Electronic Colloquium on Computational Complexity, Report No. 3 (2023)

Streaming Lower Bounds and Asymmetric Set-Disjointness

Shachar Lovett [*]
Computer Science Department
University of California San Diego
slovett@ucsd.edu

Jiapeng Zhang [†]
Department of Computer Science
University of Southern California
jiapengz@usc.edu

January 11, 2023

Abstract

Frequency estimation in data streams is one of the classical problems in streaming algorithms. Following much research, there are now almost matching upper and lower bounds for the trade-off needed between the number of samples and the space complexity of the algorithm, when the data streams are adversarial. However, in the case where the data stream is given in a random order, or is stochastic, only weaker lower bounds exist. In this work we close this gap, up to logarithmic factors.

In order to do so we consider the needle problem, which is a natural hard problem for frequency estimation studied in (Andoni et al. 2008, Crouch et al. 2016). Here, the goal is to distinguish between two distributions over data streams with t samples. The first is uniform over a large enough domain. The second is a planted model; a secret ”needle” is uniformly chosen, and then each element in the stream equals the needle with probability p, and otherwise is uniformly chosen from the domain. It is simple to design streaming algorithms that distinguish the distributions using space $s \approx 1/(p^2 t)$. It was unclear if this is tight, as the existing lower bounds are weaker. We close this gap and show that the trade-off is near optimal, up to a logarithmic factor.

Our proof builds and extends classical connections between streaming algorithms and communication complexity, concretely multi-party unique set-disjointness. We introduce two new ingredients that allow us to prove sharp bounds. The first is a lower bound for an asymmetric version of multi-party unique set-disjointness, where players receive input sets of different sizes, and where the communication of each player is normalized relative to their input length. The second is a combinatorial technique that allows to sample needles in the planted model by first sampling intervals, and then sampling a uniform needle in each interval.

[*] Research supported by NSF awards 1953928 and 2006443.
[†] Research supported by NSF CAREER award 2141536.

ISSN 1433-8092

1 Introduction

The *needle problem* is a basic question studied in the context of streaming algorithms for stochastic streams [AMS99, AMOP08, GH09, CMVW16, BVWY18]. The goal is to distinguish, using a space-efficient single-pass streaming algorithm, between streams sampled from two possible underlying distributions.

Setting notations, we let t denote the number of samples, s the space of the streaming algorithm, n the domain size, and $p \in (0, 1)$ the needle probability. The two underlying distributions are:

- **Uniform:** sample t uniform elements from $[n]$.
- **Planted:** Let $x \in [n]$ be uniformly chosen (the "needle"). Sample t elements, where each one independently with probability p equals x, and otherwise is sampled uniformly from $[n]$.

We will assume that $n = \Omega(t^2)$ so that with high probability, all elements in the stream (except for the needle in the planted model) are unique. The question is what space is needed to distinguish between the two models with high probability.

Sample-space tradeoffs for the needle problem. We start with describing some basic streaming algorithms for the needle problem, in order to build intuition. First, note that we need $p = \Omega(1/t)$ as otherwise the two distributions are statistically close, because with high probability the needle never appears in the planted model.

One possible algorithm is to check if there are two adjacent equal elements in the stream. This requires $t = \Theta(1/p^2)$ samples and space $s = \Theta(\log n)$. Another possible algorithm is to store the entire stream in memory, and check for a repeated element. This algorithm requires less samples, $t = \Theta(1/p)$, but more space, $s = t \log n$. Note that in both cases, we get a sample-space tradeoff of $st = \Theta((\log n)/p^2)$. One can interpolate between these two basic algorithms, but the value of the product st remains the same in all of them. This motivated the following conjecture, given explicitly in [CMVW16] and implicitly in [AMOP08].

Conjecture 1.1 (Sample-space tradeoff for the needle problem)**.** *Any single-pass streaming algorithm which can distinguish with high probability between the uniform and planted models, where p is the needle probability, t the number of samples and s the space, satisfies $p^2 st = \Omega(1)$.*

The best result to date towards Conjecture 1.1 is by Andoni et al. [AMOP08] who showed that $p^{2.5} st^{1.5} = \Omega(1)$ (this bound is indeed weaker since $p = \Omega(1/t)$). Guha et al. [GH09] claimed to prove Conjecture 1.1 but later a bug was discovered in the proof, as discussed in [CMVW16]. In this paper we establish Conjecture 1.1 up to logarithmic factors. We can also handle streaming algorithms which pass over the data stream multiple times, scaling linearly in the number of passes.

Theorem 1.2 (Main theorem)**.** *Any ℓ-pass streaming algorithm which can distinguish with high probability between the uniform and planted models, where p is the needle probability, t the number of samples and s the space, satisfies $\ell p^2 st \log(t) = \Omega(1)$.*

1.1 Application: lower bound for frequency estimation in stochastic streams

For many streaming problems, the current state-of-the-art streaming algorithms space requirements are known to be tight (up to poly-logarithmic terms) in the adversarial model, where the streams arrive in an adversarial order. Following a sequence of works on the random-order model [MP80, DLOM02, GM07, CCM08, CJP08, AMOP08, GM09], Crouch et al. [CMVW16] initiated the study of *stochastic streams*, where the streams are sampled from some underlying distribution. The question is if in this model one can attain better streaming algorithms compared to the adversarial model, utilizing the stochastic nature of the streams; or whether the existing lower bounds can be strengthened to this model as well. The needle problem we described is an example of a problem in the stochastic model.

A basic problem in the streaming literature, starting with the pioneering work of [AMS99], is that of estimating the *frequency moments* of a stream. Given a stream $x_1, \ldots, x_t$ of elements from $[n]$, let f_x denote the number of times an element x appears in the stream. The k-th frequency moment of the stream is

$$F_k = \sum_{x \in [n]} f_x^k.$$

In the adversarial model, there are matching upper and lower bounds of $\tilde{\Theta}(n^{1-2/k})$ [1] on the space needed for a streaming algorithm to approximate F_k [CKS03, IW05]. It was conjectured by [CMVW16] that the same lower bound also holds in the stochastic model. They showed that the result of [AMOP08] gives a somewhat weaker lower bound of $\tilde{\Omega}(n^{1-2.5/k})$ space, and that Conjecture 1.1, if true, implies the tight bound of $\tilde{\Omega}(n^{1-2/k})$. Theorem 1.2 thus verifies their conjecture, up to logarithmic terms, which still implies a lower bound of $\tilde{\Omega}(n^{1-2/k})$. We refer to [CMVW16] for further details.

We note another related application, communicated to us by David Woodruff. McGregor et al. [MPTW12] studied streaming algorithms based on sub-sampling a data stream. In particular, one of the problems they studied is that of frequency estimation. They designed space-efficient streaming algorithms based on sub-sampling, and also gave matching lower bounds, based on the results of Guha et al. [GH09]. However, as later a bug was found in this latter work, the journal version of McGregor et al. [MPTW16] removed the lower bounds. Using Theorem 1.2 the claimed lower bounds hold, up to a logarithmic factor.

[1] We use $\tilde{\Theta}, \tilde{\Omega}$ to ignore poly-logarithmic terms.

1.2 Proof approach

We prove Theorem 1.2 by a reduction to the unique set-disjointness problem in communication complexity. This is a common technique used to prove lower bounds for streaming algorithms [CKS03, BYJKS04, AMOP08, GH09, BVWY18, KPW21].

The basic idea is to partition the stream samples into intervals $I_1, \ldots, I_k$ and consider the stream distribution where we place a single needle uniformly in each interval, and sample the other elements in the stream uniformly. It is straightforward to show that any streaming algorithm which can distinguish this distribution from the uniform distribution using space s, can be used to construct a communication protocol that solves the k-party unique set-disjointness problem, where player i gets a set of size $|I_i|$, and where each player sends s bits. If for example we take the intervals to be of equal size $|I_1| = \ldots = |I_k| = t/k$, then using existing tight lower bounds for multi-party unique set-disjointness, one can prove tight sample-space lower bounds in the adversarial model[2]. This was the approach taken by many of the previous works in this area [CKS03, BYJKS04, AMOP08, GH09, BVWY18, KPW21]. Our plan is to extend this approach to the stochastic model. However, this presents two new challenges.

First, a simple calculation shows that the number of needles is $k \approx pt$ with high probability, but the gaps between needles are not uniform; for example, the two closest needles have a gap of $\approx p^2 t$. This necessitates taking intervals of very different lengths, if we still plan to place one needle per interval. In turn, this requires proving lower bounds on multi-party unique set-disjointness when the players receive inputs of different lengths. In this model, it no longer makes sense to measure the total communication of the protocols. Instead, we develop a new measure, which normalizes the communication of each player relative to their input length. We expand on this in Section 1.3.

The second challenge is that using a single partition of the stream by intervals, and then planting a uniform needle in each interval, cannot induce the planted needle distribution. Instead, we need to carefully construct a distribution over sets of intervals, such that if then one places a uniform needle in each interval, the resulting stream distribution mimics exactly the planted distribution. We expand on this in Section 1.4.

1.3 Multi-party unique set-disjointness with different set sizes

We start by defining the standard multi-party unique set-disjointness problem. Let $k \geq 2$ denote the number of players. The players inputs are sets $S_1, \ldots, S_k \subset [n]$. They are promised that one of two cases hold:

- **Disjoint**: the sets $S_1, \ldots, S_k$ are pairwise disjoint.
- **Unique intersection**: there is a common element $x \in S_1 \cap \ldots \cap S_k$, and the sets $S_1 \setminus \{x\}, \ldots, S_k \setminus \{x\}$ are pairwise disjoint.

[2] Concretely, the total communication of the protocol is ks, whereas the lower bound for k-party unique set-disjointness is $\Omega(t/k)$. Thus $ks = \Omega(t/k)$. Taking $k = pt$ gives $p^2 st = \Omega(1)$.

Their goal is to distinguish which case is it, while minimizing the communication[3].

Observe that under any of the two promise cases, one of the players' inputs has size $|S_i| \leq n/k + 1$. A simple protocol is that such a player sends their input, which allows the other players to solve the problem on their own. This simple protocol sends $O(n/k \cdot \log n)$ bits. This can be further improved to $O(n/k)$ bits using the techniques of [HW07]. A line of research [AMS99, BYJKS04, CKS03, Gro09, Jay09, YZ22] studied lower bounds. A tight lower bound was first achieved by [Gro09].

Theorem 1.3 ([Gro09, Jay09]). *Any randomized communication protocol which solves the k-party unique set-disjointness problem must send $\Omega(n/k)$ bits.*

As discussed in Section 1.2, we need a fine-grained variant of the unique set-disjointness problem, where the set sizes are fixed and can be different between the players.

Definition 1.4 (Fixed-size multi-party unique set-disjointness). *Let $s_1, \ldots, s_k \geq 1$. The $[s_1, \ldots, s_k]$-size k-party unique set-disjointness problem is a restriction of the k-party unique set-disjointness problem to input sets of size $|S_i| = s_i$.*

Consider protocols for the $[s_1, \ldots, s_k]$-size k-party unique set-disjointness problem. For any $i \in [k]$, one option is that the i-th player sends their input to the rest of the players, which requires sending $c_i = \Omega(s_i)$ bits. If the input sizes $s_1, \ldots, s_k$ are very different, it no longer makes sense to consider the total amount of bits sent by the players. Instead, we should normalize the number of bits sent by the i-th player c_i by its input length s_i. We prove that with this normalization, the simple protocols are indeed optimal.

Towards this, we make the following definition: a k-party protocol Π is called $[c_1, \ldots, c_k]$-bounded if in any transcript of Π, the i-th player sends at most c_i bits.

Theorem 1.5 (Lower bound for fixed-size multi-party unique set-disjointness). *Let Π be a randomized k-party $[c_1, \ldots, c_k]$-bounded protocol, which solves with high probability the $[s_1, \ldots, s_k]$-size k-party unique set-disjointness problem, where $\sum s_i \leq n/2$. Then*

$$\sum_{i \in [k]} \frac{c_i}{s_i} = \Omega(1).$$

We conclude this subsection with three comments. First, the condition $\sum s_i \leq n/2$ is a technical condition emerging from the proof technique; it suffices for our application, and we believe that it can be removed in future work.

Next, it is known that the hard case for the standard multi-party unique set-disjointness problem is when all the sets have about the same size, namely when $s_1 = \ldots = s_k = \Theta(n/k)$. In this case Theorem 1.5 implies $\sum c_i = \Omega(n/k)$ which recovers Theorem 1.3.

Last, we prove Theorem 1.5 by constructing a hard distribution over inputs, and then proving a lower bound for deterministic protocols under this distribution. The

[3] Formally, we consider randomized multi-party protocols in the *blackboard model*, where at each turn one of the players writes a message on a common blackboard seen by all the players.

hard distribution is a natural one, the uniform distribution over inputs of sizes $s_1, \dots, s_k$. For details see Theorem 2.13. Moreover, we show (Claim 2.15) that Theorem 1.5 and Theorem 2.13 are in fact equivalent.

1.4 Efficient reduction of the needle problem to multi-party unique set-disjointness

We establish Theorem 1.2 by reducing lower bounds for the needle problem to lower bounds for the unique set-disjointness, and then applying Theorem 1.5 (Theorem 2.13 more precisely). To do so, we need a way of mapping inputs to the unique set-disjointness problem to inputs for a streaming algorithm. A natural way to do so, taken for example by [AMOP08], is to partition the stream into intervals and assign one to each player. We follow the same approach but generalize it, so we can use it to simulate the planted distribution of the needle problem by random inputs to the unique set-disjointness problem.

Recall that n denotes the domain size, t the number of samples and p the needle probability. Our goal will be to simulate the planted distribution using inputs to multi-party unique set-disjointness. In order to do so, we define *interval systems*.

Definition 1.6 (Interval systems)**.** *An interval system F is a family of pairwise disjoint non-empty intervals $F = \{I_1, \dots, I_k\}$ with $I_1, \dots, I_k \subset [t]$.*

Given an interval system F, we define a planted distribution Planted$[F]$ over streams $X \in [n]^t$ as follows:

1. Sample uniform needle $x \in [n]$;
2. In each interval $I \in F$ sample uniform index $i \in I$ and set $X_i = x$;
3. Sample all other stream elements uniformly from $[n]$.

Using Theorem 1.5, we prove a space lower bound for streaming algorithms that can distinguish between the uniform distribution and the planted distribution for F. Here is where we exploit the fact that we can prove lower bounds for unique set-disjointness also when the set sizes vary between the players. We use the following notation: given an interval system F, its value is $\text{val}(F) = \sum_{I \in F} \frac{1}{|I|}$.

Lemma 1.7. *Let F be an interval system. Any streaming algorithm which with high probability distinguishes between Planted$[F]$ and the uniform distribution must use space*

$$s = \Omega\left(\frac{1}{\mathit{val}(F)}\right).$$

In order to complete the reduction, we need to simulate the planted distribution using planted distributions for interval systems F. Clearly, this cannot be done using a single interval system, and hence we need to consider *randomized* interval systems.

A randomized interval system $\mathcal{F}$ is a distribution over interval systems F. The planted distribution $\text{Planted}[\mathcal{F}]$ for $\mathcal{F}$ is defined by first sampling $F \sim \mathcal{F}$ and then $X \sim \text{Planted}[F]$. The value of $\mathcal{F}$ is $\text{val}(\mathcal{F}) = \mathbb{E}_{F\sim\mathcal{F}}[\text{val}(F)]$. We can extend Lemma 1.7 to randomized interval systems.

Lemma 1.8. *Let $\mathcal{F}$ be a randomized interval system. Any streaming algorithm which with high probability distinguishes between $\text{Planted}[\mathcal{F}]$ and the uniform distribution must use space*

$$s = \Omega\left(\frac{1}{\mathit{val}(\mathcal{F})}\right).$$

To prove the lower bound for the needle problem, we need $\text{Planted}[\mathcal{F}]$ to simulate exactly the planted distribution; we call such randomized interval systems *perfect*.

Definition 1.9 (Perfect randomized interval systems)**.** *A randomized interval system $\mathcal{F}$ is called* perfect *if $\text{Planted}[\mathcal{F}]$ is distributed exactly as the planted distribution.*

In light of Lemma 1.8, we need a perfect randomized interval system $\mathcal{F}$ with as low a value as possible. It is relatively simple to show that if $\mathcal{F}$ is perfect then $\text{val}(\mathcal{F}) = \Omega(p^2 t)$. The following theorem gives a construction nearly matching the lower bound.

Theorem 1.10. *There exists a perfect randomized interval system $\mathcal{F}$ with $\mathit{val}(\mathcal{F}) = O\left(p^2 t \log(t)\right)$.*

Theorem 1.2 now follows directly by combining Lemma 1.8 and Theorem 1.10.

1.5 Related works

In a seminal work, Miltersen et al. [MNSW95] first observed connections between asymmetric communication complexity and its applications to data structures in the cell probe model. Since then, several works [BR00, JKKR03, PT06, BIPW10, CKLM18] proved data structure lower bounds and streaming lower bounds via connections to asymmetric communication complexity lower bounds. To the best of our knowledge, all these works built on two-party communication problems. In contrast, we consider multi-party communication complexity in this work. It is interesting to ask if multi-party communication can provide more applications to data structure and streaming lower bounds.

Other than connections to data structure lower bounds and streaming lower bounds, Dinur et al. [DDKS16] studied the needle problem in cryptography. It would be interesting to explore more connections between our work and cryptography.

Acknowledgements. We thank David Woodruff for helpful discussions about streaming algorithms, and for insightful comments on an earlier version of this paper.

Paper organization. We prove lower bounds for multi-party unique set-disjointness with fixed set sizes (Theorem 1.5) in Section 2. We design an efficient reduction using interval systems (Lemmas 1.7 and 1.8) in Section 3. We combine both to prove our lower bound for the needle problem (Theorem 1.2) in Section 4. We discuss open problems in Section 5.

2 Lower bounds for asymmetric unique set-disjointness

We prove Theorem 1.5 in this section. First, we recall some definitions and fix some notations.

Notations. it will be convenient to identify sets with their indicator vectors; thus, we identify $X \in \{0,1\}^n$ with the set $\{i : X_i = 1\} \subset [n]$. Let $k \geq 2$ denote the number of players. The players inputs are $X = (X_1, \ldots, X_k)$, where $X_i = (X_i(1), \ldots, X_i(n)) \in \{0,1\}^n$. It will be convenient to also define $X^j = (X_1(j), \ldots, X_k(j)) \in \{0,1\}^k$, the j-th coordinate for all the players for $j \in [n]$. In this section use boldface to denote random variables (such as $\boldsymbol{X}, \boldsymbol{W}$) to help distinguish them from non-random variables.

Protocols. Let Π be a protocol. Given an input X, we denote by $\Pi(X)$ the transcript of running Π on X. We assume that every transcript also has an output value which is a bit determined by the transcript (for example, the last bit sent). A protocol solves a decision problem under input distribution ν with error δ, if it outputs the correct answer with probability at least $1 - \delta$ when the inputs are sampled from ν. We will prove lower bounds on protocols that solve unique set-disjointness under a number of input distributions. As such, we may assume unless otherwise specified that the protocols are deterministic.

Finally, recall that we call k-party protocol Π is called $[c_1, \ldots, c_k]$-bounded if in any transcript of Π, the i-th player sends at most c_i bits.

multi-party unique set-disjointness. The k-party unique set-disjointness problem is defined on inputs coming from two promise sets:

- **Disjoint**: $\mathcal{F}^0 = \{X \in (\{0,1\}^n)^k : \forall j \in [n], |X^j| \leq 1\}$,
- **Unique intersection**: $\mathcal{F}^1 = \{X \in (\{0,1\}^n)^k : \exists j \in [n], |X^j| = k, \forall j' \neq j, |X^{j'}| \leq 1\}$.

Towards proving Theorem 1.5, our first step is to consider unique set-disjointness under product distribution which assign weight asymmetrically between the players.

2.1 Lower bounds for product asymmetric distributions

Let ν be a distribution over $[k]$. We denote by ν^n the distribution over $\boldsymbol{W} \in [k]^n$, where we sample $\boldsymbol{W}_j \sim \nu$ independently for all $j \in [n]$. We define two distributions, $\mu^0_{\text{prob}}[\nu]$ supported on $\mathcal{F}^0$ and $\mu^1_{\text{prob}}[\nu]$ supported on $\mathcal{F}^1$.

Definition 2.1 (Disjoint asymmetric distribution). *Let $\boldsymbol{X} \in (\{0,1\}^n)^k$ be sampled as follows:*

1. *Sample $\boldsymbol{W} \sim \nu^n$.*
2. *For each $j \in [n]$, if $\boldsymbol{W}_j = i$ then we sample $\boldsymbol{X}_i(j) \in \{0,1\}$ uniformly, and set $\boldsymbol{X}_{i'}(j) = 0$ for all $i' \neq i$.*

We denote by $\mu^0_{prob}[\nu]$ the marginal distribution of $\boldsymbol{X}$, and note that it is supported on $\mathcal{F}^0$.

Definition 2.2 (Unique intersection asymmetric distribution)**.** *Let $\boldsymbol{Y} \in (\{0,1\}^n)^k$ be sampled as follows:*

1. *Sample $\boldsymbol{X} \sim \mu^0_{prob}[\nu]$.*
2. *Sample $\boldsymbol{j} \in [n]$ uniformly.*
3. *If $\boldsymbol{j} = j$ then we set $\boldsymbol{Y}^j = 1^k$ and $\boldsymbol{Y}^{j'} = \boldsymbol{X}^{j'}$ for all $j' \neq j$.*

We denote by $\mu^1_{prob}[\nu]$ the marginal distribution of $\boldsymbol{Y}$, and note that it is supported on $\mathcal{F}^1$.

We denote by $\mu_{\text{prob}}[\nu]$ the mixture distribution, where we sample $\boldsymbol{b} \in \{0,1\}$ uniformly, and then sample $\boldsymbol{X} \sim \mu^{\boldsymbol{b}}_{\text{prob}}[\nu]$. Our main technical result is a communication lower bound on protocols which solve unique set-disjointness under input distribution $\mu_{\text{prob}}[\nu]$. We will later reduce the fixed set size case to this model.

Theorem 2.3. *Fix $n, k \geq 1$. Let ν be a distribution on $[k]$. Let Π be a $[c_1, \ldots, c_k]$-bounded k-party deterministic protocol which solves the unique set-disjointness problem under input distribution $\mu_{prob}[\nu]$ with error 2%. Then*

$$\sum_{i \in [k]} \frac{c_i}{\nu(i)} = \Omega(n).$$

We note that Theorem 2.3 is a generalization of the lower bound for symmetric case [Gro09, Jay09], where $\nu(i) = 1/k$ for all $i \in [k]$. In this case Theorem 2.3 gives that $\sum_i c_i = \Omega(n/k)$.

2.1.1 Information theory framework

We will use information theory to prove Theorem 2.3. Although we assume that Π has small error with respect to both $\mu^0_{\text{prob}}[\nu]$ and $\mu^1_{\text{prob}}[\nu]$, we will only study its information complexity with respect to $\mu^0_{\text{prob}}[\nu]$. Below we let $\boldsymbol{W} \in [k]^n, \boldsymbol{X} \in (\{0,1\})^n$ be jointly samples as in Definition 2.1. The following observation will play an important role.

Observation 2.4. *Conditioned on $\boldsymbol{W} = W$, the random variables $(\boldsymbol{X}_i(j) : i \in [k], j \in [n])$ are independent.*

We start by giving a general bound for individual communication based on information theory, which assumes only the existence of such $\boldsymbol{W}$ under which $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k$ are independent.

Lemma 2.5. *Let Π be a k-party protocol which is $[c_1, \ldots, c_k]$-bounded. Assume joint random variables $(\boldsymbol{W}, \boldsymbol{X})$, where $\boldsymbol{X} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k)$ are the players inputs, and such that for every value W for $\boldsymbol{W}$, the random variables $\boldsymbol{X}_1|\boldsymbol{W} = W, \ldots, \boldsymbol{X}_k|\boldsymbol{W} = W$ are independent. Then for each $i \in [k]$ we have*

$$c_i \geq I(\boldsymbol{X}_i : \Pi(\boldsymbol{X})|\boldsymbol{W}).$$

Proof. We first set up some notations. We denote by π a possible transcript for Π, and let $\pi_{<t} = (\pi_1, \ldots, \pi_{t-1})$ be a partial transcript. We let $\boldsymbol{\pi} = \Pi(\boldsymbol{X})$ denote the transcript when the protocol is run on $\boldsymbol{X}$.

Fix a time step t in the protocol, and a partial transcript $\pi_{<t}$. The next player to speak is determined by the transcript so far, so denote it by $\text{next}(\pi_{<t}) \in [k]$. We also denote by $\text{locs}(\pi, i) = \{t : \text{next}(\pi_{<t}) = i\}$ the locations in transcript π where player i sent a bit. By our assumption $|\text{locs}(\pi, i)| \leq c_i$ for any transcript π.

Consider any value W for $\boldsymbol{W}$. Observe that conditioned on $\boldsymbol{\pi}_{<t} = \pi_{<t}$, the next bit sent $\boldsymbol{\pi}_t$ is a function of $\boldsymbol{X}_i$ for $i = \text{next}(\pi_{<t})$. If $i' \neq i$ then since $\boldsymbol{X}_i | \boldsymbol{W} = W, \boldsymbol{X}_{i'} | \boldsymbol{W} = W$ are independent we have

$$I(\boldsymbol{X}_{i'} : \boldsymbol{\pi}_t | \boldsymbol{W} = W, \boldsymbol{\pi}_{<t} = \pi_{<t}) = 0.$$

Since $\boldsymbol{\pi}_t \in \{0, 1\}$, we can also trivially bound

$$I(\boldsymbol{X}_i : \boldsymbol{\pi}_t | \boldsymbol{W} = W, \boldsymbol{\pi}_{<t} = \pi_{<t}) \leq 1.$$

Averaging over $\pi_{<t}$ and W gives

$$I(\boldsymbol{X}_i : \boldsymbol{\pi}_t | \boldsymbol{W}, \boldsymbol{\pi}_{<t}) \leq \Pr[\text{next}(\boldsymbol{\pi}_{<t}) = i].$$

Summing over t then gives the result:

$$I(\boldsymbol{X}_i : \boldsymbol{\pi} | \boldsymbol{W}) = \sum_t I(\boldsymbol{X}_i : \boldsymbol{\pi}_t | \boldsymbol{W}, \boldsymbol{\pi}_{<t}) = \mathbb{E}|\text{locs}(\boldsymbol{\pi}, i)| \leq c_i.$$

□

We shorthand $\boldsymbol{\pi} = \Pi(\boldsymbol{X})$ below. Using Lemma 2.5, Observation 2.4 and the data processing inequality[4] give

$$c_i \geq I(\boldsymbol{X}_i : \boldsymbol{\pi} | \boldsymbol{W}) \geq \sum_{j \in [n]} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W}).$$

Towards proving Theorem 2.3, consider the expression

$$\sum_{i \in [k]} \frac{c_i}{\nu(i)} \geq \sum_{i \in [k]} \frac{1}{\nu(i)} I(\boldsymbol{X}_i : \boldsymbol{\pi} | \boldsymbol{W}) \geq \sum_{i \in [k]} \frac{1}{\nu(i)} \sum_{j \in [n]} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W})$$

We define below

$$L := \frac{1}{n} \sum_{i \in [k]} \frac{1}{\nu(i)} \sum_{j \in [n]} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W})$$

The following lemma thus proves Theorem 2.3.

Lemma 2.6. $L = \Omega(1)$.

We prove Lemma 2.6 in the next subsection, via a reduction to protocols for the k-bit AND function.

[4] If $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ are random variables, where $\boldsymbol{x}, \boldsymbol{y}$ are independent, then $I(\boldsymbol{xy} : \boldsymbol{z}) \geq I(\boldsymbol{x} : \boldsymbol{z}) + I(\boldsymbol{y} : \boldsymbol{z})$.

2.1.2 Reduction to the information complexity of the AND function

In this section, we consider the k-bit AND function and its information complexity. Let Λ be a k-party protocol for it: each of the k players receive as input a bit, and their goal is to compute their AND. Namely, to check if they are all equal to 1.

Let $\boldsymbol{b} \in \{0,1\}$ be a random bit. For $i \in [k]$, let $e_i[\boldsymbol{b}] \in \{0,1\}^k$ denote the vector with $\boldsymbol{b}$ at coordinate i and 0 everywhere else. The following lemma reduces proving Lemma 2.6 to analyzing the information of protocols for k-bit AND which make small error on only two inputs: the all-zero and all-one inputs.

Lemma 2.7. *There is a public-randomness k-party protocol Λ for the k-bit AND function, using public-randomness $\boldsymbol{R}$, with the following guarantees:*

1. *Λ has error at most 8% with respect to the inputs 0^k and 1^k.*
2. *$L = \sum_{i\in[k]} I(\boldsymbol{b}, \Lambda(e_i[\boldsymbol{b}], \boldsymbol{R})|\boldsymbol{R})$.*

We prove Lemma 2.7 in the remainder of this subsection. First, let $\boldsymbol{d} \in [k], \boldsymbol{U} \in \{0,1\}^k$ be jointly sampled as follows:

1. Sample $\boldsymbol{d} \in [k]$ according to ν.
2. Given $\boldsymbol{d} = d$, sample $\boldsymbol{U}_d \in \{0,1\}$ uniformly and set $\boldsymbol{U}_i = 0$ for all $i \neq d$.

Let $\sigma = \sigma(\nu)$ denote the marginal distribution of $\boldsymbol{U}$, and observe that it is the same as that of $\boldsymbol{X}^j$ for any $j \in [n]$. In fact, the joint distribution of $(\boldsymbol{d}, \boldsymbol{U})$ is the same as $(\boldsymbol{W}_j, \boldsymbol{X}^j)$ for any j. The next claim uses this to extract a protocol Λ for k-bit AND from Π, such that it has related information complexity measures, and a small error with respect to the inputs 0^k and 1^k.

Claim 2.8. *There is a (public randomness) k-party protocol Λ for the k-bit AND function, using public randomness $\boldsymbol{R}$, with the following properties:*

1. *Λ has error at most 8% with respect to the inputs 0^k and 1^k.*
2. *$I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, \boldsymbol{R})|\boldsymbol{d}, \boldsymbol{R}) = \frac{1}{n}\sum_{j=1}^{n} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi}|\boldsymbol{W})$ for all $i \in [k]$.*

Proof. We first define the protocol Λ. Let $U \in \{0,1\}^k$ denote the input for the AND function. First, using public randomness, sample $\boldsymbol{j} \in [n]$ uniformly; then sample $\boldsymbol{W}_{-\boldsymbol{j}} = (\boldsymbol{W}_{j'} : j' \neq \boldsymbol{j}) \sim \nu^{n-1}$. Conditioned on $\boldsymbol{j} = j, \boldsymbol{W}_{-j} = W_{-j}$, the i-th player then constructs their input $\boldsymbol{X}_i$ for Π as follows: set $\boldsymbol{X}_i(j) = U_i$ and sample $\boldsymbol{X}_i(j')|\boldsymbol{W}_{j'} = W_{j'}$ using private randomness. The players then run the protocol Π on their joint inputs $\boldsymbol{X} = (\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k)$. Note that the public randomness used is $\boldsymbol{R} = (\boldsymbol{j}, \boldsymbol{W}_{-\boldsymbol{j}})$.

To prove the first claim, observe that if the input to the AND function $\boldsymbol{U}$ is distributed as $\boldsymbol{U} \sim \sigma$, then $\boldsymbol{X} \sim \mu^0_{\text{prob}}[\nu]$; and if $U = 1^k$ then $\boldsymbol{X} \sim \mu^1_{\text{prob}}[\nu]$. Thus Λ has error at most 2% with respect to the uniform mixture of the input distributions σ and 1^k. Thus with respect to the input 1^k, the error is at most 4%. Since $\sigma(0^k) = 1/2$, the error with respect to the input 0^k is at most 8%.

For the second claim, note that conditioned on $\boldsymbol{R} = R = (j, W_{-j})$, the joint distribution of $(\boldsymbol{d}, \boldsymbol{U}, \Lambda(\boldsymbol{U}, R))$ and of $(\boldsymbol{W}_j, \boldsymbol{X}^j, \pi)$ is identical. Thus

$$I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, R) | \boldsymbol{d}, \boldsymbol{R} = R) = I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W}_j, \boldsymbol{j} = j, \boldsymbol{W}_{-\boldsymbol{j}} = W_{-j})$$

Averaging over R gives

$$\begin{aligned} I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, R) | \boldsymbol{d}, \boldsymbol{R}) &= \frac{1}{n} \sum_{j \in [n]} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W}_j, \boldsymbol{j} = j, \boldsymbol{W}_{-\boldsymbol{j}} = W_{-j}) \\ &= \frac{1}{n} \sum_{j \in [n]} I(\boldsymbol{X}_i(j) : \boldsymbol{\pi} | \boldsymbol{W}). \end{aligned}$$

□

Proof of Lemma 2.7. Let Λ be the protocol given by Claim 2.8. Then

$$L = \sum_{i \in [k]} \frac{1}{\nu(i)} I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, \boldsymbol{R}) | \boldsymbol{d}, \boldsymbol{R}).$$

Simplifying the inner terms give

$$\begin{aligned} \frac{1}{\nu(i)} I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, \boldsymbol{R}) | \boldsymbol{d}, \boldsymbol{R}) &= \frac{1}{\nu(i)} \sum_{j \in [k]} \nu(j) \cdot I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, \boldsymbol{R}) | \boldsymbol{d} = j, \boldsymbol{R}) \\ &= I(\boldsymbol{U}_i : \Lambda(\boldsymbol{U}, \boldsymbol{R}) | \boldsymbol{d} = i, \boldsymbol{R}) \end{aligned}$$

Note that conditioned on $\boldsymbol{d} = i$, the joint distribution of $(\boldsymbol{U}_i, \boldsymbol{U})$ is the same as $(\boldsymbol{b}, e_i[\boldsymbol{b}])$. Thus

$$L = \sum_{i \in [k]} I(\boldsymbol{b} : \Lambda(e_i[\boldsymbol{b}], \boldsymbol{R}) | \boldsymbol{R}).$$

□

2.1.3 Bounding the information complexity of AND functions

We prove the following lemma in this subsection, which then proves Theorem 2.3 given Lemma 2.5, Lemma 2.6 and Lemma 2.7.

Lemma 2.9. *Let Λ be a (public randomness) protocol for the k-bit AND function, using public randomness $\boldsymbol{R}$, such that it has error at most 8% with respect to the inputs 0^k and 1^k. Then*

$$\sum_{i \in [k]} I(\boldsymbol{b}, \Lambda(e_i[\boldsymbol{b}], \boldsymbol{R}) | \boldsymbol{R}) = \Omega(1).$$

Lemma 2.9 is very similar to previous lower bounds in the literature on information complexity [BYJKS04, CKS03, Gro09]. We need the following setup. Sample jointly $\boldsymbol{e} \in [k], \boldsymbol{V} \in \{0,1\}^k$ as follows:

1. Sample $\boldsymbol{e} \in [k]$ uniformly.
2. Given $\boldsymbol{e} = e$, sample $\boldsymbol{V}_e \in \{0,1\}$ uniformly and set $\boldsymbol{V}_i = 0$ for all $i \neq e$.

Given a protocol Λ using public randomness $\boldsymbol{R}$, its conditional information complexity is

$$\mathrm{CIC}(\Lambda) = I(\boldsymbol{V} : \Lambda(\boldsymbol{V}, \boldsymbol{R}) | \boldsymbol{e}, \boldsymbol{R}).$$

This quantity comes up naturally in the study of unique disjointness using information complexity, which started with the seminal work of [BYJKS04]. Gronemeier [Gro09] and Jayram [Jay09] proved a tight lower bound on this quantity.

Theorem 2.10. *[[Gro09, Jay09]] CIC$(\Lambda) = \Omega(1/k)$.*

In fact, the proof (although not explicitly stated as such) only relies on the assumption that Λ has error $\leq 30\%$ on both the all-zero and all-one inputs (for a full proof see Gronemeier's thesis [Gro10]). As such, it applies to our protocol Λ. The following claim connects CIC(Λ) to the quantity we aim to bound, and concludes the proof of Lemma 2.9 and hence also of Theorem 2.3.

Claim 2.11. $\sum_{i \in [k]} I(\boldsymbol{b} : \Lambda(e_i(\boldsymbol{b}), \boldsymbol{R}) | \boldsymbol{R}) = k \cdot CIC(\Lambda)$.

Proof.

$$\begin{aligned} k \cdot \mathrm{CIC}(\Lambda) &= k \cdot I(\boldsymbol{V} : \Lambda(\boldsymbol{V}, \boldsymbol{R}) | \boldsymbol{e}, \boldsymbol{R}) \\ &= \sum_{i \in [k]} I(\boldsymbol{V} : \Lambda(\boldsymbol{V}, \boldsymbol{R}) | \boldsymbol{e} = i, \boldsymbol{R}) \\ &= \sum_{i \in [k]} I(\boldsymbol{b} : \Lambda(e_i(\boldsymbol{b}), \boldsymbol{R}) | \boldsymbol{R}). \end{aligned}$$

□

2.2 Extension to sub-distributions

It will be convenient to extend Theorem 2.3 to sub-distributions. A sub-distribution ν on $[k]$ satisfies $\nu(i) \geq 0$ and $\sum \nu(i) \leq 1$. We extend the definition of $\mu^0_{\text{prob}}[\nu]$, $\mu^1_{\text{prob}}[\nu]$ to sub-distributions as follows.

We first describe how to sample $\boldsymbol{X} \sim \mu^0_{\text{prob}}[\nu]$. For each $j \in [n]$, with probability $\nu(i)$ sample $\boldsymbol{X}_i(j) \in \{0,1\}$ uniformly, and set $\boldsymbol{X}_{i'}(j) = 0$ for all $i \neq i'$; and with probability $1 - \sum \nu(i)$ set $\boldsymbol{X}_i(j) = 0$ for all i. To sample $\boldsymbol{Y} \sim \mu^1_{\text{prob}}[\nu]$ we follow the same process as for the distributional case: first sample $\boldsymbol{X} \sim \mu^0_{\text{prob}}[\nu]$, then sample a uniform $\boldsymbol{j} \in [n]$ and set $\boldsymbol{Y}^{\boldsymbol{j}} = 1^k$ and $\boldsymbol{Y}^{j'} = \boldsymbol{X}^{j'}$ for all $j' \neq j$. We denote by $\mu_{\text{prob}}[\nu]$ the even mixture of $\mu^0_{\text{prob}}[\nu]$ and $\mu^1_{\text{prob}}[\nu]$. The following theorem extends Theorem 2.3 to sub-distributions.

Theorem 2.12. *Fix $n, k \geq 1$. Let ν be a sub-distribution on $[k]$. Let Π be a $[c_1, \ldots, c_k]$-bounded protocol which solves the distributional unique set-disjointness under input distribution $\mu_{prob}[\nu]$ with error 2%. Then*

$$\sum_{i \in [k]} \frac{c_i}{\nu(i)} = \Omega(n).$$

Proof. Extend ν to a distribution ν' on $[k+1]$ by setting $\nu'(i) = \nu(i)$ for $i \in [k]$ and $\nu'(k+1) = 1 - \sum \nu(i)$. Extend Π to a protocol Π' for $k+1$ players where player $k+1$ does not participate in the protocol at all. Thus Π' is a $[c_1, \ldots, c_k, 0]$-bounded protocol. The proof follows by applying Theorem 2.3 to Π' and ν'. □

2.3 Extension for fixed set sizes

We now use the results we proven to deduce Theorem 1.5. Namely, the lower bound for fixed set sizes. We first set some notations.

Let $\mathfrak{s} = [s_1, \ldots, s_k]$ denote the set sizes where $s_i \geq 1$ and $\sum s_i \leq n$. Define

$$\mathcal{F}_{\text{size}}[\mathfrak{s}] = \{X \in (\{0,1\}^n)^k : \forall i \in [k], |X_i| = s_i\}.$$

For $b \in \{0,1\}$ define $\mathcal{F}^b_{\text{size}}[\mathfrak{s}] = \mathcal{F}^b \cap \mathcal{F}_{\text{size}}[\mathfrak{s}]$ and $\mu^b_{\text{size}}[\mathfrak{s}]$ to be the uniform distribution over $\mathcal{F}^b_{\text{size}}[\mathfrak{s}]$. Our hard distribution $\mu_{\text{size}}[\mathfrak{s}]$ will be an even mixture between $\mu^0_{\text{size}}[\mathfrak{s}]$ and $\mu^1_{\text{size}}[\mathfrak{s}]$. Equivalently, sample $\boldsymbol{b} \in \{0,1\}$ uniformly and take $\boldsymbol{X} \sim \mu^{\boldsymbol{b}}_{\text{size}}[\mathfrak{s}]$. We prove a communication lower bound on protocols which solve unique set-disjointness under input distribution $\mu_{\text{size}}[\mathfrak{s}]$.

Theorem 2.13. *Let $\mathfrak{s} = [s_1, \ldots, s_k]$ with $\sum s_i \leq n/2$. Let Π be a $[c_1, \ldots, c_k]$-bounded k-party protocol which solves the unique set-disjointness problem under input distribution $\mu_{size}[\mathfrak{s}]$ with error 1%. Then*

$$\sum_{i \in [k]} \frac{c_i}{s_i} = \Omega(1).$$

It is clear that Theorem 2.13 implies Theorem 1.5, but in fact they are equivalent. Before proving it we need the following claim.

Claim 2.14. *Let $b \in \{0,1\}$, $X \in \mathcal{F}^b_{size}[\mathfrak{s}]$. Let $\boldsymbol{\Sigma}$ be a random permutation of $[n]$ and let $\boldsymbol{\Sigma}(X)$ denote the result of applying $\boldsymbol{\Sigma}$ to X. Then $\boldsymbol{\Sigma}(X)$ is uniform in $\mathcal{F}^b_{size}[\mathfrak{s}]$.*

Proof. The claim follows as permutations on $[n]$ act transitively on $\mathcal{F}^b_{\text{size}}[\mathfrak{s}]$. Namely, for any $X, X' \in \mathcal{F}^b_{\text{size}}[\mathfrak{s}]$ there exists a permutation Σ on $[n]$ such that $\Sigma(X) = X'$. This implies that a uniform permutation maps X to a uniform element in the domain $\mathcal{F}^b_{\text{size}}[\mathfrak{s}]$. □

Claim 2.15. *Theorem 1.5 and Theorem 2.13 are equivalent.*

Proof. We are comparing the multi-party unique set-disjointess problem for sizes $\mathfrak{s} = [s_1, \ldots, s_k]$ in two settings: worst-case inputs, and uniform inputs. Clearly, a protocol for worst-case inputs implies one under uniform inputs with the same communication and error guarantees. In the other direction, let $X \in \mathcal{F}^b_{\text{size}}[\mathfrak{s}]$ be any input for unique set-disjointness. The players, using public randomness, sample a uniform permutation $\mathbf{\Sigma}$ on $[n]$, and each applies it to their input. By Claim 2.14 we know that $\mathbf{\Sigma}(X)$ is distributed as $\mu^b_{\text{size}}[\mathfrak{s}]$. They can now apply a protocol that solves unique-set disjointness under input $\mu_{\text{size}}[\mathfrak{s}]$. □

We now turn to prove Theorem 2.13.

Proof of Theorem 2.13. First, note that may assume $c_i \geq 1$ for all i, since we can remove players with $c_i = 0$ from the game, as they are not allowed to send any bits.

Let Π be a protocol as assumed in Theorem 2.13. Namely, it is $[c_1, \dots, c_k]$-bounded and solves unique set-disjointness under input distribution $\mu_{\text{size}}[\mathfrak{s}]$ with error 1%, where $\mathfrak{s} = [s_1, \dots, s_k]$ satisfies $\sum s_i \leq n/2$. We will use it to design a $[c_1 + 1, \dots, c_k + 1]$-bounded protocol Π' which solves unique set-disjointness in a specific sub-distributional case with error 2%, and then appeal to Theorem 2.12.

Next, define a sub-distribution ν on $[k]$ by $\nu(i) = \frac{s_i}{4n}$. We consider its corresponding distributional input $\mu_{\text{prob}}[\nu]$ on inputs of size $n/2$ bits. Let $\boldsymbol{X} = (\boldsymbol{X}_1, \dots, \boldsymbol{X}_k) \sim \mu_{\text{prob}}[\nu]$ where $\boldsymbol{X} \in (\{0,1\}^{n/2})^k$. Each $\boldsymbol{X}_i$ is distributed Binomially $\text{Bin}(n/2, \nu(i))$ with expected size $\mathbb{E}[|\boldsymbol{X}_i|] = \frac{s_i}{2}$. Thus by the Hoeffding bound,

$$\Pr[|\boldsymbol{X}_i| > s_i] \leq \exp(-s_i/6).$$

Let E denote the event that $|\boldsymbol{X}_i| > s_i$ for some $i \in [k]$. Then

$$\Pr[E] \leq \sum_{i \in [k]} \exp(-s_i/6).$$

We first analyze the case that $\Pr[E] \geq 1\%$. In this case, since $c_i \geq 1$ by assumption, and since $\frac{1}{x} \geq C \exp(-x/6)$ for some absolute constant $C > 0$ for all $x \geq 1$, we get

$$\sum_{i \in [k]} \frac{c_i}{s_i} \geq C \sum_{i \in [k]} \exp(-s_i/6) \geq C \Pr[E] = \Omega(1).$$

From now on we assume $\Pr[E] < 1\%$.

We now design the protocol Π'. First, each player checks if their input X_i satisfies $|X_i| > s_i$. If so, the protocol aborts. This requires each player to send one bit, and by assumption it aborts with probability at most 1%. Otherwise, each player extends their input X_i to a new input $Y_i \in \{0,1\}^n$ of size $|Y_i| = s_i$ as follows.

Before the protocol starts, the players agree ahead of time on pairwise disjoint subsets $T_1, \dots, T_k$ with $|T_i| = s_i$, supported in the last $n/2$ coordinates (so they do not overlap the inputs $X_1, \dots, X_k$). Now, the i-th player adds arbitrary $s_i - |X_i|$ elements from T_i to their set X_i; we denote the new input $Y_i \in \{0,1\}^n$. Note that $Y = (Y_1, \dots, Y_k)$ satisfies the same promise as $X = (X_1, \dots, X_k)$; namely, either they are pairwise disjoint, or they have a common element and except for it they are pairwise disjoint.

We would like to apply Π to Y. However we cannot quite yet; while it is true that $Y \in \mathcal{F}^0_{\text{size}}[\mathfrak{s}]$ or $Y \in \mathcal{F}^1_{\text{size}}[\mathfrak{s}]$, its distribution is not uniform in the sets. However, here we can apply Claim 2.14 to make the distribution of Y uniform in the respective family. The players use public randomness to sample a permutation $\mathbf{\Sigma}$ on $[n]$ and apply it to Y. Now we can apply $\Pi(\mathbf{\Sigma}(Y))$ which would give the correct with error 2% by assumption. The proof now follows from Theorem 2.12. □

3 Interval systems

Recall that our plan is to use the lower bounds for multi-party unique set-disjointness in order to prove lower bounds for streaming algorithms for the needle problem. In order to effectively embed the inputs for unique set-disjointness inside streams, we introduce a combinatorial construct that we call *interval systems*.

Definition 3.1 (Interval). *An interval is a non-empty set of the form $I = \{a, a+1, \ldots, b\}$ for some $a \leq b$.*

Definition 3.2 (Interval systems). *A $[t]$-interval system is a set $F = \{I_1, \ldots, I_k\}$ of k pairwise disjoint intervals supported in $[t]$. If we want to specify the number of intervals, we say F is a $[t,k]$-interval system.*

Definition 3.3 (Randomized interval systems). *A randomized $[t]$-interval system $\mathcal{F}$ is a distribution over $[t]$-interval systems F. Similarly, a randomized $[t,k]$-interval system $\mathcal{F}$ is a distribution over $[t,k]$-interval systems F.*

Next, we define for an interval system a corresponding distribution over sets $T \subset [t]$.

Definition 3.4 (Set distribution for interval systems). *Let F be a $[t]$-interval system. We denote by Sets(F) the distribution over sets $T \subset [t]$ obtained by choosing uniformly one element from each interval $I \in F$.*

If $\mathcal{F}$ is a randomized $[t]$-interval system, then we define Sets$(\mathcal{F})$ as follows: first sample $F \sim \mathcal{F}$ and then sample $T \sim$ Sets(F).

Observe that if $\mathcal{F}$ is a randomized $[t,k]$-interval system, then Sets$(\mathcal{F})$ is a distribution over k-sets in $[t]$ (a k-set is a set of size k). Our goal will be to simulate the uniform distribution over k-sets in $[t]$. We call such randomized interval systems *perfect*.

Definition 3.5 (Perfect interval systems). *A randomized $[t,k]$-interval system $\mathcal{F}$ is called* perfect *if Sets$(\mathcal{F})$ is the uniform distribution over all k-sets in $[t]$.*

There are many ways to construct perfect randomized $[t,k]$-interval systems. For example, a naive way is to sample k uniform coordinates $i_1, \ldots, i_k \in [t]$, and then take the distribution over $F = \{\{i_1\}, \ldots, \{i_k\}\}$. However, for an efficient reduction, we would need interval systems with as long intervals as possible. Technically, the efficiency of the reduction will be controlled by the following notion of *value* of interval systems.

Definition 3.6 (Value of interval systems)**.** *Let F be a $[t]$-interval system. Its value is*

$$\mathit{val}(F) = \sum_{I \in F} \frac{1}{|I|}.$$

If $\mathcal{F}$ is a randomized $[t]$-interval system then its value is

$$\mathit{val}(\mathcal{F}) = \mathbb{E}_{F \sim \mathcal{F}}\left[\mathit{val}(F)\right].$$

In order to prove strong lower bounds on streaming algorithms, we would need a perfect distribution over $[t,k]$-intervals with as low a value as possible. The following claim gives a lower bound for this.

Claim 3.7. *Let F be a $[t,k]$-interval system. Then*

$$\mathit{val}(F) \geq \frac{k^2}{t}.$$

Proof. Let $F = \{I_1, \ldots, I_k\}$ where $|I_i| = s_i$. We have $\sum s_i \leq t$, and $\mathrm{val}(F) = \sum \frac{1}{s_i}$. This expression is minimized when all the s_i are the equal, and hence

$$\mathrm{val}(F) \geq k \cdot \frac{k}{\sum s_i} \geq \frac{k^2}{t}.$$

□

Our main technical result in this section is a construction of a perfect randomized $[t,k]$-interval system with value close to optimal. We do so by designing a randomized algorithm that samples $[t,k]$-interval systems. We will show that its output distribution is perfect, and of value close to the minimum given by Claim 3.7.

It will be convenient to make the following definition of "shifting" an interval or an interval system. For an interval $I = [a,b]$ and an integer c, define $I + c = [a+c, b+c]$. For an interval system $F = \{I_1, \ldots, I_k\}$ define $F + c = \{I_1 + c, \ldots, I_k + c\}$.

Algorithm 1: SampleIntervalSystem

```
Input: $t \geq 1$, $k \geq 0$ with $k \leq t$
Output: $[t,k]$-interval system $F$
if $k = 0$ then
    return $F = \{\}$
else if $k = 1$ then
    return $F = \{[t]\}$
else
    Let $s = \lceil t/2 \rceil$
    Sample $\boldsymbol{j} \in \{0, \ldots, k\}$ with probability $\Pr[\boldsymbol{j} = j] = \frac{\binom{s}{j}\binom{t-s}{k-j}}{\binom{t}{k}}$
    Compute $F_1 = \mathrm{SampleIntervalSystem}(s, \boldsymbol{j})$
    Compute $F_2 = \mathrm{SampleIntervalSystem}(t-s, k-\boldsymbol{j})$
    return $F = F_1 \cup (F_2 + s)$
end
```

We denote by $\mathcal{F}[t,k]$ the randomized $[t,k]$-interval system obtained by running SampleIntervalSystem(t,k).

Claim 3.8. *$\mathcal{F}[t,k]$ is perfect.*

Proof. The proof is by induction on k, t. If $k = 0$ or $k = 1$ this is clear from the base cases of the algorithm. If $k \geq 2$, then we sample the number of elements j in the interval $[s]$ with the same probability as a uniform k-set in $[t]$ would. By induction, the distribution $\mathcal{F}[s,j]$ of F_1 is a perfect randomized $[s,j]$ interval system; and the distribution $\mathcal{F}[t-s,k-j]$ of F_2 is a perfect randomized $[t-s,k-j]$ interval system. The claim follows. □

We next analyze the value of $\mathcal{F}[t,k]$; to simplify the analysis, we restrict to the case t is a power of two. This suffices for our application, and we expect the bound to extend to general t with minimal modifications. We assume below that all logarithms are in base two.

Lemma 3.9. *Assume t is a power of two. Then val$(\mathcal{F}[t,k]) \leq \frac{k^2 \log(2t)}{t}$.*

In order to prove Lemma 3.9, we will need the following technical claim, computing first and second moments for the distribution over $\boldsymbol{j}$ in the algorithm.

Claim 3.10. *Let $t, k \geq 1$, t even, and $0 \leq j \leq k$. Define $p(t,k,j) = \frac{\binom{t/2}{j}\binom{t/2}{k-j}}{\binom{t}{k}}$. Then*

$$\sum_{j=0}^{k} p(t,k,j) \cdot j = \frac{k}{2}$$

and

$$\sum_{j=0}^{k} p(t,k,j) \cdot j^2 \leq \frac{k(k+1)}{4}.$$

Proof. Let $s = t/2$. Let T be a uniform subset of $[t]$ of size k. Then $p(t,k,j) = \Pr[|T \cap [s]| = j]$. Hence

$$\sum_{j=0}^{k} p(t,k,j) \cdot j = \mathbb{E}_T\left[\sum_{i \in [s]} \mathbf{1}[i \in T]\right] = \sum_{i \in [s]} \Pr[i \in T] = s \cdot \frac{k}{2s} = \frac{k}{2}$$

and

$$\sum_{j=0}^{k} p(t,k,j) \cdot j^2 = \mathbb{E}_T\left[\sum_{i,j \in [s]} \mathbf{1}[i \in T] \cdot \mathbf{1}[j \in T]\right] = \sum_{i,j \in [s]} \Pr[i,j \in T]$$
$$= s \cdot \frac{k}{2s} + s(s-1)\frac{k(k-1)}{2s(2s-1)} \leq \frac{k}{2} + \frac{k(k-1)}{4} = \frac{k(k+1)}{4}.$$

□

Proof of Lemma 3.9. Let $f(t,k) = t \cdot \mathrm{val}(\mathcal{F}[t,k])$. We have $f(t,0) = 0, f(t,1) = 1$ and $f(t,k) = 0$ if $k > t$. The definition of $f(t,k)$ for $k \geq 2$ is recursive. Let $p(t,k,j) = \frac{\binom{t/2}{j}\binom{t/2}{k-j}}{\binom{t}{k}}$. Then

$$\mathrm{val}(\mathcal{F}[t,k]) = \sum_{j=0}^{k} p(t,k,j) \left(\mathrm{val}(\mathcal{F}[t/2,j]) + \mathrm{val}(\mathcal{F}[t/2,k-j])\right).$$

which implies

$$f(t,k) = 4\sum_{j=0}^{k} p(t,k,j) f(t/2,j).$$

It will be instructive to compute $f(t,2)$:

$$f(t,2) = \frac{t}{t-1} + \frac{t-2}{t-1} f(t/2,2) \leq 2 + f(t/2,2) \leq 2\log(t).$$

We will prove by induction that

$$f(t,k) \leq k^2 + k(k-1)\log(t).$$

We already verified this for $k = 0,1,2$. For $k \geq 3$ we have by induction:

$$f(t,k) \leq 4\sum_{j=0}^{k} p(t,k,j)\left(j^2 + j(j-1)\log(t/2)\right).$$

Applying Claim 3.10 gives

$$\begin{aligned} f(t,k) &\leq k(k+1) + k(k-1)\log(t/2) \\ &= 2k + k(k-1)\log(t) \\ &\leq k^2 + k(k-1)\log(t). \end{aligned}$$

Finally we get

$$\mathrm{val}(\mathcal{F}[t,k]) = \frac{f(t,k)}{t} \leq \frac{k^2 + k(k-1)\log(t)}{t} \leq \frac{k^2\log(2t)}{t}.$$

□

Our application for streaming algorithms for the needle problem has an additional restriction, that the total length of the intervals in the interval system be bounded away from t. We refer to such interval systems as *valid*.

Definition 3.11 (Valid interval systems)**.** *A $[t]$-interval system F is called* valid *if $\sum_{I \in F} |I| \leq t/2$. A randomized $[t]$-interval system $\mathcal{F}$ is called valid if all $[t]$-interval systems F in its support are valid.*

We next show how to refine a an interval system to obtain a valid randomized interval system, while preserving the sets distribution, and without increasing the value too much.

Lemma 3.12. *Assume $k \le t/6$. Let F be a $[t,k]$-interval system. Then there exists a randomized $[t,k]$-interval system $\mathcal{F}$ such that:*

1. *Sets$(\mathcal{F}) =$ Sets(F)*
2. *val$(\mathcal{F}) \le 5 \cdot$ val(F)*
3. *$\mathcal{F}$ is valid*

Proof. Let $F = \{I_1, \dots, I_k\}$. Given an interval I_i define $\ell_i = \min(3, |I_i|)$. Partition I_i into ℓ_i intervals $\{I_{i,a} : a \in [\ell_i]\}$ of as equal length as possible, and observe that

$$\frac{|I_i|}{5} \le |I_{i,a}| \le \frac{|I_i|}{3} + 1 \quad \forall a \in [\ell_i].$$

Let $p_{i,a} = \frac{|I_{i,a}|}{|I_i|}$. We define a randomized $[t,k]$-interval system $\mathcal{F}$, where for each $i \in [k]$ independently, we replace I_i with one of its sub-intervals. Concretely, we choose $a \in [\ell_i]$ with probability $p_{i,a}$ and replace I_i with $I_{i,a}$. We now prove the claims.

1. Observe that sampling a uniform element $x \in I_i$ can equivalently be sampled by first sampling $a \in [\ell_i]$ with probability $p_{i,a}$, and then sampling a uniform element $x \in I_{i,a}$. This implies that Sets$(\mathcal{F}) =$ Sets(F).
2. Since $|I_{i,a}| \ge |I_i|/5$ for all i, a, the claim holds for any F' in the support of $\mathcal{F}$, and hence also for $\mathcal{F}$.
3. Since $|I_{i,a}| \le (|I_i| + 1)/2$ for all i, a, we have for any $F' = \{I_{1,a_1}, \dots, I_{k,a_k}\}$ in the support of $\mathcal{F}'$ that
$$\sum_{i \in [k]} |I_{i,a_i}| \le k + \frac{1}{3} \sum_{i \in [k]} |I_i| \le k + \frac{t}{3} \le \frac{t}{2}$$
where the last inequality follows since we assume $k \le t/6$.

□

Lemma 3.12 applies also to randomized $[t,k]$-interval systems, by applying it to any interval system in their support. The following lemma summarizes all the facts we would need by applying it to $\mathcal{F}[t,k]$.

Lemma 3.13. *Let $k, t \ge 1$. Assume t is a power of two and $k \le t/6$. Then there exists a valid perfect randomized $[t,k]$-interval system $\mathcal{F}$ with*

$$\mathit{val}(\mathcal{F}) \le \frac{10k^2 \log(t)}{t}.$$

4 Lower bound for the needle problem

We prove Theorem 1.2 in this section, by combining our lower bound for unique set-disjointness with fixed set sizes (Theorem 2.13) with the efficient reduction given by interval systems (Lemma 3.13).

First, we recall the parameters: n denotes the size of the domain, t the number of samples and p the needle probability. We assume throughout that $n = \Omega(t^2)$ is large enough. We would denote by k the number of needles in a stream in the planted model, where $k \sim \text{Bin}(t, p)$. We denote by Uniform the uniform distribution over $[n]^t$.

First, we show how to prove lower bounds when k is fixed. Given a $[t, k]$-interval system $F = \{I_1, \ldots, I_k\}$, we will assume in this section that the intervals are sorted in order, namely that I_1 comes before I_2, which comes before I_3, and so on. We define its corresponding sizes as

$$\text{Sizes}(F) = (|I_1|, \ldots, |I_k|).$$

We recall the definition of a planted stream distribution from the introduction, where we now present it more formally.

Definition 4.1 (Planted distribution for interval systems)**.** *Let F be a $[t]$-interval system. we define a planted distribution Planted$[F]$ over streams $X \in [n]^t$ as follows:*

1. *Sample uniform needle $x \in [n]$;*
2. *In each interval $I \in F$ sample uniform index $a_I \in I$ and set $X_{a_I} = x$;*
3. *For all $j \in [n] \setminus \{a_I : I \in F\}$, sample $X_j \in [n]$ uniformly.*

For $\mathcal{F}$ a randomized $[t]$-interval system, we define its planted distribution Planted$[\mathcal{F}]$ by first sampling $F \sim \mathcal{F}$ and then $X \sim$ Planted$[F]$.

We start by formalizing and proving Lemma 1.7. Given a streaming algorithm $\mathcal{ALG}$ and two distributions D_0, D_1 over streams, we say that $\mathcal{ALG}$ distinguishes between D_0, D_1 with error δ if, at the end of running the algorithm, the last player can guess if the input was sampled from D_0 or D_1 and be correct with probability at least $1 - \delta$. A streaming algorithm is an ℓ-pass streaming algorithm if it makes ℓ passes over the data stream.

Lemma 4.2. *Let F be a $[t, k]$-interval system and set $\mathfrak{s} =$ Sizes(F). Let $\mathcal{ALG}$ be an ℓ-pass streaming algorithm which distinguishes between Planted$[F]$ and Uniform with error 0.5% and uses space s. Then there is a communication protocol Π which solves the unique set-disjointness problem under input distribution $\mu_{\text{size}}[\mathfrak{s}]$, in which each player sends ℓs bits, and has error 1%.*

Proof. Let $X = (X_1, \ldots, X_k) \in (\{0, 1\}^n)^k$ be the input to the players, where we assume $X \sim \mu^b_{\text{size}}[\mathfrak{s}]$ for some $b \in \{0, 1\}$. The goal of the players is to figure out b.

Let $F = \{I_1, \ldots, I_k\}$. Let $J_1, \ldots, J_k$ be a partition of $[t]$, where $I_i \subset J_i$. As a first step, each player individually constructs a stream $Y_i \in [n]^{J_i}$ based on their input X_i. The i-th player generates their stream as follows:

1. For each $j \in J_i \setminus I_i$, sample $Y_i(j) \in [n]$ uniformly.
2. Let $S_i = \{j \in [n] : X_i(j) = 1\}$, where $|S_i| = s_i$ be assumption. Let $L_i \in [n]^{s_i}$ be a random permutation of S_i. Set $(Y_i(j) : j \in I_i) = L_i$.

Let $Y = Y_1 \circ \cdots \circ Y_k \in [n]^t$ be the concatenation of the streams. The players simulate running $\mathcal{ALG}$ on the stream, where each player simulates it on their part of the stream, and send the internal memory of the streaming algorithm to the next player. At the end of each pass, the last player sends the internal memory back to the first player. Thus each player sends at most ℓs bits. To conclude, we need to show that this allows to distinguish between $b = 0$ and $b = 1$.

To conclude, we compute the distribution of Y based on the value of b, and show that when $b = 0$ the distribution of Y is close to uniform, and when $b = 1$ it is close to the planted distribution Planted$[F]$. Thus by assumption the algorithm distinguishes between these two cases, which is our goal.

First, if $b = 0$ then $X_1, \ldots, X_k$ are uniform sets of sizes $s_1, \ldots, s_k$ in $[n]$, conditioned on being pairwise disjoint. Thus the elements of Y are uniform among all choices of t distinct elements in n. Since we assume $n = \Omega(t^2)$, the statistical distance between Y and Uniform is at most t^2/n, which can be made as small as we want, say 0.1%.

Similarly, if $b = 1$ then $X_1, \ldots, X_k$ are uniform conditioned on having a unique intersection. Similarly, the assumption $n = \Omega(t^2)$ implies that the the statistical distance between Y and Planted$[F]$ can be made as small as we want, say 0.1%.

Overall, as we assume that $\mathcal{ALG}$ can distinguish between Uniform and Planted$[F]$ with error 0.5%, then it also distinguishes between the distributions of Y for $b = 0$ and $b = 1$ with slightly larger error 1%. □

Combining Lemma 4.2 with Theorem 2.13, we obtain the following corollary which formalizes Lemma 1.7.

Lemma 4.3. *Let F be a valid $[t, k]$-interval system. Let $\mathcal{ALG}$ be an ℓ-pass streaming algorithm which distinguishes between Planted$[F]$ and Uniform with error 0.5% and uses space s. Then*

$$\ell s = \Omega\left(\frac{1}{val(F)}\right).$$

Proof. Let Π be the protocol obtained by Lemma 4.2, which solves unique set-disjointness under inputs distribution $\mu_{\text{size}}[\mathfrak{s}]$ for $\mathfrak{s} = \text{Sizes}(F) = [s_1, \ldots, s_k]$, and where each player sends at most ℓs bits. Since F is valid we have $\sum s_i \leq t/2$. Theorem 2.13 then gives

$$\sum_{i \in [k]} \frac{\ell s}{s_i} = \Omega(1).$$

Recalling the definition of $\text{val}(F) = \sum_{i \in [k]} \frac{1}{s_i}$, we can rephrase this as $\ell s \cdot \text{val}(F) = \Omega(1)$. □

The following lemma, which formalizes Lemma 1.8, generalizes Lemma 4.3 to randomized interval systems.

Lemma 4.4. *Let $\mathcal{F}$ be a valid randomized $[t]$-interval system. Let $\mathcal{ALG}$ be an ℓ-pass streaming algorithm which distinguishes between Planted$[\mathcal{F}]$ and Uniform with error 0.1% and uses space s. Then*

$$\ell s = \Omega\left(\frac{1}{\mathit{val}(\mathcal{F})}\right).$$

Proof. Sample $F \sim \mathcal{F}$. Since $\mathrm{val}(\mathcal{F}) = \mathbb{E}[\mathrm{val}(F)]$, by Markov's inequality we have

$$\Pr_F[\mathrm{val}(F) > 2\mathrm{val}(\mathcal{F})] \leq 50\%.$$

Next, let $\mathrm{err}(F)$ denote the error of $\mathcal{ALG}$ in distinguishing Planted$[F]$ from Uniform. Since Planted$[\mathcal{F}]$ is a mixture of Planted$[F]$, then the average of $\mathrm{err}(F)$ is the error of $\mathcal{ALG}$ in distinguishing Planted$[\mathcal{F}]$ from Uniform, which we assume is 0.1%. Thus

$$\Pr_F[\mathrm{err}(F) > 0.5\%] \leq 20\%.$$

Overall, there is some choice of F in the support of $\mathcal{F}$ such that $\mathrm{val}(F) \leq 2\mathrm{val}(\mathcal{F})$ and $\mathrm{err}(F) \leq 0.5\%$. The lemma follows by applying Lemma 4.3 to F. □

We now in place to finally prove Theorem 1.2, giving sample-space lower bounds for any streaming algorithm that solves the needle problem.

Proof of Theorem 1.2. Let $\mathcal{ALG}$ be an ℓ-pass streaming algorithm which can distinguish with high probability between the uniform and planted needle distribution using t samples. As the inputs are stochastic, we may repeat it a few times to decrease its error. Thus, by increasing t by a constant multiplicative factor, we may assume that the error is at most 0.1% and that t is a power of two.

For $k \leq t$ let $\mathcal{F}_k$ be the valid perfect randomized $[t, k]$-interval system given by Lemma 3.13. We construct a randomized $[t]$-interval system $\mathcal{F}$ by sampling $k \sim \mathrm{Bin}(t, p)$ and taking $\mathcal{F}_k$. Observe that Planted$[\mathcal{F}]$ is identical to the planted needle distribution. If $\mathcal{ALG}$ uses s bits of space then Lemma 4.4 gives that

$$\ell s = \Omega\left(\frac{1}{\mathrm{val}(\mathcal{F})}\right).$$

To conclude the proof we just need to compute $\mathrm{val}(\mathcal{F})$. For any fixed k we have by Lemma 3.13 that

$$\mathrm{val}(\mathcal{F}_k) \leq \frac{10k^2 \log(t)}{t}.$$

Since $k \sim \mathrm{Bin}(t, p)$ we have $\mathbb{E}[k^2] = p(1-p)t + p^2t^2$. Since we assume $p = \Omega(1/t)$, the dominant term is the quadratic term, and hence $\mathbb{E}[k^2] = \Theta(p^2t^2)$. Thus we get

$$\mathrm{val}(\mathcal{F}) = O(p^2 t \log(t)).$$

Rearranging the terms concludes the proof, since it gives $\ell p^2 s t \log(t) = \Omega(1)$. □

5 Open problems

We proved in Theorem 1.2 near-tight bound for the sample vs space complexity needed for the needle problem, which proves similar near-tight bounds for the frequency estimation in stochastic streams problem. It still remains open to prove sharp bounds, removing the remaining logarithmic factor. We propose the following natural conjecture.

Conjecture 5.1. *Any ℓ-pass streaming algorithm which can distinguish with high probability between the uniform and planted models, where p is the needle probability, t the number of samples, s the space and n the domain size, satisfies $\ell p^2 st = \Omega(1)$.*

Another natural conjecture is to remove the artificial restriction of $\sum s_i \leq n/2$ from Theorem 1.5. We need it because we do not prove the theorem directly, but rather via a reduction to the asymmetric product distribution case. We speculate that there may be a direct proof which overcomes this technical barrier (although we don't really have any application where the general bound is needed, it will be aesthetically pleasing to have a more complete result).

References

[AMOP08] Alexandr Andoni, Andrew McGregor, Krzysztof Onak, and Rina Panigrahy. Better bounds for frequency moments in random-order streams. *arXiv preprint arXiv:0808.2222*, 2008. 2, 3, 4, 6

[AMS99] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *Journal of Computer and system sciences*, 58(1):137–147, 1999. 2, 3, 5

[BIPW10] Khanh Do Ba, Piotr Indyk, Eric Price, and David P Woodruff. Lower bounds for sparse recovery. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1190–1197. SIAM, 2010. 7

[BR00] Omer Barkol and Yuval Rabani. Tighter bounds for nearest neighbor search and related problems in the cell probe model. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 388–396, 2000. 7

[BVWY18] Vladimir Braverman, Emanuele Viola, David P Woodruff, and Lin F Yang. Revisiting frequency moment estimation in random order streams. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018. 2, 4

[BYJKS04] Ziv Bar-Yossef, Thathachar S Jayram, Ravi Kumar, and D Sivakumar. An information statistics approach to data stream and communication complexity. *Journal of Computer and System Sciences*, 68(4):702–732, 2004. 4, 5, 13

[CCM08] Amit Chakrabarti, Graham Cormode, and Andrew McGregor. Robust lower bounds for communication and stream computation. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 641–650, 2008. 3

[CJP08] Amit Chakrabarti, TS Jayram, and Mihai Pătraşcu. Tight lower bounds for selection in randomly ordered streams. In *Proceedings of the nineteenth annual ACM-SIAM Symposium on Discrete Algorithms*, pages 720–729, 2008. 3

[CKLM18] Arkadev Chattopadhyay, Michal Kouckỳ, Bruno Loff, and Sagnik Mukhopadhyay. Simulation beats richness: New data-structure lower bounds. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1013–1020, 2018. 7

[CKS03] Amit Chakrabarti, Subhash Khot, and Xiaodong Sun. Near-optimal lower bounds on the multi-party communication complexity of set disjointness. In *18th IEEE Annual Conference on Computational Complexity, 2003. Proceedings.*, pages 107–117. IEEE, 2003. 3, 4, 5, 13

[CMVW16] Michael Crouch, Andrew McGregor, Gregory Valiant, and David P Woodruff. Stochastic streams: Sample complexity vs. space complexity. In *24th Annual European Symposium on Algorithms (ESA 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016. 2, 3

[DDKS16] Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Memory-efficient algorithms for finding needles in haystacks. In *Annual International Cryptology Conference*, pages 185–206. Springer, 2016. 7

[DLOM02] Erik D Demaine, Alejandro López-Ortiz, and J Ian Munro. Frequency estimation of internet packet streams with limited space. In *European Symposium on Algorithms*, pages 348–360. Springer, 2002. 3

[GH09] Sudipto Guha and Zhiyi Huang. Revisiting the direct sum theorem and space lower bounds in random order streams. In *International Colloquium on Automata, Languages, and Programming*, pages 513–524. Springer, 2009. 2, 3, 4

[GM07] Sudipto Guha and Andrew McGregor. Space-efficient sampling. In *Artificial Intelligence and Statistics*, pages 171–178. PMLR, 2007. 3

[GM09] Sudipto Guha and Andrew McGregor. Stream order and order statistics: Quantile estimation in random-order streams. *SIAM Journal on Computing*, 38(5):2044–2059, 2009. 3

[Gro09] Andre Gronemeier. Asymptotically optimal lower bounds on the NIH-multiparty information complexity of the AND-function and disjointness. In *26th International Symposium on Theoretical Aspects of Computer Science*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2009. 5, 9, 13

[Gro10] André Gronemeier. Information complexity and data stream algorithms for basic problems. 2010. https://eldorado.tu-dortmund.de/bitstream/2003/27529/1/Gronemeier2010.pdf. 13

[HW07] Johan Håstad and Avi Wigderson. The randomized communication complexity of set disjointness. *Theory of Computing*, 3(1):211–219, 2007. 5

[IW05] Piotr Indyk and David Woodruff. Optimal approximations of the frequency moments of data streams. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 202–208, 2005. 3

[Jay09] T. S. Jayram. Hellinger strikes back: A note on the multi-party information complexity of AND. APPROX '09 / RANDOM '09, page 562–573, Berlin, Heidelberg, 2009. Springer-Verlag. 5, 9, 13

[JKKR03] TS Jayram, Subhash Khot, Ravi Kumar, and Yuval Rabani. Cell-probe lower bounds for the partial match problem. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, pages 667–672, 2003. 7

[KPW21] Akshay Kamath, Eric Price, and David P Woodruff. A simple proof of a new set disjointness with applications to data streams. *arXiv preprint arXiv:2105.11338*, 2021. 4

[MNSW95] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. In *Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, pages 103–111, 1995. 7

[MP80] J Ian Munro and Mike S Paterson. Selection and sorting with limited storage. *Theoretical computer science*, 12(3):315–323, 1980. 3

[MPTW12] Andrew McGregor, A Pavan, Srikanta Tirthapura, and David Woodruff. Space-efficient estimation of statistics over sub-sampled streams. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGAI symposium on Principles of Database Systems*, pages 273–282, 2012. 3

[MPTW16] Andrew McGregor, A Pavan, Srikanta Tirthapura, and David P Woodruff. Space-efficient estimation of statistics over sub-sampled streams. *Algorithmica*, 2(74):787–811, 2016. 3

[PT06] Mihai Pătraşcu and Mikkel Thorup. Time-space trade-offs for predecessor search. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 232–240, 2006. 7

[YZ22] Guangxu Yang and Jiapeng Zhang. Lifting theorems meet information complexity: Known and new lower bounds of set-disjointness. *Manuscript*, 2022. 5