Lower Bounds for Randomized Read/Write Stream Algorithms

Paul Beame[*]
Computer Science and Engineering
University of Washington
Seattle, WA 98195-2350
beame@cs.washington.edu

T. S. Jayram[†]
IBM Almaden Research Center
650 Harry Road
San Jose, CA 95120
jayram@almaden.ibm.com

Atri Rudra[‡]
Computer Science and Engineering
University of Washington
Seattle, WA 98195-2350
atri@cs.washington.edu

ABSTRACT

Motivated by the capabilities of modern storage architectures, we consider the following generalization of the data stream model where the algorithm has sequential access to multiple streams. Unlike the data stream model, where the stream is *read only*, in this new model (introduced in [8, 9]) the algorithms can also *write* onto streams. There is no limit on the size of the streams but the number of passes made on the streams is restricted. On the other hand, the amount of internal memory used by the algorithm is scarce, similar to data stream model.

We resolve the main open problem in [7] of proving lower bounds in this model for algorithms that are allowed to have *2-sided error.* Previously, such lower bounds were shown only for deterministic and 1-sided error randomized algorithms [9, 7]. We consider the classical *set disjointness* problem that has proved to be invaluable for deriving lower bounds for many other problems involving data streams and other randomized models of computation. For this problem, we show a near-linear lower bound on the size of the internal memory used by a randomized algorithm with 2-sided error that is allowed to have $o(\log N/\log\log N)$ passes over the streams. This bound is almost optimal since there is a simple algorithm that can solve this problem using logarithmic memory if the number of passes over the streams is allowed to be $O(\log N)$.

Applications include near-linear lower bounds on the internal memory for well-known problems in the literature: (1) approximately counting the number of distinct elements in the input (F_0); (2) approximating the frequency of the *mode* of an input sequence (F^*_∞); (3) computing the join of two relations; and (4) deciding if some node of an XML document matches an XQuery (or XPath) query.

Our techniques involve a novel direct-sum type of argument that yields lower bounds for many other problems. Our results asymptotically improve previously known bounds for any problem even in deterministic and 1-sided error models of computation.

Categories and Subject Descriptors

F.2.3 [**Analysis of Algorithms and Problem Complexity**]: Tradeoffs between Complexity Measures

General Terms

Algorithms Theory

Keywords

data stream algorithms, communication complexity

1. INTRODUCTION

The use of massive data in many application areas has led to an interest in computational models that adequately capture the notion of efficient computation over such data sets. Perhaps the most important of these is the *data stream model* in which access to the input is restricted to be *sequential.* The literature on data stream algorithms for massive data sets is vast. We refer the reader to two overview papers: one by Muthukrishnan [11] which contains many applications and algorithmic techniques, and another by Babcock et al. [3] for a database perspective on the applications.

Despite these successes, the data stream model is still a very limited model of computation. This model is most relevant when the data is truly streaming, i.e. there is no access to a read/write storage media. However, in many applications such as the web, the data resides on disks that can be accessed in a read-write manner using modern hardware architectures. Classically, memory is arranged in a hierarchical fashion where the most significant gap in performance is between the main memory and the next lower level in the hierarchy that uses magnetic disks. Modern storage architectures use sophisticated prefetching and caching techniques to reduce the effective seek time on disks. Consequently, the I/O rates for *sequential* access to data on disks are as good as, if not better than, the I/O rates for random access to internal memory (cf. [13, Chapter 6]).

A computational model that incorporates these features was proposed recently by Grohe and Schweikardt [9]. Informally, it has both internal memory and external memory in

[*] Research supported by NSF grant CCF-0514870.

[†] Part of this research was done while the author was visiting University of Washington

[‡] Research supported in part by NSF grant CCF-0343672. Part of this research was done while the author was visiting IBM Almaden.

STOC'07, June 11–13, 2007, San Diego, California, USA.
Copyright 2007 ACM 978-1-59593-631-8/07/0006 ...$5.00.

the form of a constant number of read/write streams, and it has the property that random access to internal memory and sequential access to the streams are equally efficient. The resources of interest are the required amount of internal memory, and the number of scans of the external streams. The model in [9] is essentially a variant of the traditional multi-tape Turing machine. The machine has t tapes, representing the *read-write external memory*, and one of these tapes contains the input. The contents of these tapes can be read/written only in a sequential manner. (Each tape has its own head that can move left/right independent of the heads on other tapes.) The machine also has access to some internal memory that can be accessed arbitrarily[1]. As with standard data streams, the *space*, the required amount of internal memory, is a key resource for this model. To deal with sequential access to external memory, Grohe and Schweikardt considered the number of *reversals* made by the machine. This measure counts the total number of times that the heads on the tapes reverse direction during the course of computation[2]. We call this the *read/write stream* computational model.

The three resource parameters that are important to an read/write stream algorithm A are: (1) the number of external read/write tapes t that A uses, (2) the maximum space s used by A, and (3) the maximum number of reversals r made by A on all the external tapes. In this case we call A an (r, s, t)-read/write stream algorithm. In general r and s will depend on the input size, though we typically consider the number of tapes to be constant.

The read/write stream model generalizes the data stream model, which corresponds to the case that $t = 1$ and the r passes over the input are *read-only*. As pointed out in [9], the read/write stream model is strictly more powerful than the ordinary data stream model: the read/write stream model can sort lists of size N with $O(\log N)$ reversals and constant space using 3 tapes, but by standard hardness results on the communication complexity of set disjointness [12], even deciding the simpler element distinctness problem in a randomized data stream model satisfies $r \cdot s = \Omega(N)$.

Compared to the data stream model, proving lower bounds in the read/write stream model is much harder. Lower bounds for data stream and related models for massive data sets are proved essentially via communication complexity. For a data stream algorithm that uses s space and r passes one can, for example, divide the input into two parts and associate each part with a player. In each pass of the data stream algorithm, the number of bits communicated between the players is at most s for a total communication of $r \cdot s$ bits which must be at least the communication complexity of the problem in the two player game. However, in the more powerful read/write stream model, this argument fails. As Grohe and Schweikardt [9] point out, the ability to copy the contents of one external tape to another allows the parties to share information without communicating via the internal memory.

One might hope that randomized time-space tradeoff lower bounds (e.g. [4]) could be used to obtain tradeoffs between reversals and space in the read/write stream model by bounding the time used as a function of the number of reversals. However, those lower bounds only apply with read-only inputs and sub-linear space bounds; with the ability of read/write stream algorithms to write to linear (or even significantly super-linear) numbers of memory locations on the external tapes, neither the time nor the space bounds are small enough for the time-space tradeoffs to apply.

Grohe and Schweikardt [9] circumvent these issues by proving lower bounds directly for the read/write stream model using combinatorial arguments. They showed that the $O(\log N)$ upper bound on reversals for sorting is necessary: any read/write stream algorithm for sorting that uses $o(\log N)$ reversals requires $\widetilde{\Omega}(N^{1/5})$ space. A followup work by Grohe, Hernich and Schweikardt [7] extends these techniques to decision problems; one of their main results shows that checking whether two sets are equal requires space $\widetilde{\Omega}(N^{1/4})$ when the number of reversals is $o(\log N)$. In fact, they show that the lower bound also holds for one-sided error randomized read/write stream algorithms.

However, much of our current understanding in the data stream model involves proving lower bounds for *decision* problems in which the computation can have *two*-sided error. This is because in problems where a numerical output (e.g. number of distinct elements in the input) needs to be approximated, the lower bounds are proven using suitable reductions from "hard" decision problems. The approximation guarantees on the original problem then translate to allowing 2-sided error for the corresponding decision problem. To obtain a comparable understanding, we need similar lower bounds to be proven in the read/write stream model as well.

Finding such a lower bound was posed as the main open problem in [7]. It is easy to see that the main decision problem shown to be hard in [7], set equality, has a simple randomized 1-pass read-only data stream algorithm based on random fingerprints of $O(\log N)$ bits and thus is trivial for randomized $(1, O(\log N), 1)$-read/write stream algorithms. Grohe et al. [7] also ask if set disjointness is hard for read/write stream algorithms. Set disjointness is a particularly important problem because, via suitable reductions, lower bounds for set disjointness have proved invaluable for deriving lower bounds for many other problems involving data streams and other randomized models of computation.

Our Results.

We resolve the open problems posed by Grohe et al. [7]. We develop the first lower bounds for randomized read/write stream algorithms with two-sided error that apply to a variety of decision problems, including set disjointness. We do this by developing general techniques that yield lower bounds for 2-sided error randomized read/write stream algorithms computing direct sums of functions with small discrepancy or large "corruption", combined using either $\oplus$ or $\vee$ connectives. (These are the first such generic lower bound results even for the deterministic read/write stream model.) In addition to generalizing the class of algorithms to which our bounds apply, we also obtain bounds that are quantitatively better than those in [9, 7].

[1] In [9] the definition assumed that internal memory was also represented using tapes but their results apply equally to the model defined here, which motivates us to use somewhat different terminology from [9].

[2] Observe that this model allows for the input to be accessed in reverse; moreover, it allows the heads to reverse direction in the middle of a tape. In modern disks, these incur significant performance costs and the assumption of efficient sequential access is violated. However, for the purposes of proving lower bounds, which was the goal of their paper and ours as well, this only makes the results stronger.

For example, we show that any randomized read/write stream algorithm with two-sided error for the INTERSECTION-MOD-2 problem (given two sets A, B, is $|A \cap B|$ even?) using $o(\log N)$ reversals requires space $\Omega(N^{1-\delta})$, for any $\delta > 0$ where N is the inputs size.

The main result of this paper, and also technically the most involved, concerns the SET DISJOINTNESS problem. We show that any randomized read/write stream algorithm with two sided error for SET DISJOINTNESS using $o(\log N/\log\log N)$ reversals requires space $\Omega(N^{1-\delta})$, for any $\delta > 0$, where N is the input size.

Our lower bound for SET DISJOINTNESS implies similar lower bounds for other problems in the read/write stream model, given suitable sub-logarithmic bounds on the number of reversals. This includes a lower bound of essentially $\Omega(1/\epsilon^{1-\delta})$ for any $\delta > 0$ on the space required to compute a $(1+\epsilon)$-approximation of the frequency moment F_0, the number of distinct inputs in the stream, on any randomized read/write stream algorithm with two sided error. (Until [10, 15], which proved a tight bound of $\Omega(1/\epsilon^2)$ space, this was nearly the best result known even in the weaker data stream model.) We also derive a lower bound of $\Omega(N^{1-\delta})$ on the space for computing a 2-approximation of F^*_∞ which denotes the largest frequency of any element in the stream. Also, following [8], we obtain a lower bound of $\Omega(N^{1-\delta})$ on the size of the internal memory for randomized read/write stream algorithms for the following problems: (1) Given two input relations, decide if their join is empty. (2) Given an XQuery (or XPath) query and an XML document, decide if the query filters the document, that is, check if the result of evaluating the query on the document is non-empty.

Techniques.

Note that the set disjointness problem in the read/write stream model is trivial if the elements of the set are always presented in increasing order. On the other hand if the common elements of the two sets are located arbitrarily within each set, then a significant price needs to be paid in terms of the number of reversals. Grohe and Schweikardt [9] show how to exploit this observation in a clever manner by considering an appropriate fixed permutation of the locations of elements of the second set with respect to the first set. The key result of their paper is that for any deterministic read/write stream algorithm, there exists a partition of the inputs into a small number of *skeletons* such that the inputs that map to the same skeleton satisfy a "rectangle" property. This is similar in spirit to the rectangle property of transcripts of a deterministic protocol in communication complexity. However, the proofs in [9, 7] only use the fact that there exists a rectangle that contains at least four elements. Our proofs require larger rectangles. Further, the rectangles need to satisfy some extra properties.

Our methodology for proving lower bounds is a general template that converts any problem P that is essentially hard for randomized communication protocols to a new direct-sum type problem consisting of many copies of P that is hard in the read/write stream model. We consider two types of problems that are known to be hard for randomized communication complexity (a) functions that have low discrepancy, e.g. inner product [6, 2], and (b) functions that have high corruption, e.g. set disjointness [12]. We consider two kinds of direct-sum problems. First, we consider the case of taking the parity of many instances of P in which each instance of P either has a low discrepancy or high corruption for sufficiently large rectangles. In this case our proof by contradiction uses a careful counting argument to reduce the parity of many instances of P to a single instance where the violation of discrepancy or corruption occurs. Next, we consider the case where the direct sum involves taking the OR of many instances of P. In this case notice that since we are taking independent copies, our hard distribution for OR of many copies of P should have negligible mass on the 1-inputs of each individual instance of P. Thus the previous argument of reducing to a single instance fails completely. On the other hand, the results of [9, 7] guarantees a set of coordinates where the rectangular property applies to each instance, but not if the instances are all considered together. We overcome this technical obstacle via a more subtle argument that assigns inputs in a weighted manner to suitable chosen combinatorial rectangles, and then appealing to the corruption bound for each rectangle.

Other related work.

There are a few other models that have some similarity to the read/write stream model. In "external memory algorithms" [14], the data is partitioned into blocks that can be accessed in a non-sequential manner by incurring a significant cost. Once a block is retrieved into main memory, the items within the block can be accessed arbitrarily (random-access). This model does not however address *sequential access* to external memory, which is an important aspect of the read/write stream model. The StrSort model in [1] allows streams to be sorted on-the-fly; simulating this in our model will incur a $\Omega(\log N)$ blow-up in the number of reversals. Moreover, their model allows a logarithmic number of streams. Another related model is the *bounded reversal Turing machine* [5] which is a weaker read/write stream model with no internal memory. There, the known results apply only when r is $\omega(N)$ whereas the focus of this paper is when the number of reversals is at most logarithmic.

2. PRELIMINARIES

Let X denote a base domain of cardinality n. All the problems that we will consider in this paper will have $2m$ values from the base domain as its input. For notational brevity, let $[\ell]$ denote the set of integers $\{1, \ldots, \ell\}$. A crucial notion that we will need is the *sortedness* of a permutation ϕ on m elements, denoted by $\mathsf{sortedness}(\phi)$, which is defined to be the length of the largest monotone (increasing or decreasing) subsequence in ϕ.

Definition 1. Let $r, s : \mathbb{N} \to \mathbb{N}$ and $t \geq 1$ be an integer. An (r, s, t)-read/write stream algorithm A is a family $\{A_N\}_{N\in\mathbb{N}}$ of t-tape automata over a fixed tape alphabet Σ, with the property that A_N has at most $2^{s(N)}$ states, each tape of A_N is unbounded with one read/write head that moves independently of the heads on other tapes and, on every N-bit input, A_N halts after having made at most $r(N)$ reversals.

Our definition of read/write stream algorithms is a non-uniform extension of the definition of Grohe and Schweikardt. The proofs in [7, 9] go via an intermediate machine model called *list machines* and uses the concept of the *skeleton* of an input to a list machine. We will not define these concepts explicitly but will abstract out the main results that are needed for our work. In fact, we will not talk about list machines at all. Further, we will only use the property of skeletons that they partition the input space into disjoint sets such that the inputs within each skeleton satisfy certain special properties. The following notions will be used to describe these properties.

Definition 2. For $\mathbf{v} \in X^{2m}$ and $I \subseteq [2m]$, define $\mathbf{v}_I \in X^I$ to be the *projection* of $\mathbf{v}$ on the coordinates I. For $S \subseteq X^{2m}$ define $S_I = \{\mathbf{v}_I \mid \mathbf{v} \in S\}$.

Definition 3. A set $R \subseteq X \times X$ is called a *rectangle* if and only if $R = A \times B$ for some $A, B \subseteq X$.

The fundamental theorem of communication complexity states that in a deterministic communication protocol the inputs that are produce any fixed transcript form a rectangle. Such a strong property does not hold for the inputs that produce a fixed skeleton of a read/write stream algorithm. A weaker property was shown in [9, 7] for read/write stream algorithms based on the following notion.

Definition 4. A set of inputs $S \subseteq X^{2m}$ is an (i,j)-*rectangle* for some $i \neq j \in [2m]$ if and only if:
(a) For all $\mathbf{u}, \mathbf{v} \in S$, the projections $\mathbf{u}_{[2m]\setminus\{i,j\}} = \mathbf{v}_{[2m]\setminus\{i,j\}}$; i.e., the inputs in S agree on all coordinates outside of i and j.
(b) The projection $S_{\{i,j\}}$ is a rectangle on $X \times X$.

The following captures the results from [9, 7] that we will use. A sketch of its proof is given in the appendix.

Proposition 1 ([7, 9]). *Let A be an (r,s,t)-deterministic read/write stream algorithm on $2m$ inputs, each of which belongs to a base domain X with $n = \lceil \log_2 |X| \rceil$. Then A induces a function σ on X^{2m} (mapping each input to its associated skeleton) such that:*

(a) $\log_2 |\sigma(X^{2m})| \leq dm^2(r \cdot s + \log_2(mn))$, for some constant d depending only on t;

(b) For all $\mathbf{u}, \mathbf{v} \in X^{2m}$, if $\sigma(\mathbf{u}) = \sigma(\mathbf{v})$ then $\mathsf{A}(\mathbf{u}) = \mathsf{A}(\mathbf{v})$;

(c) For any skeleton $\xi \in \sigma(X^{2m})$ and permutation ϕ on $[m]$, there exists a set $I \subset [m]$ with

$$|I| = m - t^{2r}\mathsf{sortedness}(\phi)$$

satisfying the following property:

Let $J = [2m] \setminus \{i, m + \phi(i)\}$. Then, for every $i \in I$ and every $\mathbf{a} \in X^J$ the set

$$\{\mathbf{v} \in X^{2m} \mid \sigma(\mathbf{v}) = \xi \text{ and } \mathbf{v}_J = \mathbf{a}\}$$

is an $(i, m + \phi(i))$-rectangle.

Following Grohe et. al. we will apply this lemma using a permutation ϕ^* on $[m]$, which satisfies $\mathsf{sortedness}(\phi^*) \leq 2\sqrt{m}$. One such permutation sorts the indices into $\sqrt{m}$ blocks of $\sqrt{m}$ inputs that are increasing within a block but decreasing between blocks.

2.1 Hardness Measures

The lower bounds for the read/write stream model rely on 2 measures of hardness of Boolean functions, namely *discrepancy* and *corruption*. These measures have been classically used for proving randomized communication complexity lower bounds with 2-sided error. We define these measures formally below:

Definition 5. Given an boolean function $f : X \times X \to \{0,1\}$, a rectangle $R \subseteq X \times X$, and a distribution μ on $X \times X$, we define the *discrepancy of f on R under μ* as

$$\mathrm{Disc}_\mu(R, f) = \left|\mu(R \cap f^{-1}(1)) - \mu(R \cap f^{-1}(0))\right|$$

and the *discrepancy* of f under μ, denoted by $\mathrm{Disc}_\mu(f)$, as the maximum over all rectangles $R \subseteq X \times X$ of $\mathrm{Disc}_\mu(R, f)$.

Intuitively, a function f that has low discrepancy does not have any large rectangle that is nearly-monochromatic. Therefore any randomized protocol for f must necessarily "use" small rectangles resulting in large randomized communication complexity. Discrepancy is a two-sided measure since it implies that neither $f^{-1}(0)$ nor $f^{-1}(1)$ approximately contains any large rectangles. A one-sided analogue of discrepancy is *corruption* which only rules out large rectangles in one of $f^{-1}(0)$ or $f^{-1}(1)$. (For simplicity of notation we only state the version that rules out large rectangles in $f^{-1}(0)$.)

Definition 6. Given an boolean function $f : X \times X \to \{0,1\}$ and a distribution μ on $X \times X$, we say that f has *corruption* (C, Δ) under μ if and only if for every rectangle $R \subseteq X \times X$ with $\mu(R) \geq 2^{-C}$,

$$\mu(R \cap f^{-1}(1)) \geq \Delta \cdot \mu(R).$$

Corruption bounds are useful for proving lower bounds for functions such as set intersection that have low nondeterministic communication complexity. For such a function f, there exist large monochromatic rectangles in $f^{-1}(1)$ of f which implies that the discrepancy of f is large. On the other hand, a good corruption bound implies that the number of rectangles that are nearly monochromatic with the zeros of f must be large.

3. PARITY OF FUNCTIONS

In this section, we prove lower bounds for functions in the read/write stream model that are obtained by taking the parity of many suitable copies of a primitive function.

Definition 7. Given a function $f : X \times X \to \{0,1\}$ and a permutation ϕ on $[m]$, define $f_\phi^\oplus$ on X^{2m} by

$$f_\phi^\oplus(v_1, \ldots, v_m, v'_1, \ldots, v'_m) = \bigoplus_{i=1}^m f(v_i, v'_{\phi(i)}).$$

We have the following general result.

Theorem 2. *Let $\delta < 1/2$. Let $f : X \times X \to \{0,1\}$ for some set X with $n = \lceil \log_2 |X| \rceil$ and let μ be a probability distribution on $X \times X$ such that either*

(a) $Disc_\mu(f) \leq 2^{-\gamma n}$ for some $\gamma > 0$ or

(b) f has corruption $(\gamma n, \rho)$ under μ and $\mu(f^{-1}(0)) \geq \eta$, for some $\gamma > 0$ and constants $\rho, \eta > 0$.

For any integer $t \geq 1$, $\epsilon > 0$ and $c \geq 4/\varepsilon$ such that $m = n^{1/c}$ is a sufficiently large integer, there is constant $a > 0$ depending only on c and t with the following property:

Let $N = 2mn$, $r \leq a \log_2 N$ and $s \leq N^{1-\varepsilon}$. Then there is no randomized (r,s,t)-read/write stream algorithm with 2-sided error at most δ that can solve $f_{\phi^*}^\oplus$ on X^{2m}.

Proof. For each of the 2 cases above, we prove the lower bound by an argument that prunes the inputs belonging to a skeleton in a careful manner so as to create a rectangle that contradicts the corresponding hardness measure of f.

Part (a). We first consider the case where f has low discrepancy. Suppose that $c \geq 4/\varepsilon$ and $m = n^{1/c}$ is integer and that we have an (r, s, t)-read/write stream algorithm as above. We can assume without loss of generality that the error δ of the randomized (r, s, t)-read/write stream algorithm in computing $f^{\oplus}_{\phi^*}$ is at most $1/12$ by repeating it a constant number of times and taking the majority of the answers. This only increases the number of reversals by a constant factor and adds a constant to the space used.

Define a probability distribution $\nu = \mu^m_{\phi^*}$ on X^{2m} by choosing $(v_i, v'_{\phi^*(i)})$ from $X \times X$ according to μ independently for each $i \in [m]$ and interleaving them to produce $(v_1, \ldots, v_m, v'_1, \ldots, v'_m) \in X^{2m}$. We use Yao's principle to derive a *deterministic* (r, s, t)-read/write stream algorithm A that computes $f^{\oplus}_{\phi^*}$ on X^{2m} with error probability at most $\delta \leq 1/12$ under distribution ν.

As in Proposition 1, let σ be the function that maps each input $\mathbf{v} \in X^{2m}$ to its skeleton and define $\kappa = |\sigma(X^{2m})|$. Let L be the set of $\xi \in \sigma(X^{2m})$ on which A has relative error at most 2δ on $\sigma^{-1}(\xi)$; i.e., $\nu(\{\mathbf{v} \in \sigma^{-1}(\xi) \mid \mathsf{A}(\mathbf{v}) \neq f^{\oplus}_{\phi^*}(\mathbf{v})\}) \leq 2\delta \cdot \nu(\sigma^{-1}(\xi))$. By Markov's inequality,

$$\sum_{\xi \in L} \nu(\sigma^{-1}(\xi)) \geq \nu(\bigcup_{\xi \in L} \sigma^{-1}(\xi)) \geq 1/2.$$

Thus, there exists a $\xi \in L \subseteq \sigma(X^{2m})$ such that $\nu(\sigma^{-1}(\xi)) \geq 1/(2|L|) \geq 1/(2\kappa)$ and A has relative error at most 2δ on $\sigma^{-1}(\xi)$. Fix such a ξ.

Since $N = 2nm = 2m^{c+1}$, we have $\log_2 N \leq 2c \log_2 m$. Therefore for sufficiently small $a > 0$ depending only on c and t, we have $t^{2r} \leq t^{2a \log_2 N} \leq t^{4ac \log_2 m} \leq \sqrt{m}/100$ and so $t^{2r} \cdot \mathsf{sortedness}(\phi^*) \leq m/50$. From Proposition 1(c), there exists a set I depending on ξ, with

$$|I| = m - t^{2r} \cdot \mathsf{sortedness}(\phi^*) \geq 49m/50 > 1$$

such that for every $i \in I$, for any $\mathbf{a} \in X^J$ where $J = [2m] \setminus \{i, m + \phi^*(i)\}$, the set $S^i_{\mathbf{a}}$ defined below is an $(i, m + \phi^*(i))$-rectangle:

$$S^i_{\mathbf{a}} = \{\mathbf{v} \in X^{2m} \mid \sigma(\mathbf{v}) = \xi \text{ and } \mathbf{v}_J = \mathbf{a}\}$$

Fix any $i \in I$ and for brevity let $i' = m + \phi^*(i)$. By definition, the sets $S^i_{\mathbf{a}}$ for $\mathbf{a} \in X^J$ partition $\sigma^{-1}(\xi)$ and each $S^i_{\mathbf{a}} = \{\mathbf{a}\} \times R_{\mathbf{a}}$ where $R_{\mathbf{a}} = (S^i_{\mathbf{a}})_{\{i,i'\}}$ is the projection of $S^i_{\mathbf{a}}$ on $\{i, i'\}$. For each $\mathbf{a}$, by definition $R_{\mathbf{a}}$ is a rectangle on $X \times X$. Moreover $\nu(S^i_{\mathbf{a}}) = p_{\mathbf{a}} \cdot \mu(R_{\mathbf{a}})$ where $p_{\mathbf{a}} = \mu^{m-1}(\{\mathbf{a}\})$ and $\sum_{\mathbf{a}} p_{\mathbf{a}} = 1$ since ν is a product distribution over coordinates. Now

$$\sum_{\mathbf{a}} p_{\mathbf{a}} \cdot \mu(R_{\mathbf{a}}) = \sum_{\mathbf{a}} \nu(S^i_{\mathbf{a}}) = \nu(\sigma^{-1}(\xi)) \geq 1/(2\kappa).$$

Therefore by Markov's inequality there is an $\mathbf{a}$ such that $\mu(R_{\mathbf{a}}) \geq 1/(4\kappa)$ and the relative error of A on $S^i_{\mathbf{a}} = \{\mathbf{a}\} \times R_{\mathbf{a}}$ is at most $4\delta \leq 1/3$. (The total ν measure of $S^i_{\mathbf{a}}$ with $\mu(R_{\mathbf{a}}) \leq 1/(4\kappa)$ is at most $1/(4\kappa) \leq \nu(\sigma^{-1}(\xi))/2$. Discard these. Since the average error on $\sigma^{-1}(\xi)$ is at most 2δ, the average error under ν on the remaining half is at most twice that or 4δ and choosing any remaining $\mathbf{a}$ with at most the average error suffices.)

Now by Proposition 1(b), A outputs the same answer, call it $b \in \{0, 1\}$, on all inputs $\mathbf{v}$ in $S^i_{\mathbf{a}} = \{\mathbf{a}\} \times R_{\mathbf{a}}$. Since all coordinates of $\mathbf{v}$ outside of $\{i, i'\}$ are fixed to $\mathbf{a}$, their contribution to $f^{\oplus}_{\phi^*}(\mathbf{v})$ is some constant $b' \in \{0, 1\}$. Since A has relative error at most $1/3$ under ν on $S^i_{\mathbf{a}}$, we must have that $f(v_i, v_{i'}) = b \oplus b'$ for at least $2/3$ of $(v_i, v_{i'}) \in R_{\mathbf{a}}$ under distribution μ. Therefore

$$\mathrm{Disc}_\mu(f) \geq \mathrm{Disc}_\mu(f, R_{\mathbf{a}}) \geq \mu(R_{\mathbf{a}})/3 \geq 1/(12\kappa),$$

and thus $\kappa \geq 2^{\gamma n}/12$ by the assumption on $\mathrm{Disc}_\mu(f)$. Therefore $\log_2 \kappa \geq \gamma m^c - 4$.

Now by Proposition 1(a), there is a constant d depending on t such that $\log_2 \kappa \leq dm^2(r \cdot s + \log_2 nm)$. Recalling that $N = 2nm = 2m^{c+1}$, we have $r \leq a \log_2 N \leq 2ac \log_2 m$ and $s \leq N^{1-\varepsilon} \leq (2m^{c+1})^{(1-\varepsilon)}$, so we have

$$\log_2 \kappa \leq dm^2(4ac\, m^{(c+1)(1-\varepsilon)} \log_2 m + (c+1) \log_2 m).$$

Combining the upper and lower bounds on $\log_2 \kappa$, it follows that

$$m^c \leq c' m^{(c+1)(1-\varepsilon)+2} \log_2 m \tag{1}$$

for some constant $c' > 0$ depending on c, d, γ, and a. Since $c \geq 4/\varepsilon$, $(c+1)(1-\varepsilon) + 2 = c + 3 - (c+1)\varepsilon \leq c - 1 - \varepsilon < c$ which contradicts (1) for m sufficiently large. This concludes the proof for part (a).

Part (b). We now turn to the case where f has high corruption. The proof for this case follows the same outline as before but the manner in which we account for inputs is slightly more involved. As in the previous case, we will use Yao's principle to derive a lower bound for a deterministic read/write stream algorithm A with error $\delta = \min\{\eta\rho/32, 1/8\}$.

Call an input $\mathbf{v} \in X^{2m}$ *rich* if the fraction of coordinates where f evaluates to 0 is at least $\eta/2$. Since $\mu(f^{-1}(0)) = \eta$ and the distribution $\nu = \mu^m$ on X^{2m} is a product-wise distribution, the set of poor inputs has negligible measure under ν. In fact, it can be shown using the Chernoff bound that this is exponentially small in m. Thus, we can assume wlog that A gives the right answer only on rich inputs and that the error of A is at most 2δ. Let Bad denote the set of inputs on which A makes an error.

Since $\mu(f^{-1}(0)) \geq \eta$, a simple calculation shows that the measure under ν of the 0's of $f^{\oplus}$ is biased away from $1/2$ by only an exponentially small quantity. Now consider the set of skeletons on which A outputs a 1. The 0's of $f^{\oplus}$ which get mapped to these skeletons can have measure at most 2δ under ν. Therefore, the set of skeletons in which A outputs a 0 must account for the 0's of $f^{\oplus}$ whose measure is at least $1/2 - \exp(-\Theta(m)) - 2\delta \geq 1/4$. Now, using the same proof as above, we obtain a skeleton ξ on which A outputs a 0 such that $\nu(\sigma^{-1}(\xi)) \geq 1/(8\kappa)$ and A has relative error at most 4δ on $\sigma^{-1}(\xi)$. Similarly, for this skeleton ξ, there exists a set I with $|I| \geq m(1 - \eta/4)$ such that for every $i \in I$, for any $\mathbf{a} \in X^J$ where $J = [2m] \setminus \{i, m + \phi^*(i)\}$, the set $S^i_{\mathbf{a}}$ as defined above is an $(i, m + \phi^*(i))$-rectangle. As before, for every $i \in I$, the sets $S^i_{\mathbf{a}}$ partition the set of inputs $\sigma^{-1}(\xi)$ as we vary $\mathbf{a}$. Unlike the previous proof, however, we will only be interested in those $\mathbf{a}$'s that contribute a 0 to $f^{\oplus}_{\phi^*}$ i.e. $\bigoplus_{j \in [m] \setminus i} f(a_j, a_{m+\phi^*(j)}) = 0$, and furthermore, we will not fix an $i \in I$. For any $i \in I$ and such an $\mathbf{a}$, we call $S^i_{\mathbf{a}}$ a *fragment* of ξ.

Now, every rich input $\mathbf{v}$ such that $f^{\oplus}(\mathbf{v}) = 0$ and $\sigma(\mathbf{v}) = \xi$ has at least $m\eta/2$ coordinate pairs where f evaluates to 0. Therefore, $\mathbf{v}$ belongs to at least $m(\eta/2 - \eta/4) = m\eta/4$ fragments $S^i_{\mathbf{a}}$, where $i \in I$, and $\mathbf{a}$ is the projection of $\mathbf{v}$ onto $[2m] \setminus \{i, m + \phi^*(i)\}$. Write $S^i_{\mathbf{a}} = \{\mathbf{a}\} \times R_{\mathbf{a}}$, where $R_{\mathbf{a}}$ is a rectangle, so that $\nu(S^i_{\mathbf{a}}) = \mu^{(m-1)}(\mathbf{a}) \cdot \mu(R_{\mathbf{a}})$. Since we assumed that every input on which A gives the correct output is rich, we sum over all fragments to obtain

$$\begin{aligned}\sum \nu(S^i_{\mathbf{a}}) &\geq \frac{m\eta}{4} \cdot \nu(\sigma^{-1}(\xi) \cap \overline{\mathsf{Bad}}) \\ &\geq \frac{m\eta}{4} \cdot (1 - 4\delta) \cdot \nu(\sigma^{-1}(\xi)) \\ &\geq \frac{m\eta}{8} \cdot \nu(\sigma^{-1}(\xi)) \geq \frac{m\eta}{64\kappa} \qquad (2)\end{aligned}$$

On the other hand, each input on which A errs can be in no more than m fragments. Therefore, $\sum \nu(S^i_{\mathbf{a}} \cap \mathsf{Bad}) \leq 2\delta m \cdot \nu(\sigma^{-1}(\xi))$. Combining this with (2), we obtain

$$\sum \nu(S^i_{\mathbf{a}} \cap \mathsf{Bad}) \leq \frac{16\delta}{\eta} \cdot \sum \nu(S^i_{\mathbf{a}}) \leq \frac{\rho}{2} \cdot \sum \nu(S^i_{\mathbf{a}}).$$

Hence, there exists a fragment $S^i_{\mathbf{a}}$ with relative error at most ρ and $\mu(R_{\mathbf{a}}) \geq \eta/(128\kappa)$.

Now A outputs the same answer 0 on all inputs $\mathbf{v}$ in $S^i_{\mathbf{a}} = \{\mathbf{a}\} \times R_{\mathbf{a}}$. Since $\mathbf{a}$ contributes a 0 to $f^{\oplus}_{\phi^*}$, and A has relative error at most ρ under ν on $S^i_{\mathbf{a}}$, we must have that $f(v_i, v_{m+\phi^*(i)}) = 0$ for at least 2/3 of $(v_i, v_{m+\phi^*(i)}) \in R_{\mathbf{a}}$ under distribution μ. Thus, this rectangle is not highly corrupt, so $2^{-\gamma n} \geq \mu(R_{\mathbf{a}}) \geq \eta/(128\kappa)$. The rest of the argument is similar to that of part (a). □

By taking the inner product function IP on n bits ($IP(x, y) = \sum_{i=1}^{n} x_i \cdot y_i \mod 2$) to be the primitive function, and using the well-known discrepancy bound for IP [6, 2], Theorem 2 implies:

Corollary 3. *For any constants $\varepsilon > 0$, $\delta < 1/2$, there is no randomized $(o(\log N), o(N^{1-\varepsilon}), O(1))$-read/write stream algorithm with 2-sided error at most δ solving $IP^{\oplus}_{\phi^*}$ on N bit inputs.*

4. DISJUNCTION OF FUNCTIONS

In this section, we prove lower bounds for functions in the read/write stream model that are obtained by taking the disjunction of many suitable copies of a primitive function.

Definition 8. Given a function $f : X \times X \to \{0, 1\}$ and a permutation ϕ on $[m]$, define $f^{\vee}_{\phi}$ on X^{2m} by

$$f^{\vee}_{\phi}(v_1, v_2, \ldots, v_m, v'_1, v'_2, \ldots, v'_m) = \bigvee_{i=1}^{m} f(v_i, v'_{\phi(i)}).$$

Theorem 4. *Let $\delta < 1/2$. Let $f : X \times X \to \{0, 1\}$ for some set X with $n = \lceil \log_2 |X| \rceil$ and let μ be a probability distribution on $X \times X$ such that $\mu(f^{-1}(1)) \leq 1/m$. For any integer $t \geq 1$, $\epsilon > 0$, $0 < \gamma \leq 1$, and $c \geq 4/\varepsilon$ such that $m = n^{1/c}$ is a sufficiently large integer, and f has corruption (n^{γ}, Δ) under μ for $\Delta > 0$, there is constant $a > 0$ depending only on c, δ and t with the following property:*
Let $N = 2mn$, $r \leq a\frac{\log_2 N}{\log_2(1/(m\Delta))}$ and $s \leq N^{\gamma-\varepsilon}$. Then there is no randomized (r, s, t)-read/write stream algorithm with 2-sided error at most δ that can solve $f^{\vee}_{\phi^*}$ on X^{2m}.

Proof. Suppose that $c \geq 4/\varepsilon$ and $m = n^{1/c}$ is integer and that we have a randomized (r, s, t)-read/write stream algorithm as above. Note that by the definition of corruption, $\Delta \leq 1/m$ since $\mu(f^{-1}(1)) \leq 1/m$. By repeating the algorithm $O_{\delta}(\log(1/(m\Delta)))$ times and taking the majority of the answers we can reduce the error to at most $m\Delta/16$. Thus we obtain a randomized (r', s', t)-read/write stream algorithm with $r' \leq c' r \log_2(1/(m\Delta)) \leq c' a \log_2 N$ and $s' \leq c's$ for some constant c' depending only on δ and 2-sided error at most $m\Delta/16$.

Define a probability distribution $\nu = \mu^m_{\phi^*}$ on X^{2m} by choosing $(v_i, v'_{\phi^*(i)})$ from $X \times X$ according to μ independently for each $i \in [m]$ and interleaving them to produce $(v_1, \ldots, v_m, v'_1, \ldots, v'_m) \in X^{2m}$. We use Yao's principle to derive a deterministic (r', s', t)-read/write stream algorithm A that computes $f^{\oplus}_{\phi^*}$ on X^{2m} with error probability at most $\delta' \leq m\Delta/16$ under distribution ν.

Since $\mu(f^{-1}(1)) \leq 1/m$, $\mu(f^{-1}(0)) \geq 1 - 1/m$ and by the independence of the coordinates, the probability under ν that $f^{\vee}_{\phi^*}(\mathbf{v}) = 0$ is at least $(1 - 1/m)^m \geq 1/4$. It follows that since A has error at most $m\Delta/16 \leq 1/16$, we must have $\nu(\mathsf{A}^{-1}(0)) \geq 1/6$.

As in Proposition 1, let σ be the function that maps each input $\mathbf{v} \in X^{2m}$ to its skeleton and define $\kappa = |\sigma(X^{2m})|$. By Proposition 1(b), for any $\mathbf{v} \in X^{2m}$, $\mathsf{A}(\mathbf{v}) = 0$ if and only if $\sigma(\mathbf{v}) \in \sigma(\mathsf{A}^{-1}(0))$. By Markov's inequality, there exists a $\xi \in \sigma(\mathsf{A}^{-1}(0))$ such that $\nu(\sigma^{-1}(\xi)) \geq 1/(12\kappa)$ and A has relative error at most $2\delta'$ on $\sigma^{-1}(\xi)$ under ν. Fix such an ξ and let $S = \sigma^{-1}(\xi) \subseteq \mathsf{A}^{-1}(0)$. Then we have $\nu(S) \geq 1/(12\kappa)$, $\sigma(\mathbf{v}) = \xi$ and $\mathsf{A}(\mathbf{v}) = 0$ for all $\mathbf{v} \in S$, and $\nu(S \cap f^{-1}(1)) \leq 2\delta' \cdot \nu(S)$.

Since $N = 2nm = 2m^{c+1}$, we have $\log_2 N \leq 2c \log_2 m$. Therefore for sufficiently small $a > 0$ depending on c, c' and t, we have $t^{2r'} \leq t^{2ac' \log_2 N} \leq t^{4ac'c \log_2 m} \leq \sqrt{m}/100$ and so $t^{2r'} \cdot \mathsf{sortedness}(\phi^*) \leq (\sqrt{m}/100) \cdot 2\sqrt{m} \leq m/50$. By Proposition 1(c), there exists a set I with $|I| = m - t^{2r} \cdot \mathsf{sortedness}(\phi^*) \geq 49m/50$ such that for every $i \in I$, for any $\mathbf{a} \in X^J$ where $J = [2m] \setminus \{i, m + \phi^*(i)\}$, the set $\{\mathbf{v} \in X^{2m} \mid \sigma(\mathbf{v}) = \xi \text{ and } \mathbf{v}_J = \mathbf{a}\}$ is an $(i, m + \phi^*(i))$-rectangle. Let $m' = |I| \geq 49m/50$.

Let $K = [2m] \setminus (I \cup \{m + \phi^*(i) \mid i \in I\})$. For each $\mathbf{b} \in X^K$, define

$$S^I_{\mathbf{b}} = \{\mathbf{v} \in X^{2m} \mid \sigma(\mathbf{v}) = \xi \text{ and } \mathbf{v}_K = \mathbf{b}\}.$$

The sets $S^I_{\mathbf{b}}$ for various $\mathbf{b} \in X^K$ partition S. Moreover, we can write each $S^I_{\mathbf{b}} = \{\mathbf{b}\} \times T_{\mathbf{b}}$ for some set $T_{\mathbf{b}} \subseteq X^{[2m] \setminus K}$. Let ν_I be the induced probability distribution on the coordinates $[2m] \setminus K = I \cup \{m + \phi^*(i) \mid i \in I\}$. By Markov's inequality there is a $\mathbf{b}$ such that $\nu_I(T_{\mathbf{b}}) \geq 1/(24\kappa)$ and the relative error of A on $S^I_{\mathbf{b}} = \{\mathbf{b}\} \times T_{\mathbf{b}}$ is at most $4\delta'$.

We say that $\mathbf{v} = (v_1, \ldots, v_m, v'_1, \ldots, v'_m)$ is *one in position* i if $f(v_i, v'_{\phi^*(i)}) = 1$ (and is *zero in position* i otherwise). Since $S^I_{\mathbf{b}}$ has relative error at most $4\delta' \leq 1/4 < 1$, every $\mathbf{v} \in S^I_{\mathbf{b}}$ is zero in every position in $[2m] \setminus I$ (the positions of $\mathbf{b}$). $S^I_{\mathbf{b}}$ may contain inputs that are one in multiple positions. Let $T \subseteq T^I_{\mathbf{b}} \subseteq X^I$ be such that $\{\mathbf{b}\} \times T$ consists of all inputs in $S^I_{\mathbf{b}}$ that are one in at most one position. Since A outputs 0 on all inputs in $S^I_{\mathbf{b}}$ and has relative error at most $4\delta' \leq 1/4$ on $S^I_{\mathbf{b}}$ it follows that $\nu_I(T) \geq \frac{3}{4}\nu_I(T^I_{\mathbf{b}}) \geq 1/(32\kappa)$ and the relative error of A on $\{\mathbf{b}\} \times T$ is still at most $4\delta'$.

Let $T_0 \subset T$ be chosen so that $\{\mathbf{b}\} \times T$ is the set of zero inputs in $\{\mathbf{b}\} \times T$ and $T_1 = T \setminus T_0$. Write $V^0 = \{\mathbf{b}\} \times T_0$ and $V^1 = \{\mathbf{b}\} \times T_1$ and $V = V^0 \cup V^1$. Then $\nu_I(V^1_I) \leq 4\delta'\ \nu_I(V_I)$

and $\nu_I(V_I) \geq 1/(32\kappa)$. We now define an undirected graph G with vertex set V and edges with labels from I. For distinct $\mathbf{u}, \mathbf{v} \in V$, there is an edge between $\mathbf{u}$ and $\mathbf{v}$ labeled i if and only if $\mathbf{u}_J = \mathbf{v}_J$ for $J = [2m] \setminus \{i, m + \phi^*(i)\}$ and every $j \in I \setminus \{i\}$ is a zero position for both inputs. Observe that for each $i \in I$, the edge relation given by the edges labelled i is an equivalence relation; that is, for each $i \in I$ the edges labelled i partition V into cliques.

By Proposition 1(c), the vertices of each a clique of i-edges in G form an $(i, m+\phi^*(i))$-rectangle. By definition, because each $\mathbf{v} \in V^1$ is one in precisely one position, if that position is i, the only possible edges incident to $\mathbf{v}$ have label i. On the other hand, vertices $\mathbf{v} \in V^0$ may have incident edges with as many as m' different labels. For each $i \in I$, let $V^{1,i}$ denote the set of vertices in V^1 that are one in position i. Since $\sum_{i \in I} \nu_I(V_I^{1,i}) = \nu_I(V_I^1) \leq 4\delta' \ \nu_I(V_I)$, there is some i such that $\nu_I(V_I^{1,i}) \leq 4\delta' \ \nu_I(V_I)/m'$.

Fix this i and let $U = V^0 \cup V^{1,i}$. By construction, all edges of G with label i have endpoints in U and the cliques of these edges partition U. Moreover, $\nu_I(U_I) \geq \nu_I(V_I^0) \geq 1/(48\kappa)$ and

$$\begin{aligned} \nu_I((V^1 \cap U)_I) &\leq 4\delta' \ \nu_I(V_I)/m' \\ &\leq \frac{16}{3}\delta' \ \nu_I(U_I)/m' \leq 6\delta' \ \nu_I(U_I)/m \end{aligned}$$

so A's output of 0 has relative error at most $6\delta'/m$ on U under ν_I. Observe that each clique of edges labelled i corresponds to a unique assignment $\mathbf{c} \in X^L$ where $L = [2m] \setminus (K \cup \{i, i + \phi^*(m)\})$ (since the assignment $\mathbf{b} \in X^K$ has already been fixed). Thus the relative distribution within each such clique given by ν_I is the same as μ on coordinates $\{i, m + \phi^*(i)\}$. By Markov's inequality, there is a $\mathbf{c} \in X^L$ such that the clique corresponding to $\mathbf{c}$ has μ measure at least $\nu_I(U_I)/2 \geq 1/(96\kappa)$ and A's output of 0 has relative error at most $12\delta'/m$ under μ. Since this clique is an $(i, m + \phi^*(i))$-rectangle, it consists of $\{\mathbf{b}\} \times \{\mathbf{c}\} \times R$ for some rectangle $R \in X^{\{i, m+\phi^*(i)\}}$ with $\mu(R) \geq 1/(96\kappa)$ and $\mu(R \cap f^{-1}(1)) \leq 12\delta'\mu(R)/m \leq 3\Delta\mu(R)/4$.

Since f has corruption (n^γ, Δ) under μ, it must be the case that $1/(96\kappa) < 2^{-n^\gamma}$ and thus $\log_2 \kappa \geq n^\gamma - 7$.

Now by Proposition 1(a), there is a constant d depending on t such that $\log_2 \kappa \leq dm^2(r \cdot s + \log_2 nm)$. Recalling that $N = 2nm = 2m^{c+1}$, we have $r' \leq ac' \log_2 N \leq 2ac'c \log_2 m$ and $s' \leq c'N^{\gamma-\varepsilon} \leq c'(2m^{c+1})^{(\gamma-\varepsilon)}$, so we have $\log_2 \kappa \leq dm^2(4a(c')^2 c \, m^{(c+1)(\gamma-\varepsilon)} \log_2 m + (c+1)\log_2 m)$. Combining the upper and lower bounds on $\log_2 \kappa$, it follows that

$$m^{c\gamma} \leq c'' m^{(c+1)(\gamma-\varepsilon)+2} \log_2 m \tag{3}$$

for some constant $c'' > 0$ depending on c, c', d, and a. Since $c \geq 4/\varepsilon$, $(c+1)(\gamma - \varepsilon) + 2 \leq c\gamma - c\varepsilon + 2 + \gamma \leq c\gamma - 1 < c\gamma$ which contradicts (3) for m sufficiently large. □

5. LOWER BOUNDS FOR SET DISJOINTNESS AND OTHER PROBLEMS

In this section, we prove lower bounds for the set disjointness problem using the lower bounds proved in Section 4. We then prove similar lower bounds for other problems (mostly via reductions from the set disjointness problem).

First let us consider the set disjointness problem. Given sets A and B, $\textsc{Disj}(A, B) = 0$ if and only if $A \cap B = \emptyset$.

THEOREM 5. *Let $\varepsilon > 0$ and $\delta < 1/2$ be any constant real numbers. If r is $o(\log N / \log\log N)$ and s is $o(N^{1-\varepsilon})$ for some large enough N, then for every $t \geq 1$, there is no randomized (r, s, t) read/write stream algorithm with 2-sided error at most δ that can decide* $\textsc{Disj}$ *on inputs of size N.*

The proof of the above result follows by a reduction from a related problem $SD^{\vee}_{\phi^*}$ where the primitive function

$$SD(u, v) = \begin{cases} 1 & \text{if } u \cap v = \emptyset \\ 0 & \text{otherwise,} \end{cases}$$

where $u, v \in X = \{0, 1\}^n$.

Let $\mathcal{D}$ denote the strings in $\{0, 1\}^n$ that have weight exactly $\sqrt{\frac{n}{m}}$. We will need the following corruption result for SD in order to apply Theorem 4.

LEMMA 6. *Under the uniform distribution μ on $\mathcal{D} \times \mathcal{D}$ SD has corruption $\left(\frac{1}{16}\sqrt{\frac{n}{m}}, \frac{1}{96m\log_2 m}\right)$.*

PROOF. The following is a generalization of an argument of Babai, Frankl, and Simon [2]. Its appears in a somewhat different form in [4]. Let $k = \sqrt{n/m}$. Let $R = A \times B \subseteq \mathcal{D} \times \mathcal{D}$ be a rectangle with $|R| \geq 2^{-k/16}|\mathcal{D}|^2$. Set $\epsilon = \Pr_\mu[S \cap T \neq \emptyset \mid (S, T) \in R]$. Assume for the purposes of contradiction that $\epsilon \leq 1/(96m\log_2 m)$. Since $|\mathcal{D}| = \binom{n}{k}$,

$$|A| \geq 2^{-k/16}\binom{n}{k} \geq 4(8/9)^{k/2}\binom{n}{k}.$$

For $S \in A$, define ϵ_S to be the fraction of elements T of B such that $S \cap T \neq \emptyset$. Let $A' \subseteq A$ be the set of $S \in A$ such that $\epsilon_S \leq 2\epsilon$. By Markov's inequality, $|A'| \geq |A|/2 \geq 2(8/9)^{k/2}\binom{n}{k}$.

PROPOSITION 7. *Let $d \geq 3$ and let A' be a collection of k-subsets of $[n]$. If*

$$|A'| > 2(4(d-1)/d^2)^{k/2}|\mathcal{D}| = 2(4(d-1)/d^2)^{k/2}\binom{n}{k}$$

then A' contains a sequence of $p = \lceil n/(dk) \rceil$ sets $S_1, \ldots, S_p$ such that $|S_j \cap (\bigcup_{i<j} S_i)| \leq k/2$ for $j = 1, \ldots, p$, i.e. at least half the elements of S_j do not occur in earlier sets.

PROOF OF PROPOSITION 7. We construct $S_1, \ldots, S_p$ inductively. Select $S_1 \in A'$ arbitrarily. For $j > 1$, having chosen $S_1, \ldots, S_{j-1}$, we show that for $j \leq p$, the number of sets that have more than half their elements in earlier sets is less than $|A'|$ and so we can select $S_j \in A'$ as required. Let $U_j = \bigcup_{i<j} S_i$. Since $j \leq \lceil n/(dk) \rceil$, $|U_j| \leq n/d$, the number of k-subsets of $[n]$ having more than half their elements in U_j is at most

$$\sum_{h \geq k/2} \binom{|U_j|}{h}\binom{n - |U_j|}{k-h} \leq \sum_{h \geq k/2} \binom{n/d}{h}\binom{(1-1/d)n}{k-h}$$

. It is easy to check that since $d \geq 3$ as h increases, each successive term is at most half the previous so the sum is at most $2\binom{n/d}{\lceil k/2 \rceil}\binom{(1-1/d)n}{\lfloor k/2 \rfloor}$. Using the easily verifiable inequalities that for $b \geq a \geq c \geq d$,

$$\binom{a}{c}\binom{b}{d} < (\frac{4ab}{a+b})^c \binom{(a+b)/2}{c}\binom{(a+b)/2}{d}$$

and

$$\binom{n/2}{c}\binom{n/2}{d} \leq \binom{n}{c+d}$$

we upper bound this strictly by $2(4(d-1)/d^2)^{k/2}\binom{n}{k}$ which is less than $|A'|$. □

We continue the proof of the lemma. Apply Proposition 7 with $d = 3$ to the set A' to find $p = \lceil n/(3k) \rceil$ sets $S_1, \ldots, S_p$ in A' each of which contains at least $k/2$ elements not occurring in earlier sets. For each $T \in B$, let w_T be the number of S_j that intersect it. Since each $S_j \in A'$, $\frac{1}{|B|}\sum_T w_T \leq 2\epsilon p$, so at most half of the $T \in B$ have $w_T > 4\epsilon p$. Let B' be the set of $T \in B$ with $w_T \leq 4\epsilon p$. Thus $|B'| \geq |B|/2$.

We now upper bound the number of elements in B' and thus B using ϵ. An element T of B' can be described by giving a subset $J \subseteq [p]$ of $(1-4\epsilon)p$ indices such that $T \cap S_j = \emptyset$ for all $j \in J$ and then specifying T as a t-subset of the elements outside these subsets. By the claim, any collection of $(1-4\epsilon)p$ of the sets has a total of $t(1-4\epsilon)p \geq n/9$ elements since $\epsilon \leq \frac{1}{12}$. Therefore

$$|B| \leq 2|B'| \leq 2\binom{p}{4\epsilon p}\binom{8n/9}{k} < 2^{1+H_2(4\epsilon)(\frac{n}{3k})}(8/9)^k\binom{n}{k}$$
$$\leq 2^{1+H_2(4\epsilon)(\frac{n}{3k})-\frac{k}{6}}\binom{n}{k} \qquad (4)$$

We have $k = \lceil\sqrt{n/m}\rceil$ so $\frac{n}{3k} \leq \frac{mk}{3}$. Now $H_2(\delta) \leq 2\delta\log_2(\frac{1}{\delta})$ for $\delta \leq 1/2$, so if $\epsilon \leq 1/(96m\log_2 m)$ then

$$H_2(4\epsilon) \leq \frac{2\log_2(24m\log_2 m)}{24m\log_2 m} \leq \frac{1}{6m}$$

for $m \geq 200$. Putting the above two inequalities along with (4),

$$|B| \leq 2^{1+\frac{k}{12}-\frac{k}{6}}\binom{n}{k} \leq 2^{-\frac{k}{16}}\binom{n}{k} = 2^{-\frac{k}{16}}|\mathcal{D}|$$

Thus $|R| = |A| \cdot |B| < |\mathcal{D}| \cdot 2^{-k/16}|\mathcal{D}| = 2^{-k/16}|\mathcal{D}|^2$, contradicting the size lower bound on R. □

Thus, Lemma 6 and Theorem 4 imply the following result.

Corollary 8. *The following holds for the $SD^{\vee}_{\phi^*}$ problem on $(\{0,1\}^n)^{2m}$. For every constants $\varepsilon > 0$ and $\delta < 1/2$, there exists a constant a such that if $r \leq a\log N/\log\log N$ and $s \leq N^{1/2-\varepsilon}$, where $N = 2nm$, then no randomized (r, s, t) read/write stream algorithm with 2-sided error at most δ that can solve $SD^{\vee}_{\phi^*}$.*

Proof of Theorem 5. We will reduce the $SD^{\vee}_{\phi^*}$ problem to the Disj problem. Recall that the input to the $SD^{\vee}_{\phi^*}$ problem is a vector $\mathbf{v} = (v_1, \ldots, v_m, v'_1, \ldots, v'_m)$, where for every $1 \leq i \leq m$, $v_i, v'_i \in \{0,1\}^n$. The reduction will produce sets A and B that have elements from the set $[m] \times [n]$. The reduction is pretty simple. The vectors $v_1, \ldots, v_m$ will contribute to the set A, while $v'_1, \ldots, v'_m$ will contribute to the set B as follows. For every $1 \leq i \leq m$ and $1 \leq j \leq n$, add the element $(\phi^*(i), j)$ to A if and only if the j^{th} bit of v_i is 1. Further, for every $1 \leq i \leq m$ and $1 \leq j \leq m$, add the element (i, j) to B if and only if the j^{th} bit of v'_i is 1.

One can easily check that $SD^{\vee}_{\phi^*}(\mathbf{v}) = 0$ if and only if Disj$(A, B) = 0$. Further, one can implement the above reduction using a $(2, O(\log(mn)), 2)$-restricted deterministic read/write stream algorithm. Recall that in the hard instances for the $SD^{\vee}_{\phi^*}$ problem (Corollary 8), for every $1 \leq i \leq m$, v_i and v'_i were n-bit strings of weight $\sqrt{n/m}$. Thus, the number of bits required to represent the sets A and B is $N = 2m \cdot \sqrt{n/m} \cdot (\log m + \log n) = O\left(\sqrt{nm}\log(nm)\right)$. Recall that $N' = nm$ was the number of bits required to represent the inputs to $SD^{\vee}_{\phi^*}$ problem. Note that this implies that $\log N = \Theta(\log N')$ and $N^{1-\varepsilon} = o(N'^{1/2-\varepsilon/2})$. Thus, applying Corollary 8 completes the proof. □

The reduction used in the proof above can be used to reduce the $IP^{\oplus}_{\phi^*}$ problem to the Intersection-Mod-2 problem (recall that for sets A and B, Intersection-Mod-2$(A, B) = |A \cap B| \mod 2$). Corollary 3 implies the following result.

Theorem 9. *Let $\varepsilon > 0$ and $\delta < 1/2$ be any constant real number. If r is $o(\log N)$ and s is $o(N^{1-\varepsilon})$ for some large enough N, then for every $t \geq 1$, there is no randomized (r, s, t)-read/write stream algorithm with 2-sided error at most δ that can decide* Intersection-mod-2 *on inputs of size N.*

We now use the lower bound of Theorem 5 to get similar lower bounds for other problems. In the ε-Num-Distinct-Elements problem, the goal is to approximate the the number of distinct elements within a factor $(1+\varepsilon)$. In the ε-Mode problem, the goal is to approximate the frequency of the most frequently occurring element in the input within a $(1+\varepsilon)$ factor.

Corollary 10. *Let $\varepsilon, \gamma > 0$ and $0 < \delta < 1/2$ be arbitrary constants. Then for every r in $o(\log(1/\varepsilon)/\log\log(1/\varepsilon))$, s in $o(1/\varepsilon^{1-\gamma})$ and $t \geq 1$, there are no randomized (r, s, t)-read/write stream algorithms with 2-sided error at most δ that can solve the ε-*Num-Distinct-Elements *problem.*

Proof. Let A and B be inputs to the Disj problem. Let C be the multi-set union of A and B. C is then the input to the ε-Num-Distinct-Elements problem. Note that if Disj$(A, B) = 0$ then Num-Distinct-Elements$(C) = |A| + |B|$ while if Disj$(A, B) = 1$ then Num-Distinct-Elements$(C) \leq |A| + |B| - 1$. Recall that in the hard instance of the Disj problem, $|A| = |B| = N$ and the space lower bound is $o(N^{1-\gamma})$ for every $\gamma > 0$. Set $\varepsilon = \frac{1}{2N-1}$. Now if A and B are disjoint then Num-Distinct-Elements$(C) = 2N$ and if A and B are not disjoint then Num-Distinct-Elements$(C) \leq \frac{2N}{1+\varepsilon}$. Thus, there exists a ε such that ε-Num-Distinct-Elements cannot be computed by a randomized read/write stream algorithm that uses space $o(N^{1-\gamma}) = o(1/\varepsilon^{1-\gamma})$. □

Corollary 11. *Let $\varepsilon, \gamma > 0$ and $0 < \delta < 1/2$ be arbitrary constant real numbers. For large N and for every r in $o(\log N/\log\log N)$, s in $o(N^{1-\gamma})$ and $t \geq 1$, there are no randomized (r, s, t)-read/write stream algorithms with 2-sided error at most δ that can solve $(1-\epsilon)$-*Mode *on N bit inputs*

Proof. We again start with sets A and B that are inputs to the Disj problem. Let C be the multi-set union of A and B. The simple observation in this case is that if Disj$(A, B) = 0$ then Mode$(C) = 1$ and if Disj$(A, B) = 1$ then Mode$(C) \geq 2$. Thus, a randomized read/write stream algorithm that can solve the $(1-\varepsilon)$-Mode problem can also decide the Disj problem with the same number of reversal and space requirement. Thus, for any $\gamma > 0$, no randomized read/write stream algorithm can solve the $(1-\varepsilon)$-Mode problem with $o(N^{1-\gamma})$ space for every $\gamma > 0$. □

Following Grohe, Koch and Schweikardt ([8]), who gave reductions from set disjointness to derive lower bounds in the deterministic read/write stream model with only 1 tape, we give lower bounds on arbitrary randomized read/write stream algorithms for the following problems related to databases: Empty-Join, XQuery-Filtering and XPath-Filtering.

We first look at the Empty-Join problem. Consider two relations $A, B \subseteq X^2$, where X is some base domain. We will use the predicate $A(x, y)$ (and $B(x, y)$) to decide if the tuple (x, y) is in the relation A (and B). Further, $A \bowtie_1 B$ will denote the *join* of the relations A and B on their first component. In other words

$$A \bowtie_1 B = \{(x, y) \mid \exists z\ A(z, x) \wedge B(z, y)\}.$$

Definition 9. The Empty-Join decision problem takes as input relations $A, B \subseteq X^2$, where X is some base domain and outputs 0 if and only if $A \bowtie_1 B = \emptyset$.

We now sketch the reduction from Disj to Empty-Join. Given the input sets $A', B' \subseteq X'$ for the Disj problem, define the input relations for Empty-Join as follows. Define $X = X' \times \{1, 2\}$ and relations A and B as

$$A = \{(a, 1) \mid a \in A'\} \text{ and } B = \{(b, 2) \mid b \in B'\}.$$

It is easy to check that Empty-Join$(A, B) = 0$ if and only if Disj$(A', B') = 0$. Further, the relations A and B can be generated by one scan of the inputs A' and B' (and thus, can be implemented by a deterministic $(1, O(\log N), 1)$-read/write stream algorithm, where $N = (|A| + |B|)\lceil \log_2 |X| \rceil$).

For the remaining problems, we assume that the reader is familiar with XQuery and XPath. Given an XML document T and an XQuery query Q, let $E(T, Q)$ denote the result of evaluating the query Q on the input document T.

Definition 10. The XQuery-Filtering decision problem takes as input an XQuery Q and a XML document T and outputs 0 if and only if $E(T, Q) = \emptyset$.

The reduction for XQuery-Filtering, uses the same reduction as the one from Disj to Empty-Join. The only difference is that the reduction needs to encode the input sets A' and B' for Disj in an XML document T. Further, one needs to define an XQuery Q that outputs a "tuple" for every tuple in $A \bowtie_1 B$. The problem of XPath-Filtering is the same as XQuery-Filtering except that the query Q is an XPath query. Again the crux of the reduction is to encode the inputs sets A' and B' for Disj in a suitable XML document and to design an XPath query Q that returns all nodes that correspond to elements in $A' \cap B'$. We refer the reader to [8] for the details of these reductions.

For both XPath-Filtering and XQuery-Filtering, we will use N to denote the number of bits needed to represent the query Q and the XML document T. For the Empty-Join problem, we will use N to denote the number of bits required to represent the input relations A and B.

Corollary 12. *Let $\varepsilon > 0$ and $0 < \delta < 1/2$ be arbitrary constant real numbers. For sufficiently large N and for every r in $o(\log N/\log\log N)$, s in $o(N^{1-\varepsilon})$ and $t \geq 1$, there are no randomized (r, s, t)-read/write stream stream algorithms with 2-sided error at most δ that can solve any of the following problems on N bit inputs:* Empty-Join, XQuery-Filtering, XPath-Filtering.

6. REFERENCES

[1] G. Aggarwal, M. Datar, S. Rajagopalan, and M. Ruhl. On the streaming model augmented with a sorting primitive. In *45th Symposium on Foundations of Computer Science (FOCS)*, pages 540–549, 2004.

[2] L. Babai, P. Frankl, and J. Simon. Complexity classes in communication complexity theory. In *27th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 337–347, Toronto, Ontario, Oct. 1986. IEEE.

[3] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. In *Proceedings of the 21st ACM Sympoisum on Principles of Database Systems (PODS)*, pages 1–16, 2002.

[4] P. Beame, M. Saks, X. Sun, and E. Vee. Time-space trade-off lower bounds for randomized computation of decision problems. *Journal of the ACM*, 50(2):154–195, 2003.

[5] J. Chen and C.-K. Yap. Reversal complexity. *SIAM J. Comput.*, 20(4):622–638, 1991.

[6] B. Chor and O. Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM J. Comput.*, 17(2):230–261, 1988.

[7] M. Grohe, A. Hernich, and N. Schweikardt. Randomized computations on large data sets: Tight lower bounds. In *Proceedings of the 25th ACM Symposium on Principles of Database Systems (PODS)*, pages 243–252, 2006.

[8] M. Grohe, C. Koch, and N. Schweikardt. Tight lower bounds for query processing on streaming and external memory data external memory. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP), LNCS3580*, pages 1076–1088, 2005.

[9] M. Grohe and N. Schweikardt. Lower bounds for sorting with few random accesses to external memory. In *Proceedings of the 24th ACM Symposium on Principles of Database Systems (PODS)*, pages 238–249, 2005.

[10] P. Indyk and D. P. Woodruff. Tight lower bounds for the distinct elements problem. In *44th Symposium on Foundations of Computer Science (FOCS)*, pages 283–292, 2003.

[11] S. Muthukrishnan. Data streams: Algorithms and applications. *Foundations and Trends in Theoretical Computer Science*, 1(2), 2006.

[12] A. A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106(2):385–390, 1992.

[13] M. Ruhl. *Efficient Algorithms for New Computational Models.* PhD thesis, Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2003.

[14] J. S. Vitter. External memory algorithms and data structures. *ACM Comput. Surv.*, 33(2):209–271, 2001.

[15] D. P. Woodruff. Optimal space lower bounds for all frequency moments. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 167–175, 2004.

APPENDIX

PROOF SKETCH FOR PROPOSITION 1. [7, 9] We give a brief overview of the main ideas that Grohe, Hernich, and Schweikardt in [7, 9] used to derive Proposition 1. (Although their original argument applied to a uniform model it applies at least as easily to the non-uniform model we consider.) The main idea is to capture the "information flow" among the tapes in various stages of the computation. For the ease of exposition, we only consider deterministic read/write stream algorithms.

Information flow in a read/write stream algorithm's execution is captured via an object called a *skeleton*, which is a condensation of the computation on a given input string at various stages. Assume that the input of N bits is the sequence of strings $\mathbf{v} = v_1, v_2, \ldots, v_{2m} \in X^{2m}$ which is written on the first tape. Also assume that the read/write stream algorithm A (more precisely A_N) makes a total of r reversals on input $\mathbf{v}$. The skeleton corresponding to $\mathbf{v}$, denoted by $\sigma(\mathbf{v})$, allows one to reconstruct the t tapes of the read/write stream algorithm just after each of the r reversals. There are $r+2$ levels in $\sigma(\mathbf{v})$ – level 0 corresponds to the beginning of the computation and level $r+1$ corresponds the end of the computation. Level k for $1 \leq k \leq r$ encodes the contents of each of the t tapes just after the k-th reversal in the following manner. The contents of the tapes are divided into "blocks". However, instead of explicitly specifying the contents of each cell, the skeleton will maintain a list of blocks it "depends" on (along with some state information). Assume without loss of generality that at any step only one of the t heads move.

The crucial observation about read/write stream algorithms is the following: When a symbol is written in a particular cell by the read/write stream algorithm between its k-th and $(k+1)$-st reversal, what is being written on that cell can only depend on the current state and the t symbols currently being scanned. However, the values of these t symbols were determined *before* the k-th reversal. In terms of blocks, this implies that any cell in a block at level $k+1$ depends on at most t blocks in level k. The blocks at level $k+1$ are then defined in such a manner that every cell in such a block depends on the *same* set of level k blocks.

We now describe the blocks and their dependencies more precisely. There are $2m+t$ blocks at level 0. The first $2m$ blocks correspond to the $2m$ input values $v_1, \ldots, v_{2m}$ while the last t values correspond to the infinite sequence of blanks at the end of each tape. As the computation proceeds, an existing block at level k can be divided into smaller blocks at level $k+1$. Such a process happens only under two scenarios: when one of the tape heads does a reversal, or when some tape head crosses the boundary between level k blocks on the same tape. We sketch the latter case since it can split blocks more often.

When some head crosses between level k blocks on the same tape, all the blocks on other tapes containing tape heads are split at the tape head position (more precisely, just behind each tape head in the direction from which it made its last move) into a new level $k+1$ block (in the direction from which the head has come) and the remainder of the level k block which may be further split. Moreover, the remaining portion of the level k block just exited by the tape head that moved across a boundary also becomes a level $k+1$ block. Each new level $k+1$ block created is said to depend on the level k blocks containing each of other tape heads just prior to the move.

The skeleton consists of a layered directed graph of blocks of in-degree t representing the block dependencies, with each block labelled by the internal state of the read/write stream algorithm immediately prior its creation as well as its left and right boundary positions and the position and direction of movement of each of the t heads (at the time it was created) within each of the blocks on which it depends.

Thus, one can think of the skeleton $\sigma(\mathbf{v})$ as a layered "circuit", where each gate (block) has t inputs as outputs from t gates in the previous layer as well as extra input from the auxiliary information labelling the block/gate. It can be verified that given $\sigma(\mathbf{v})$, the whole computation of the read/write stream algorithm on $\mathbf{v}$ can be recovered. In particular, the state information associated with the last block created at level $r+1$ determine whether or not the input $\mathbf{v}$ was accepted. This in turn implies that if for two vectors $\mathbf{u}$ and $\mathbf{v}$, $\sigma(\mathbf{u}) = \sigma(\mathbf{v})$, then either both are accepted by A_N or both are rejected from which Proposition 1(b) follows.

We sketch the arguments for parts (a) and (c). To upper bound the number of possible skeletons, let B upper bound the number of blocks at any level and let q upper bound the number of possible labellings of any block. Since each block depends on t predecessors, there are at most $(B^t q)^B$ distinct ways of creating the $(k+1)$-st layer of the skeleton from the k-th layer. Therefore, over all the layers there are at most $(B^t q)^{B(r+1)}$ skeletons. Since each level $k+1$ block is attributable to a reversal or the movement of a head out of some level k block one can show that B is $O(t^{2r} m)$. For suitable parameters, q is dominated by the bound on the number of states and t^{2r} is at most m and thus $(B^t q)^{B(r+1)}$ is approximately $2^{O(m^2 \cdot r \cdot s)}$. A more careful analysis gives the bound of part (a).

In part (c), the essential idea is to show that for a large number of pairs of the form $\{i, m+\phi(i)\}$, no block in the skeleton $\sigma(\mathbf{v})$ depends on *both* v_i and $v_{m+\phi(i)}$ and that, via a cut-and-paste argument, inputs that have the same skeleton and only differ on such positions must be in a common rectangle of inputs having the same skeleton. For the latter argument, let $\mathbf{u}$ and $\mathbf{v}$ be such that $\sigma(\mathbf{u}) = \sigma(\mathbf{v})$ and that $\mathbf{u}$ and $\mathbf{v}$ agree on all positions except $\{i, m+\phi(i)\}$. Now consider an input $\mathbf{w}$, which agrees with $\mathbf{u}$ on all positions, except that $w_i = v_i$. Essentially, $\mathbf{w}$ will "behave" either as $\mathbf{u}$ or $\mathbf{v}$ while the blocks (and their dependencies) in $\sigma(\mathbf{w})$ are being created, depending on whether the block depends on i or $m+\phi(i)$ (it cannot depend on both). As $\sigma(\mathbf{u}) = \sigma(\mathbf{v})$ and they agree outside of $\{i, m+\phi(i)\}$, one can verify that indeed $\sigma(\mathbf{w}) = \sigma(\mathbf{u}) = \sigma(\mathbf{v})$. The argument for the existence of many such pairs $\{i, m+\phi(i)\}$ crucially relies on the fact that the last m inputs in $\mathbf{v}$ are not sorted relative to the first m indices. The key property to show is that if one considers a sequence of input indices from $\{1, \ldots, 2m\}$ derived by ordering the set of input dependencies within each block arbitrarily and concatenating the sequences for the blocks at level k in an order that respects their relative order on the tapes, then such a sequence can be written as the interleaving of at most t^k increasing or decreasing sequences. Further, [7, 9] show that if $\mathsf{sortedness}(\phi)$ is small, this property severely limits the number of input pairs $\{i, m+\phi(i)\}$ that can be in the dependency sets of blocks in the skeleton. □