g(A) = g(A \cup B)$ and we reduce (2) and (4) to (1) and (3) respectively. The case where $|B| \geq d + 1$ is symmetric.
2. $A \subset B$. In this case $A = A \cap B$ and $B = A \cup B$. (2) follows. In addition, if $A \in \mathcal{A}$ then $A \cap B \in \mathcal{A}$ and so (4) vacuously follows. The case where $B \subset A$ is symmetric.

3. Assume $|A|, |B| \leq d+1$ and that $A \cup B \neq A, B$. We show that $g(A) - g(A \cap B) \geq g(A \cup B) - g(B) + 1$, thus showing both (4) and (2). We denote $C = A - (A \cap B)$ and $D = (A \cup B) - B$. Note that $C = D$ and let $\ell := |C| = |D|$. This implies that $g(A) - g(A \cap B)$ is the sum of the last ℓ consecutive integers of $g(A)$, denote this sum by $x_1 + \cdots + x_\ell$. Also, $g(A \cup B) - g(B)$ is the sum of the last ℓ consecutive integers of $g(A \cup B)$, denote this sum by $y_1 +, \ldots, +y_\ell$. Since A is a strict subset of $A \cup B$, it holds that for every i, $x_i > y_i$, and so (2) and (4) follow.

Returning to the original definition of g (with the -1), we note that this substraction matters only if one of the sets is empty. The cases where $A = \emptyset$ or $B = \emptyset$ are easily validated. In case $A \cap B = \emptyset$ we argue that

$$g(A) + g(B) \geq g(A \cup B) + 1.$$

Denote $a = \min\{|A|, d+1\}$, $b = \min\{|B|, d+1\}$ and $c = \min\{a+b, d+1\}$. On the LHS we have $(\sum_{i=0}^{a-1}(d+1-i) + \sum_{i=0}^{b-1}(d+1-i)) - 2$, and on the RHS we have $(\sum_{i=0}^{c-1}(d+1-i)) - 1$. One can easily verify that the LHS is indeed at least as big as the RHS, with equality in case $a = b = 1, c = 2$. □

6 Reducing partial-PSM to CDS

In this section we show how to reduce partial-PSM to CDS with better overhead than the one achieved in [AARV17]. Let $f : (\mathcal{X} \times \mathcal{W}) \times (\mathcal{Y} \times \mathcal{Z}) \to \{0,1\}$ be the target function where $\mathcal{X}$ and $\mathcal{Y}$ are the private domains and $\mathcal{W}$ and $\mathcal{Z}$ are the public domains. We associate with f the function family

$$\mathcal{F} = \{f(\cdot, w, \cdot, z) : w \in \mathcal{W}, z \in \mathcal{Z}\} \tag{1}$$

that consists of all two-party functions that can be derived from f after fixing some values for the public domains. For the sake of simplicity, we assume the private input domains $\mathcal{X}$ and $\mathcal{Y}$ are both $\{0,1\}^t$, and the public domains $\mathcal{W}$ and $\mathcal{Z}$ are both $\{0,1\}^{\ell-t}$. That is, Alice and Bob each hold ℓ bits, out of which t bits are considered private. By abuse of notation, we sometimes view the domain of f as $\{0,1\}^\ell \times \{0,1\}^\ell$. We will use the following notations:

- We denote by $\mathsf{CDS}(f, b)$ the minimal total communication complexity of a perfect CDS for f supporting b-bit secrets.
- We denote by $\mathsf{CDS}(\ell, b)$ the maximal value of $\mathsf{CDS}(f, b)$ over all functions $f : \{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}$.

Overview. The general idea behind the reductions is as follows: Let $(x, w_0), (y, z_0)$ be the input for Alice and Bob respectively. Let f_{w_0,z_0} be the function f restricted to $w = w_0, z = z_0$. The function f_{w_0,z_0} is known to Carol, but not to Alice and Bob. Suppose that we have a family of PSM protocols $\{F_{(w,z)} = (F_{(w,z),1}, F_{(w,z),2})\}_{w,z}$ for all possible functions $f_{w,z}$. The idea is to release only the transcript of $F_{(w_0,z_0)}(x, y, r)$ via the aid of CDS. Naively, this can be done by letting Alice generate, for every (w, z), the PSM messages $F_{(w,z),1}$ and use the result as a secret for a CDS over the 2-party predicate "Is (w_0, z_0) equal to (w, z)?", and do the same with Bob's messages. Clearly, the overhead in this case is huge (exponential in the length of the public input (w, z)). To see how this overhead can be reduced, imagine that the underlying PSM has the property that Alice's (resp., Bob's) computation can be decomposed to blocks where in the i-th block we compute one

of L functions $g_1(x,r),\ldots,g_L(x;r)$ depending on the value of (w,z). Then, we can release each block of $F_{(w,z),1}$ by making only L calls to a CDS. We start with a formalization of this idea with the notion of PSM *compilers*, and then give concrete examples of this approach.

6.1 PSM Compilers

Definition 6.1 (PSM Compiler)**.** *Let $\mathcal{F}$ be a function family. We say that C is a* PSM *compiler for $\mathcal{F}$, if C maps every function $f \in \mathcal{F}$ to a (fully secure)* PSM *$F = (F_1, F_2)$. As usual, let x and y be Alice's and Bob's inputs respectively, and let r be the randomness of the* PSM*. We say that C is (c, v, b, L)-uniform if there exist v families of functions $\mathcal{G}_1, ..., \mathcal{G}_v$ and a pair of functions h_A, h_B with the following properties:*

1. *Every* PSM *$F = (F_1, F_2)$ in the image of C can be written as a concatenation of functions $(h_A, h_B, g_1, ..., g_v)$, where $g_i \in \mathcal{G}_i$ is chosen based on f (and h_A and h_B are identical for all $f \in \mathcal{F}$). Every function $g_i \in \mathcal{G}_i$ depends either on (x,r) or on (y,r), and the functions h_A and h_B depend on (x,r) and (y,r) respectively.*
2. *Every function family $\mathcal{G}_i$ contains at most L functions.*
3. *The output length of every function $g \in \cup\mathcal{G}_i$ is at most b bits, and the total output length of h_A and h_B is at most c bits.*

Lemma 6.2. *Let f be a two-party predicate whose private and public domains are $\{0,1\}^t$ and $\{0,1\}^{\ell-t}$, for each party. Let $\mathcal{F}$ be the function family associated with f as in Eq. (1). Then, a (c, v, b, L)-uniform* PSM *compiler for $\mathcal{F}$ implies a partial-*PSM *for f with communication complexity $O(c + L \cdot v \cdot \mathsf{CDS}(\ell - t, b))$.*

Proof. Let x and y be the private inputs of Alice and Bob, and let w and z denote their public inputs. Let $(h_A, h_B, g_1, ..., g_v)$ be the compiled representation of the PSM for $f_{w,z} = f(\cdot, w, \cdot, z)$ and let r be the randomness used by that PSM. Recall that for every i, g_i is chosen from $\mathcal{G}_i$ according to the public inputs w, z. Hence, for every g, i, we can define a predicate $P_{g,i}$ that given w, z as an input outputs 1 if $g_i = g$. To execute a partial PSM, Alice and Bob sample joint randomness r and send the following messages:

- Alice sends $h_A(x,r)$ and Bob sends $h_B(y,r)$.
- For every $i \in [v]$ and $g \in \mathcal{G}_i$ the parties invoke a CDS (with fresh randomness) on the public inputs w and z, predicate $P_{g,i}$ (i.e., "Is g equal to g_i?"), and secret $g(x,r)$ (if g depends on Alice's input) or $g(y,r)$ (if g depends on Bob's input).

Note that the secret is known either to Alice or Bob, but not to both. Hence we should use a proper CDS which operates even if the secret is known only to one of the parties. Recall that this feature can be obtained from any (standard) CDS at the expense of increasing the total communication by $|s|$, the length of the secret (see Remark 2.3). It follows that the overall communication complexity is at most $c + L \cdot v \cdot (\mathsf{CDS}(\ell - t, b) + b) \leq c + 2L \cdot v \cdot \mathsf{CDS}(\ell - t, b)$, as required. (The inequality follows by noting that $\mathsf{CDS}(\ell - t, b) \geq b$.)

The correctness of CDS guarantees that Carol, who knows w and z, can recover the value

$$\hat{f}_{w,z}(x, y; r) = (h_A(x,r), h_B(y,r), g_1(x,y,r), ..., g_v(x,y,r)),$$

which, by the correctness of the PSM for $f_{w,z}$, can decoded to $f(x, w, y, z)$.

On the other hand, we can perfectly simulate the view of Carol based on w, z and $f(x, w, y, z)$ as follows. First sample $\hat{f}_{w,z}(x, y; r)$ using the PSM simulator; Then, use the corresponding values to perfectly sample the transcript of the CDS calls in which the predicate was satisfied. Finally, use the CDS simulator to sample the transcripts for the CDS calls that did not satisfy the predicate. The lemma follows. □

6.2 Partial-PSM for General Functions

Our first reduction employs a simple PSM compiler which reduces the evaluation of an arbitrary function to the case of inner product. (This can be viewed as a special case of the multilinear PSM from [BIKK14].)

Theorem 6.3. *Every two-party functionality $f : \{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}$ with private domain of $\{0,1\}^t$ admits a prefect partial-PSM with communication complexity $O(2^t + 2^{2t} \cdot \mathsf{CDS}(\ell - t, 1))$.*

Proof. By Lemma 6.2 it suffices to show that the family $\mathcal{F}_t$ of all all two-party functionality over $\{0,1\}^t \times \{0,1\}^t$ admit a (c, v, b, L)-uniform PSM compiler PSM with $c = O(2^t), v = O(2^{2t})$ and $b = L = O(1)$.

We describe the compiler in two steps beginning with following PSM compiler (which does not achieve the required efficiency properties).

- **Public input:** A function $f : \{0,1\}^t \times \{0,1\}^t \to \{0,1\}$, represented as its truth table $P \in \{0,1\}^{2^{2t}}$.
- **Alice's inputs:** $x \in \{0,1\}^t$ represented as the indicator vector $e_x \in \{0,1\}^{2^t}$.
- **Bob's inputs:** $y \in \{0,1\}^t$ represented as the indicator vector $e_y \in \{0,1\}^{2^t}$.
- **Carol's output:** $f(x, y)$ represented by the inner product $\langle P, e_x \otimes e_y \rangle$, where $\otimes$ denotes tensor product.
- **Shared randomness:** random bit r and random strings $a', b' \in \{0,1\}^{2^t}$.

The Protocol:

- Alice and Bob send to Carol

$$\alpha = e_x + a' \qquad \text{and} \qquad \beta = e_y + b', \tag{2}$$

  respectively. In addition, Alice sends

$$\gamma = -\big\langle P, (e_x + a') \otimes b' \big\rangle + r, \tag{3}$$

  and Bob sends

$$\delta = -\big\langle P, a' \otimes e_y \big\rangle - r. \tag{4}$$

- Carol outputs the value $\alpha\beta + \gamma + \delta$.

Correctness follows directly from the construction, by noting that the product $\alpha\beta$ simplifies to

$$\left\langle P, (e_x + a') \otimes (e_y + b') \right\rangle = \left\langle P, e_x \otimes e_y \right\rangle + \left\langle P, (e_x + a') \otimes b' \right\rangle + \left\langle P, a' \otimes e_y \right\rangle.$$

Privacy is due to the fact that the messages α, β, γ are uniform, and the last message δ is uniquely determined by all other messages and $f(x, y)$. Hence, there exists a simulator S_f that, given $f(x, y)$ perfectly samples the transcript $(\alpha, \beta, \gamma, \delta)$.

The protocol above forms a $(2 \cdot 2^t, 2, 1, 2^{2^{2t}})$-uniform PSM compiler for $\mathcal{F}_t$. Indeed, $h_A = e_x + a'$, $h_B = e_y + b'$ and the function families $\mathcal{G}_1$ and $\mathcal{G}_2$ correspond to computations of $-\left\langle P, (e_x + a') \otimes b' \right\rangle + r$ and $-\left\langle P, e_y \otimes a' \right\rangle - r$ respectively, with all possible values for P. To avoid this double-exponential blow-up, we replace the inner-product computations in (3) and (4) by their randomized encoding. Concretely, letting $u = (e_x + a') \otimes b'$ we replace (3) by

$$\Big(P_i \cdot u_i + s_i \Big)_{i=1}^{2^{2t}}, \tag{5}$$

where $s = (s_1, \ldots, s_{2^{2t}-1})$ is a string of random bits (added to the shared randomness) and $s_{2^{2t}} = r - \sum_{i=1}^{2^{2t}-1} s_i$. Similarly, letting $u' = a' \otimes e_y$ we replace (4) by

$$\Big(- P_i \cdot u'_i + s'_i \Big)_{i=1}^{2^{2t}-1}, \tag{6}$$

where $s' \in \{0,1\}^{2^{2t}-1}$ is a string of random bits (added to the shared randomness) and $s'_{2^{2t}} = -r - \sum_{i=1}^{2^{2t}-1} s'_i$.

The resulting PSM protocol is still correct since Carol can recover the original messages of (3) and (4) by summing-up the entries in (5) and (6) sent by Alice and Bob in the modified protocol. To see that privacy is preserved, observe that, given $f(x, y)$, we can first sample a transcript $(\alpha, \beta, \gamma, \delta)$ for the original protocol, and then sample (5) and (6) by sampling 2^{2t} random bits which sum up to γ together with 2^{2t} random bits which sum up to δ. It is not hard to verify that this simulation is perfect. (Indeed, this is just a special case of the general composition property of randomized encoding, cf. [AIK06].)

The modified compiler now uses $2 \cdot 2^{2t}$ function families $\mathcal{G}_i$ where each family consists of exactly 2 functions (selected according to the i-th bit of P) whose output is a single bit. Hence, we get $(2 \cdot 2^t, 2 \cdot 2^{2t}, 1, 2)$-uniform PSM compiler for $\mathcal{F}_m$, as required. □

Plugging in the CDS construction of [LVW17] to theorem 6.3, we derive the following corollary.

Corollary 6.4. *Let f be a two-party predicate with input domains $\mathcal{X} = \mathcal{Y} = \{0,1\}^{2t}$ then there exists a partial-PSM protocol with overall complexity of $(2^{2t})^{1+o(1)}$.*

The resulting partial-PSM is is quasilinear in the alphabet size, $|\mathcal{X} \times \mathcal{Y}|$, of the private inputs. Note that a direct application of the fully secure PSM of [BIKK14] yields a complexity of $O(2^{\ell/2})$, hence our construction becomes useful only when the length of the secret part t is smaller than $\ell/4$.

6.3 Partial-PSM for Formulas

Our second reduction is based on an information theoretic version of Yao's garbled circuit [IK02]. Recall that a *formula* is a Boolean circuit in which every non-input gate has a fan-out of 1. The *size* of a formula is the number of gates, and its *depth* is the length of longest path from a leaf to the root.

Theorem 6.5. *Let f be a two-party predicate whose private and public domains are $\{0,1\}^t$ and $\{0,1\}^{\ell-t}$, for each party. Let $\mathcal{F}$ be the function family associated with f as in Eq. (1), and assume that every function in $\mathcal{F}$ can be computed by a formula of size B and depth D. Then there is a partial-*PSM *for f with communication complexity of $O(B^3 \cdot \mathsf{CDS}(\ell - t, 2^D))$.*

Proof. By Lemma 6.2 the theorem follows from the existence of a $(O(1), B, 2^D, O(B^2))$-uniform PSM compiler for formulas of size B and depth D. Such a compiler follows immidiately from the information-theoretic variant of garbled circuits for formulas. Roughly speaking, in this construction, Alice and Bob assign a function each gate of the circuit in a way that depends only in the "local" neighborhood of the wire. When viewed as a compiler, this means that, for a wire i, the function g_i is chosen from a relatively small family of functions (of size $O(B^2)$).

For the sake of completeness, we outline the PSM protocol for formulas, as described in [IK02].[4]

- **Public knowledge:** Formula with B wires numbered $1, \ldots, B$ where the output wire is numbered B. Without loss of generality we assume that negations are applied directly on input wires.
- **Inputs:** Alice holds $x \in \{0,1\}^t$ and Bob holds $y \in \{0,1\}^t$.
- **Shared randomness:** random bits r_i for $i \in [B]$ and random strings W_i^β for $i \in [B], \beta \in \{0,1\}$ where the length of W_i^β is defined as follows: $|W_B^\beta| = 0$ and $|W_i^\beta| = 2(|W_o^\beta| + 1)$ where o is the output wire of the gate whose i is an input of. For every W_i^β we denote by $W_i^{\beta,0}$ and $W_i^{\beta,1}$ its equal-size halves (say the left part and right part, respectively).
- **Carol's output:** $C(x, y)$

The protocol is defined as follows.

- **Input wire:** For an input wire i associated with either x_j or $\bar{x_j}$ (denote this value as u), Alice sends $W_i^u || (v \oplus r_i)$. Bob does the same with respect to input wires associated with his input.
- **Output wire of gate:** Let h be a gate with input wires i, j and output wire o. Alice (or Bob) sends $Q_o^{c_i,c_j} = W_i^{c_i \oplus r_i, c_j} \oplus W_i^{c_j \oplus r_j, c_i} \oplus (W_o^{h(c_i \oplus r_i, c_j \oplus r_j)} || h(c_i \oplus r_i, c_j \oplus r_j) \oplus r_o)$ for every $c_i, c_j \in \{0,1\}$, where $||$ denotes concatenation.
- **Output wire of circuit:** Alice (or Bob) sends r_B in addition to $Q_B^{c_i,c_j}$ for $c_i, c_j \in \{0,1\}$

It is not hard to verify that Carol can recover, for each wire i, the value $T_i = (W_i^{\beta_i} || \beta_i \oplus r_i)$ where β_j is the value of wire j induced by applying the circuit C on (x, y). (Indeed, these values are sent directly for the input wires, and can be recovered for a non-input wire, o, based on the Q^o's and on the values of T_i and T_j for the wires i, j that enter the gate which computes o.) The value $C(x, y)$ is then revealed to Carol since r_B is sent to him. (A full and detailed proof of correctness and privacy can be found in [IK02].)

This protocol can be described as a PSM compiler in a very natural way. Let m_i, $1 \leq i \leq l$ be a messege of the protocol. We define a function family $\mathcal{G}_i$ in the following way:

[4] In fact, the protocol yields a stronger form of fully-decomposable randomized encoding, or equivalently multi-party PSM in which each party holds a single bit.

1. If i is an input wire, then (apart from the randomness) m_i depends on the relevant input bit and whether the gate uses the negation of the bit. Therefore there are $2 \cdot 2 \cdot t \leq O(B) \leq O(B^2)$ possible functions in $\mathcal{G}_i$.

2. if i is an intermidiate wire or the output wire, then m_i depends on the wiring of the gate whose output is i, and the type of the gate. Again there are $O(B^2)$ possible functions in $\mathcal{G}_i$.

The output length of each of these functions is at most 2^D. Overall we get a $(0, B, 2^D, O(B^2))$-uniform PSM compiler. The theorem follows from Lemma 6.2. □

References

[AARV17] B. Applebaum, B. Arkis, P. Raykov, and P. N. Vasudevan. Conditional disclosure of secrets: Amplification, closure, amortization, lower-bounds, and separations. In Katz and Shacham [KS17], pages 727–757.

[AIK06] B. Applebaum, Y. Ishai, and E. Kushilevitz. Cryptography in NC^0. *SIAM J. Comput.*, 36(4):845–888, 2006.

[AIR01] W. Aiello, Y. Ishai, and O. Reingold. Priced oblivious transfer: How to sell digital goods. In B. 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.

[Bei11] A. Beimel. Secret-sharing schemes: A survey. In Y. M. Chee, Z. Guo, S. Ling, F. Shao, Y. Tang, H. Wang, and C. Xing, editors, *Coding and Cryptology - Third International Workshop, IWCC 2011, Qingdao, China, May 30-June 3, 2011. Proceedings*, volume 6639 of *Lecture Notes in Computer Science*, pages 11–46. Springer, 2011.

[BFM16] A. Beimel, O. Farràs, and Y. Mintz. Secret-sharing schemes for very dense graphs. *J. Cryptology*, 29(2):336–362, 2016.

[BFMP17] A. Beimel, O. Farrs, Y. Mintz, and N. Peter. Linear secret-sharing schemes for forbidden graph access structures. To appear in TCC 2017, 2017.

[BIKK14] A. Beimel, Y. Ishai, R. Kumaresan, and E. Kushilevitz. On the cryptographic complexity of the worst functions. In Y. 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 317–342. Springer, 2014.

[BL88] J. C. Benaloh and J. Leichter. Generalized secret sharing and monotone functions. In S. Goldwasser, editor, *Advances in Cryptology - CRYPTO '88, 8th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1988, Proceedings*, volume 403 of *Lecture Notes in Computer Science*, pages 27–35. Springer, 1988.

[Bla79] G. R. Blakley. Safeguarding cryptographic keys. In *Proc. AFIPS 1979 National Computer Conference*, pages 313–317. AFIPS, 1979.

[CKGS98] B. Chor, E. Kushilevitz, O. Goldreich, and M. Sudan. Private information retrieval. *J. ACM*, 45(6):965–981, 1998.

[CSGV93] R. M. Capocelli, A. D. Santis, L. Gargano, and U. Vaccaro. On the size of shares for secret sharing schemes. *J. Cryptology*, 6(3):157–167, 1993.

[Csi97] L. Csirmaz. The size of a share must be large. *J. Cryptology*, 10(4):223–231, 1997.

[FKN94] U. Feige, J. Kilian, and M. Naor. A minimal model for secure computation (extended abstract). In F. T. Leighton and M. T. Goodrich, editors, *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada*, pages 554–563. ACM, 1994.

[GIKM00] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin. Protecting data privacy in private information retrieval schemes. *J. Comput. Syst. Sci.*, 60(3):592–629, 2000.

[GKW15] R. Gay, I. Kerenidis, and H. Wee. Communication complexity of conditional disclosure of secrets and attribute-based encryption. In R. Gennaro and M. Robshaw, editors, *Advances in Cryptology - CRYPTO 2015 - 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 485–502. Springer, 2015.

[GPSW06] V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In A. Juels, R. N. Wright, and S. D. C. di Vimercati, editors, *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006*, pages 89–98. ACM, 2006.

[IK00] Y. Ishai and E. Kushilevitz. Randomizing polynomials: A new representation with applications to round-efficient secure computation. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*, pages 294–304. IEEE Computer Society, 2000.

[IK02] Y. Ishai and E. Kushilevitz. Perfect constant-round secure computation via perfect randomizing polynomials. In P. Widmayer, F. T. Ruiz, R. M. Bueno, M. Hennessy, S. Eidenbenz, and R. Conejo, editors, *Automata, Languages and Programming, 29th International Colloquium, ICALP 2002, Malaga, Spain, July 8-13, 2002, Proceedings*, volume 2380 of *Lecture Notes in Computer Science*, pages 244–256. Springer, 2002.

[Ish13] Y. Ishai. Randomization techniques for secure computation. In M. Prabhakaran and A. Sahai, editors, *Secure Multi-Party Computation*, volume 10 of *Cryptology and Information Security Series*, pages 222–248. IOS Press, 2013.

[ISN87] M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. In *Proceedings IEEE Globecom '87*, pages 99–102. IEEE, 1987.

[IW14] Y. Ishai and H. Wee. Partial garbling schemes and their applications. In J. Esparza, P. Fraigniaud, T. Husfeldt, and E. Koutsoupias, editors, *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, volume 8572 of *Lecture Notes in Computer Science*, pages 650–662. Springer, 2014.

[KGH83] E. D. Karnin, J. W. Greene, and M. E. Hellman. On secret sharing systems. *IEEE Trans. Information Theory*, 29(1):35–41, 1983.

[KS17] J. Katz and H. Shacham, editors. *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part I*, volume 10401 of *Lecture Notes in Computer Science*. Springer, 2017.

[KW93] M. Karchmer and A. Wigderson. On span programs. In *Proceedings of the Eigth Annual Structure in Complexity Theory Conference, San Diego, CA, USA, May 18-21, 1993*, pages 102–111. IEEE Computer Society, 1993.

[LVW17] T. Liu, V. Vaikuntanathan, and H. Wee. Conditional disclosure of secrets via non-linear reconstruction. In Katz and Shacham [KS17], pages 758–790.

[Min12] Y. Mintz. Information ratios of graph secret-sharing schemes. Master's thesis, Dept. of Computer Science, Ben Gurion University, 2012.

[Sha79] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, 1979.

[SS97] H. Sun and S. Shieh. Secret sharing in graph-based prohibited structures. In *Proceedings IEEE INFOCOM '97, The Conference on Computer Communications, Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Driving the Information Revolution, Kobe, Japan, April 7-12, 1997*, pages 718–724. IEEE, 1997.

[Sti94] D. R. Stinson. Decomposition constructions for secret-sharing schemes. *IEEE Trans. Information Theory*, 40(1):118–125, 1994.

[SW05] A. Sahai and B. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2005.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il