Upper and Lower Bounds on the Power of Advice

Arkadev Chattopadhyay [*] Jeff Edmonds [†] Faith Ellen [‡] Toniann Pitassi [§]

March 1, 2016

Abstract

Proving superpolylogarithmic lower bounds for dynamic data structures has remained an open problem despite years of research. Pătraşcu proposed an exciting approach for breaking this barrier via a two player communication model in which one player gets private advice at the beginning of the protocol. He gave reductions from the problem of solving an asymmetric version of set-disjointness in his model to a diverse collection of natural dynamic data structure problems in the cell probe model. He also conjectured that, for any hard problem in the standard two-party communication model, the asymmetric version of the problem is hard in his model, provided not too much advice is given.

In this paper, we prove several surprising results about his model. We show that there exist Boolean functions requiring linear randomized communication complexity in the two-party model, for which the asymmetric versions in his model have deterministic protocols with exponentially smaller complexity. For set-disjointness, which also requires linear randomized communication complexity in the two-party model, we give a deterministic protocol for the asymmetric version in his model with a quadratic improvement in complexity. These results demonstrate that Pătraşcu's conjecture, as stated, is false. In addition, we show that the randomized and deterministic communication complexities of problems in his model differ by no more than a logarithmic multiplicative factor.

We also prove lower bounds in some restricted versions of this model for natural functions such as set-disjointness and inner product. All of our upper bounds conform to these restrictions. Moreover, a special case of one of these lower bounds implies a new proof of a strong lower bound on the tradeoff between the query time and the amortized update time of dynamic data-structures with non-adaptive query algorithms.

1 Introduction

In the cell probe model [25, 9, 34], the complexity of an algorithm is measured by the number of (fixed size) memory cells it accesses. Lower bounds in the cell probe model have been obtained for numerous static data structure problems, yielding lower bounds for these problems in the unit cost random access machine.

Obtaining lower bounds for dynamic data structures in the cell probe model has been a challenge. In 1989, Fredman and Saks [10] introduced the chronogram method and used it to prove an $\Omega(\log n/\log\log n)$ lower bound on the worst case time per operation for the partial sums problem.

[*] Tata Institute of Fundamental Research, email: arkadev.c@tifr.res.in

[†] York University, email: jeff@cs.yorku.ca

[‡] University of Toronto, email: faith@cs.toronto.edu

[§] University of Toronto, email: toni@cs.toronto.edu

In 1998, Alstrup, Husfeldt and Rauhe [1] got the same bound for the dynamic marked ancestor problem. Reductions from these problems to a variety of other dynamic data structure problems have also been obtained [1, 12, 13, 14]. In 2004, Pǎtraşcu and Demaine [30] introduced a beautiful information theoretic technique to prove $\Omega(\log n)$ lower bounds for the partial sums problem and dynamic connectivity in undirected graphs. Later, Pǎtraşcu [29] used a reduction from set-disjointness in an asymmetric two-party communication model to prove an $\Omega(\log n/(\log\log n)^2)$ lower bound for the dynamic marked ancestor problem. More recently, Larsen proved the first $\Omega(\log^2 n)$ lower bounds, for the dynamic range counting problem [21] and dynamic polynomial evaluation (over a large field) [22]. Despite these advances, it remains a longstanding open problem to prove polynomial (or even super-polylogarithmic) lower bounds for any dynamic data structure problem.

1.1 Pǎtraşcu's Conjecture

Pǎtraşcu [28] listed a diverse collection of natural dynamic data structure problems that are conjectured to require superpolylogarithmic time per operation, including determining the existence of paths in dynamic directed graphs and finding the length of shortest paths in dynamic undirected graphs. He proposed an exciting new approach for obtaining polynomial lower bounds for all of these problems using a new communication model that we call the $A \overset{B}{\to} (B \leftrightarrow C)$ model. It augments the standard two-party communication model between two players Bob and Charlie, by providing advice (given by Alice) to one of the players (Bob).

For any Boolean function $f : X \times Y \to \{0,1\}$, Pǎtraşcu defined an asymmetric communication problem $\mathrm{SEL}_f^{k\times 1} : \{1,\ldots,k\} \times X^k \times Y \to \{0,1\}$, where $\mathrm{SEL}_f^{k\times 1}(i, x_1, \ldots, x_k, y) = f(x_i, y)$. In the $A \overset{B}{\to} (B \leftrightarrow C)$ model, there are two players, Bob and Charlie, who, with advice from Alice, compute $\mathrm{SEL}_f^{k\times 1}(i, x_1, \ldots, x_k, y)$ as follows: Alice receives $x_1, \ldots, x_k$ and y, Bob receives y and i, and Charlie receives $x_1, \ldots, x_k$ and i. Alice first sends some advice privately to Bob and then remains silent. Thereafter, Bob and Charlie can communicate back and forth, alternating arbitrarily, until they have computed the output of the function. The last bit that is sent is the output of the protocol, which is supposed to be the value of the function. This can also be viewed as a restricted version of a three-player number-on-the-forehead problem, in which Alice has i on her forehead, Bob has $x_1, \ldots, x_k$ on his forehead, and Charlie has y on his forehead.

Pǎtraşcu presented simple reductions from the problem of computing $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, where DISJ denotes the set-disjointness problem, to many dynamic problems in the cell probe model. These reductions prove that, if $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ cannot be solved by a protocol in which Alice gives $o(ntw)$ bits of advice and Bob and Charlie communicate a total of $o(tw)$ bits, then the worst case time per operation of the dynamic problems is $\Omega(t)$ in the cell probe model with w bit words.

He conjectured that there exist positive constants $\delta < 1$ and $\gamma > 1+\delta$ such that $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ cannot be solved for $k \in \Theta(n^\gamma)$ if Alice gives $o(n^{1+\delta})$ bits of advice and Bob and Charlie communicate a total of $o(n^\delta)$ bits. If his conjecture is true, then all of the dynamic problems presented in [28] require $n^{\Omega(1)}$ time per operation in the cell probe model with $O(\log n)$ bit words. More generally, he stated the following conjecture, which does not specify whether the communication protocols involved are deterministic or randomized.

Conjecture 1 (Pǎtraşcu) *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be any function. Consider a protocol*

π for computing $\mathrm{SEL}_f^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. If Alice sends $o(k)$ bits, then the cost of communication between Bob and Charlie is $\Omega(c)$, where c is the 2-party communication complexity of f.

The intuition is that, if Alice sends $o(k)$ bits of advice, then, for many of the instances $f(x_1, y), \ldots, f(x_k, y)$, she is providing very little information. This suggests that, in the worst case, solving one of the these instances should be essentially as hard as computing f in the standard two-party model. Furthermore, the generality of this conjecture, namely that it makes no assumptions about the structure of f, invites the possibility of an information theoretic round elimination argument.

1.2 Refutations

To our surprise, this intuition is not correct. While it is true that Alice cannot provide much information about the x_i's, it turns out that she *can* provide a succinct message that will help Charlie learn y. This is the main intuition behind all of our upper bounds.

For example, it is easy to disprove Pătraşcu's conjecture for deterministic protocols by considering the equality function, EQ, where $\mathrm{EQ}(x, y) = 1$ if and only if $x = y$. It has a very simple deterministic protocol in which Alice sends Bob the minimum $j \in \{1, \ldots, k\}$ such that $y = x_j$. If there is no such j, she sends him 0. Bob forwards this message to Charlie, who can determine that the output should be 1 if and only if he receives $j \neq 0$ and $x_j = x_i$. Here, Alice teaches y to Charlie (via Bob) using a very short message.

We exploit this intuition to prove a much stronger result, using notions from learning theory and recent results about sign matrices. Specifically, we show that, even if a Boolean function f has large randomized complexity in the two-party model, $\mathrm{SEL}_f^{k\times 1}$ can have small deterministic complexity in the $A \overset{B}{\to} (B \leftrightarrow C)$ model.

Theorem 2 *There exists a Boolean function f with two-party* **randomized** *communication complexity $\Omega(n)$ such that $\mathrm{SEL}_f^{k\times 1}$ has a* **deterministic** *protocol in the $A \overset{B}{\to} (B \leftrightarrow C)$ model in which the total number of bits communicated is $O(\log^2 k)$.*

Note that when $k \in n^{O(1)}$, the total amount of communication is $O(\log^2 n)$.

Interestingly, we prove the upper bound using the harder side of Yao's min-max principle. Although it is standard to use the min-max principle for proving lower bounds, we are not aware of its application to prove upper bounds, especially for communication protocols.

A natural hope would be that Pătraşcu's conjecture is still true for certain specific Boolean functions with $\Omega(n)$ two-party randomized complexity, such as set-disjointness. Our next result shows that this is not the case for set-disjointness. We directly design a protocol for set-disjointness, in which Alice reveals a carefully chosen subset of y's bits so that, on the remaining bits, determining $\mathrm{DISJ}(x_i, y)$ is easy, for each $i \in \{1, \ldots, k\}$, because either x_i has few 1's or a large fraction of the positions of 1's in x_i are also positions of 1's in y.

Theorem 3 *There is a deterministic protocol for $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, in which Alice sends at most $\sqrt{n}\log k$ bits, Bob sends at most $1 + \sqrt{n}\log k$ bits, and Charlie sends at most $\sqrt{n}\log n$ bits.*

Note that when $k \in n^{O(1)}$, the total amount of communication is $O\big(\sqrt{n}\log n\big)$.

It is worth remarking that Theorem 3 does not eliminate the possibility of proving strong lower bounds for dynamic data structure problems via the $A \overset{B}{\to} (B \leftrightarrow C)$ model. To obtain polynomial lower bounds for the dynamic problems listed above, it suffices to prove that, for every protocol in which Alice sends $o(k)$ bits of advice, Bob and Charlie must communicate $\Omega(n^{\delta})$ bits to compute $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$, for some constant $0 < \delta \leq 1/2$. Theorem 2 and Theorem 3 show that such a lower bound argument has to crucially use the structure of the set-disjointness function.

We also show that the randomized and deterministic communication complexities of computing $\mathrm{SEL}^{k\times 1}_{f}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model do not differ by much. Furthermore, for any Boolean function f with two-party randomized communication complexity R, we show that $\mathrm{SEL}^{k\times 1}_{f}$ has deterministic communication complexity $O((R + \log n + \log k)\log k)$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. This immediately shows that problems which, in the 2-party model, have efficient randomized protocols, but are hard deterministically, give rise to easy asymmetric problems in the $A \overset{B}{\to} (B \leftrightarrow C)$ model.

1.3 Restricted Lower Bounds

Finally, we provide lower bounds in the $A \overset{B}{\to} (B \leftrightarrow C)$ model for some restricted classes of protocols, which include those protocols used for our upper bounds in Theorem 2 and Theorem 3. In those protocols, Alice sends far fewer bits of advice than she is allowed to. Moreover, after Alice's message is sent, Bob and Charlie engage in a very limited form of interaction. Our lower bounds show that each of these restrictions, by itself, does not allow improvements in our upper bounds. For analyzing protocols where Alice's advice is less than $\sqrt{n}$ bits, we convert the problem into a direct product problem with $\sqrt{n}$ instances. Then we obtain our lower bounds using recent strong direct product theorems. For analyzing restricted interactions between Bob and Charlie, we present an information theoretic argument. We show that, if there is a limited interaction protocol in which Bob and Charlie communicate few bits, then $x_1, \ldots, x_k$ can be compressed to substantially fewer than kn bits, which is impossible, in general.

While limited interaction protocols are a natural restriction of general ones, there is an additional motivation to study them. In particular, they are very related to restricted dynamic data structure algorithms. Consider Pătraşcu's three phase dynamic problem [28]. In the first phase, data gets inserted and the algorithm pre-processes this efficiently. The second phase consists of a series of updates which are each processed in amortized time t_u. The third phase gets a query and the algorithm outputs the answer in time t_q. Suppose that the third phase is *non-adaptive* in the sense that, for each query, there is a fixed set of cells that are probed, which does not depend on the results of the queries. All other phases have no restrictions. Can one prove lower bounds for such algorithms? After the preliminary version of our work [7] appeared, Brody and Larsen [6] proved strong lower bounds for such algorithms, which are *independent* of the pre-processing time allowed in the first phase of the algorithm. We show that these lower bounds follow from our lower bounds on restricted protocols. More precisely, we observe that every non-adaptive query algorithm yields a very restrictive, non-interactive protocol in the $A \overset{B}{\to} (B \leftrightarrow C)$ model: In these protocols, Bob sends no messages at all. Our lower bounds for less restricted protocols then imply their non-adaptive lower bounds.

1.4 Organization

In Section 2, we introduce notation and define necessary concepts from two-party communication complexity. In Section 3.1, we prove that randomization does not help for solving $\mathrm{SEL}_f^{k\times 1}$ in the $A \overset{B}{\rightarrow} (B \leftrightarrow C)$ model. Then, in Section 3.2, we present a function f with $\Omega(n)$ randomized two-party complexity such that $\mathrm{SEL}_f^{k\times 1}$ can be computed deterministically with only $\log^{O(1)} k$ bits of communication. In Section 3.3, we prove our upper bound for set-disjointness. In Section 4, we prove lower bounds in restricted settings and, in Section 5, we apply our lower bounds to prove strong polynomial lower bounds for nonadaptive dynamic data structure problems. We conclude in Section 6 with some open problems.

2 Preliminaries

Communication complexity was first studied for the two-party model in which the input is partitioned between two players, who compute a Boolean function of their inputs [36]. At the end of the computation, both players know the value of the function.

For any function $f : X \times Y \rightarrow \mathbb{Z}$, we use $D(f)$ to denote the *deterministic complexity* of f, which is the minimum over all deterministic protocols π that compute f correctly on all inputs, of the maximum number of bits communicated during any execution of the protocol. If $\mu : X \times Y \rightarrow [0, 1]$ is a probability distribution and $0 < \epsilon < 1$, we use $D_\mu^\epsilon(f)$ to denote the *ϵ-error distributional complexity* of *f for distribution μ*, which is the minimum over all deterministic protocols that compute a function g differing from f on a set of inputs with probability at most ϵ, of the maximum number of bits communicated during any execution of the protocol. Note that $D_\mu^\epsilon(f) \leq D(f) \leq n+1$ for any $f : X \times Y \rightarrow Z$, $\mu : X \times Y \rightarrow [0, 1]$, and $0 < \epsilon < 1$, since one player can send its input to the other player, who responds with the answer.

The computation of a randomized two-party protocol can be expressed as a function of the players' inputs, x and y, and a public (shared) sequence r of random bits that is provided to both players. A protocol π for f has *error probability* ϵ if

$$\max\{\Pr[\pi(x, y, r) \text{ does not compute } f(x, y)] \mid x \in X, y \in Y\} = \epsilon,$$

where the probability is taken over all choices of r. The *ϵ-randomized complexity* of f, which we denote by $R^\epsilon(f)$, is the minimum over all randomized protocols for f with error probability at most ϵ, of the maximum number of bits communicated during any execution of the protocol. Yao [35] gave the following relationship between randomized and distributional communication complexities. It is often called Yao's min-max principle.

Theorem 4 *For any function f and any $0 < \epsilon < 1$, $R^\epsilon(f) = max_\mu\{D_\mu^\epsilon(f)\}$.*

In fact, the proof of Theorem 4 [20] applies to any reasonable non-uniform model of computation with public coins.

There are simple Boolean functions that have very high deterministic complexity in the 2-party model, but have efficient 2-party randomized protocols. For example, consider the equality function, $\mathrm{EQ} : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$, where

$$\mathrm{EQ}(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y. \end{cases}$$

$D(\mathrm{EQ}) \in \Theta(n)$, but $R^{\epsilon}(\mathrm{EQ}) \in O(1)$ for any constant $0 < \epsilon < \frac{1}{2}$. Another example is the greater than function, $\mathrm{GT} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, defined by

$$\mathrm{GT}(x, y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{if } x \leq y, \end{cases}$$

where x and y are interpreted as n-bit integers. $D(\mathrm{GT}) \in \Theta(n)$, but $R^{\epsilon}(\mathrm{GT}) \in O(\log n)$ for any constant $0 < \epsilon < \frac{1}{2}$. [27]. Thus, in the context of two-player games, randomization can be substantially more powerful than determinism.

The set-disjointness problem on n-bit strings, denoted by $\mathrm{DISJ} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, is defined by

$$\mathrm{DISJ}(x, y) = \bigwedge_{i=1}^{n} (\overline{x[i]} \vee \overline{y[i]}).$$

In other words, if x and y are viewed as the characteristic vectors of two subsets of $\{1, \ldots, n\}$, then $\mathrm{DISJ}(x, y) = 1$ if and only if the two subsets are disjoint, i.e., for all $i \in \{1, \ldots, n\}$, either $x[i] = 0$ or $y[i] = 0$.

Babai, Frankl and Simon [2] were the first to prove a randomized lower bound of $\Omega(\sqrt{n})$ for set-disjointness. The lower bound was improved to $\Theta(n)$ by the celebrated result of Kalyanasundaram and Schnitger [15] and later simplified by Razborov [31]. Subsequently, Bar-Yossef et.al. [3] gave an elegant information theoretic proof of the bound, which has been very influential in the current development of information complexity.

Newman [26] proved that any randomized two-party communication protocol (with public randomness) in which each player has an input in $\{0,1\}^n$ can be simulated by a two-party protocol that uses $O(\log n)$ random bits. Implicit in Newman's proof is a more general result, which holds for any nonuniform model of computation, such as communication protocols, boolean circuits, decision trees and non-uniform Turing machines:

Theorem 5 *If there is a randomized computation for a function with domain U and error probability at most $\epsilon < 1/2$, then there is a randomized computation for that function with the same cost and error probability $O(\epsilon)$ that uses only $O(\log \log |U|)$ random bits.*

Proof: Let $F(u, r)$ denote the randomized computation of a function f with input u from domain U and the (infinite) public sequence r of random bits. Suppose F has error probability at most ϵ, i.e. for all inputs $u \in U$, the probability that $F(u, r)$ computes $f(u)$ is at least $1 - \epsilon$, where the probability is taken over the choices, r, for the public binary sequence.

We show that there exist $\delta \in O(\epsilon)$ and $t \in O(\log |U|)$ binary sequences $r_1, \ldots, r_t$ such that, for each input u, if we choose a sequence r at random from $r_1, \ldots, r_t$, then $F(u, r)$ computes $f(u)$ with probability at least $1 - \delta$. Suppose $r_1, \ldots, r_t$ are chosen independently at random from the space of binary sequences. For any input $u \in U$ and any $i \in \{1, \ldots, t\}$,

$$\Pr\left[F(u, r_i) \text{ computes } f(u)\right] \geq 1 - \epsilon,$$

so $\mathbb{E}\big[\#\{i \in \{1, \ldots, t\} \mid F(u, r_i) \text{ computes } f(u)\}\big] \geq (1 - \epsilon)t$. By the Chernoff bound [18], for all $0 < \delta' < 1$,

$$\Pr\left[\#\{i \in \{1, \ldots, t\} \mid F(u, r_i) \text{ computes } f(u)\} < (1 - \delta')(1 - \epsilon)t\right] < e^{-(1-\epsilon)t(\delta')^2/2}.$$

Let $\delta' = 1/2$, let $\delta = (1+\epsilon)/2 \in O(\epsilon)$, and let $t = 8(1-\epsilon)^{-1}\ln|U| \in O(\log|U|)$ since $\epsilon < \frac{1}{2}$. Then $(1-\delta')(1-\epsilon) = 1-\delta$ and $(1-\epsilon)t(\delta')^2/2 = \ln|U|$, so

$$\Pr\left[\#\{i \in \{1,\ldots,t\} \mid F(u,r_i) \text{ computes } f(u)\} < (1-\delta)t\right] < 1/|U|.$$

The union bound implies that

$$\Pr\left[\text{ exists } u \in U \text{ such that } \#\{i \in \{1,\ldots,t\} \mid F(u,r_i) \text{ computes } f(u)\} < (1-\delta)t\right] < 1.$$

Hence, there exist choices of $r_1,\ldots,r_t$ such that, for every input $u \in U$,

$$\#\{i \in \{1,\ldots,t\} \mid F(u,r_i) \text{ computes } f(u)\} \geq (1-\delta)t.$$

On any input $u \in U$, the computation F' chooses $i \in \{1,\ldots,t\}$ uniformly at random and performs the computation $F(u,r_i)$. Then

$$\Pr[F'(u,i) \text{ computes } f(u)] \geq 1-\delta,$$

so $F'(u,i)$ computes f with error probability $\delta \in O(\epsilon)$. ■

3 Upper Bounds in the $A \xrightarrow{B} (B \leftrightarrow C)$ model

For any protocol π in the $A \xrightarrow{B} (B \leftrightarrow C)$ model, we define $CC_{A\to B}(\pi)$ to be the worst case number of bits sent by Alice and $CC_{B\leftrightarrow C}(\pi)$ to be the worst case number of bits communicated between Bob and Charlie.

3.1 Alice can Derandomize

We begin by showing that every randomized protocol for $\mathrm{SEL}_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model can be efficiently derandomized.

Theorem 6 *Consider any Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. Let π be a randomized protocol for $SEL_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model with $CC_{A\to B}(\pi) = m$, $CC_{B\to C}(\pi) = \ell$, $CC_{C\to B}(\pi) = q$, and error probability at most $\frac{1}{2} - \epsilon$, for some constant $0 < \epsilon < \frac{1}{2}$. Then, there exists a deterministic protocol π' for $SEL_f^{k\times 1}$ such that $CC_{A\to B}(\pi') \in O((m + \log k + \log n)(\log k)/\epsilon^2)$, $CC_{B\to C}(\pi') \in O((\ell + \log k + \log n)(\log k)/\epsilon^2)$, and $CC_{C\to B}(\pi') \in O(q(\log k)/\epsilon^2)$.*

Proof: By Theorem 5, we may assume that π uses only $h \in O(\log k + \log n)$ random bits. Let $t \geq (1+2\epsilon)(\ln k)/\epsilon^2$ be an odd integer. Choose t strings $r_1,\ldots,r_t$ independently at random from $\{0,1\}^h$. Let $x \in \{0,1\}^{nk}$, $y \in \{0,1\}^n$, and $i \in \{1,\ldots,k\}$. Then, for each $j \in \{1,\ldots,t\}$, $\Pr[\pi(x,y,i,r_j) \text{ outputs } f(x_i,y)] \geq \frac{1}{2} + \epsilon$. Let $\delta = 1 - 1/(2\epsilon+1)$. Then $(1-\delta)(\frac{1}{2}+\epsilon) = \frac{1}{2}$, so by the Chernoff bound,

$$\Pr\left[\#\{j \in \{1,\ldots,t\} \mid \pi(x,y,i,r_j) \text{ does not output } f(x_i,y)\} < t/2\right] < e^{-(\frac{1}{2}+\epsilon)t\delta^2/2} \leq e^{-\ln k} = 1/k.$$

Hence, there is a nonzero probability that, for all $i \in \{1,\ldots,k\}$, $\pi(x,y,i,r_j)$ outputs $f(x_i,y)$ for the majority of $j \in \{1,\ldots,t\}$. Thus, given x and y, Alice can find a sequence of $t \in \Theta((\log k)/\epsilon^2)$ strings $r_1,\ldots,r_t \in \{0,1\}^h$ for which this is true. She sends these strings to Bob, together with the messages $a_1,\ldots,a_t$ she sends in $\pi(x,y,i,r_j)$ for all $j \in \{1,\ldots,t\}$. Bob forwards the strings $r_1,\ldots,r_t$ to Charlie. Then, for all $j \in \{1,\ldots,t\}$, Bob and Charlie run $\pi(x,y,i,r_j)$ with Alice's message a_j and take the output that is produced most often. ■

Any two-party protocol for computing a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is a protocol for computing $\mathrm{SEL}_f^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model in which Alice sends nothing. Thus, we immediately get the following corollary:

Corollary 7 *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be any Boolean function such that $R^\epsilon(f) \in (\log n)^{O(1)}$, for some constant $0 < \epsilon < 1/2$. If $k \in n^{O(1)}$, then there exists a deterministic protocol for $SEL_f^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model where Alice sends $O\big(\log^2 n\big)$ bits to Bob and Bob and Charlie communicate $(\log n)^{O(1)}$ bits.*

It follows from Corollary 7 that both $\mathrm{SEL}_{\mathrm{EQ}}^{k\times 1}$ and $\mathrm{SEL}_{\mathrm{GT}}^{k\times 1}$ have efficient deterministic protocols in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. These functions refute Pătraşcu's conjecture for deterministic protocols.

3.2 Alice as a Teacher

Next, we show that there exists a Boolean function f with very large randomized complexity in the two-party model, for which $\mathrm{SEL}_f^{k\times 1}$ has efficient deterministic protocols in the $A \overset{B}{\to} (B \leftrightarrow C)$ model.

We need some definitions from computational learning theory. For any set $\mathcal{S}$ of Boolean functions over $\{0,1\}^n$, we associate a Boolean matrix $M_{\mathcal{S}}$, whose rows are indexed by $\{0,1\}^n$ and whose columns are indexed by $\mathcal{S}$, such that $M_{\mathcal{S}}[x,f] = f(x)$. A randomized algorithm L is said to *learn* $\mathcal{S}$ with *confidence* δ and *accuracy* ϵ from m random examples drawn from a distribution μ on $\{0,1\}^n$ if, for each $f \in \mathcal{S}$ and for $x_1, \ldots, x_m \in \{0,1\}^n$ chosen independently from the distribution μ, given $(x_1, f(x_1)), \ldots, (x_m, f(x_m))$, L outputs a Boolean hypothesis function $h : \{0,1\}^n \to \{0,1\}$ that, with probability at least $1-\delta$, is ϵ-close to f, i.e. if x is chosen from μ, then $\Pr\big[h(x) \neq f(x)\big] \leq \epsilon$. The *Vapnik-Chervonenkis (VC) dimension*, $vc(M)$, of a matrix M is the largest number d such that M has a $d \times 2^d$ sub-matrix all of whose columns are distinct, i.e., each vector in $\{0,1\}^d$ appears exactly once as a column in the sub-matrix. The following result, known as the VC Theorem [16], shows the relevance of VC dimension to learning.

Theorem 8 *Let $\mathcal{S}$ be a set of Boolean functions over $\{0,1\}^n$ and let μ be an arbitrary distribution on $\{0,1\}^n$. Then there exists a randomized algorithm L that learns $\mathcal{S}$ with confidence δ and accuracy ϵ from m random examples drawn from μ, where*

$$m \in O\left(\frac{1}{\epsilon}\log\frac{1}{\delta} + \frac{vc\big(M_{\mathcal{S}}\big)}{\epsilon}\log\frac{1}{\epsilon}\right).$$

For any Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, let M_f denote the matrix, whose rows and columns are indexed by $\{0,1\}^n$, such that $M_f[x,y] = f(x,y)$. If, for each $y \in \{0,1\}^n$, we define the Boolean function $f_y : \{0,1\}^n \to \{0,1\}$ such that $f_y(x) = f(x,y)$ and we let $\mathcal{S} = \{f_y \mid y \in \{0,1\}^n\}$, then $M_{\mathcal{S}} = M_f$. Using an elegant argument, Kremer, Nisan and Ron [19] showed that, if M_f has small VC-dimension, then f has small distributional communication complexity under product distributions (i.e. under distributions that can be expressed as the product of two distributions over $\{0,1\}^n$). We exploit this connection to learning theory to prove the following result.

Theorem 9 *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a Boolean function and let $0 < \epsilon < \frac{1}{2}$ be a constant. Then, there exists a randomized protocol π for $SEL_f^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model with error probability at most ϵ such that*

$$CC_{A\to B}(\pi), CC_{B\leftrightarrow C}(\pi) \in O\left(\frac{1}{\epsilon} vc(M_f) \log \frac{1}{\epsilon} \log k\right).$$

Proof: Using Yao's min-max principle (Theorem 4) in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, our task reduces to showing that, for every distribution μ on $\{1,\ldots,k\} \times \left(\{0,1\}^n\right)^k \times \{0,1\}^n$, there exists a deterministic protocol π_μ for $\mathrm{SEL}_f^{k\times 1}$ with error probability at most ϵ and $\mathrm{CC}_{A\to B}(\pi_\mu), \mathrm{CC}_{B\leftrightarrow C}(\pi_\mu)$ having the desired bound. We do this by first constructing a randomized protocol that has the required error probability over its internal coin tosses and over μ. A standard averaging argument then yields the desired deterministic protocol.

Let $\mathcal{S}$ denote the set of functions $\{f_y \mid y \in \{0,1\}^n\}$. For any inputs $x = (x_1,\ldots,x_k) \in \{0,1\}^{nk}$ and $y \in \{0,1\}^n$, Alice can determine the conditional distribution $\mu_{x,y}$ induced on $\{1,\ldots,k\}$. By Theorem 8, there is a randomized algorithm L that learns the function $f_y \in \mathcal{S}$ with confidence and accuracy $\epsilon/2$ from m random examples drawn from $\mu_{x,y}$, where

$$m \in O\left(\frac{2}{\epsilon}\log\frac{2}{\epsilon} + \frac{2vc(M_{\mathcal{S}})}{\epsilon}\log\frac{2}{\epsilon}\right).$$

Alice draws m samples $i_1,\ldots,i_m$ from $\mu_{x,y}$ and sends Bob a message containing

$$\left(i_1, f(x_{i_1}, y)\right), \ldots, \left(i_m, f(x_{i_m}, y)\right).$$

This requires communicating at most $m(1+\log k)$ bits. Bob transmits this message to Charlie. In learning theoretic terms, Alice, the teacher, is trying to teach f_y to the learning algorithm Charlie. Charlie uses the randomized algorithm L to compute a hypothesis h consistent with Alice's m examples such that the probability $h(x_i) \neq f(x_i, y)$ is at most ϵ. Note that this probability is over the random coin tosses used by Alice to sample points and over the distribution $\mu_{x,y}$ for i. Finally, Charlie completes the protocol by sending $h(x_i)$ to Bob.

By a standard averaging argument, Alice's coin tosses can be fixed such that the resulting deterministic protocol has error probability at most ϵ for distribution μ. ■

The above theorem shows that if M_f has at most polylogarithmic VC-dimension, then $\mathrm{SEL}_f^{k\times 1}$ has very efficient protocols in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. Using earlier results of Ben-David et.al. [5] and Linial and Shraibman [24], Sherstov [33] showed (implicitly in the proof of Theorem 3.5) that there exists a function f with high randomized communication complexity in the two-party model such that M_f has low VC-dimension.

Theorem 10 *For any constant $0 < \epsilon < 1$, there are functions f such that M_f has VC-dimension $O(1)$ and $R^\epsilon(f) \in \Omega(n)$.*

Now, we have everything in place to prove our first main result.

Theorem 2. *There exists a Boolean function f with two-party* **randomized** *communication complexity $\Omega(n)$ such that, for $n \leq k \in 2^{(\log n)^{O(1)}}$, $SEL_f^{k\times 1}$ has a* **deterministic** *protocol in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, in which Alice sends Bob $O\left(\log^2 k\right)$ bits and then Bob and Charlie communicate a total of $O\left(\log^2 k\right)$ bits.*

Proof: By Theorem 10, there is a function f such that $R^{\epsilon}(f) \in \Omega(n)$ and $vc\big(M_f\big) = O(1)$. It follows from Theorem 9 that $\mathrm{SEL}_f^{k\times 1}$ has a randomized protocol π in the $A \overset{B}{\to} (B \leftrightarrow C)$ model in which Alice sends $O(\log k)$ bits of advice and Bob and Charlie communicate $O(\log k)$ bits. Finally, applying Theorem 6, we derandomize π to obtain a deterministic protocol π' such that $\mathrm{CC}_{A\to B}\big(\pi'\big) = O\big(\log^2 k\big)$ and $\mathrm{CC}_{B\leftrightarrow C}\big(\pi'\big) = O\big(\log^2 k\big)$. ■

This disproves Conjecture 1, even for randomized protocols.

3.3 An Upper Bound for Set-Disjointness

We construct a protocol for $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ with $o(n)$ communication complexity in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. Throughout the construction, it is helpful to view the inputs $x_1, \ldots, x_k$, and y as subsets of $\{1, \ldots, n\}$.

Theorem 3 *There is a deterministic protocol for* $\mathrm{SEL}_{\mathrm{DISJ}}^{k\times 1}$ *in the* $A \overset{B}{\to} (B \leftrightarrow C)$ *model, in which Alice sends at most* $\sqrt{n}\log k$ *bits, Bob sends at most* $1 + \sqrt{n}\log k$ *bits, and Charlie sends at most* $\sqrt{n}\log n$ *bits.*

Proof: Given $x_1, \ldots, x_k$, and y, Alice repeatedly picks a set from among $x_1, \ldots, x_k$ that is:

- disjoint from y and
- contains a least $\sqrt{n}$ elements that are not in the union of the sets she has already picked.

Then Alice send the indices of these sets to Bob. Note that each time Alice picks a set, the number of elements in the union of the sets she has picked increases by at least $\sqrt{n}$. Since these sets are all subsets of $\{1, \ldots, n\}$, Alice picks at most $\sqrt{n}$ sets. A set index can be represented using $\log k$ bits, so Alice sends at most $\sqrt{n}\log k$ bits to Bob.

Bob forwards the information he receives from Alice to Charlie. Charlie computes the union of these sets and removes them from x_i, since none of them are in y. Let x' denote the resulting set, so $x' \cap y = x_i \cap y$. If x' contains at least $\sqrt{n}$ elements, then x is not disjoint from y, since, otherwise, Alice would have picked more sets and sent more indices. In this case, Charlie sends 0 to Bob and the protocol terminates.

If x' contains fewer than $\sqrt{n}$ elements, then, Charlies send 1 to Bob, followed by each of the elements in x'. Since each element can be represented using $\log n$ bits, Charlie sends at most $\sqrt{n}\log n$ bits to Bob. In this case, Bob computes $x' \cap y = x_i \cap y$ and sends the answer to Charlie. Thus Bob sends at most $1 + \sqrt{n}\log k$ bits. ■

4 Lower Bounds in Restricted Models

An interesting fact is that our upper bounds do not use the full power of the $A \overset{B}{\to} (B \leftrightarrow C)$ model. First, Alice sends far fewer bits than she is allowed to. Second, Bob, the receiver of Alice's advice, is merely forwarding it to Charlie without processing it in any way. Third, the algorithms in Sections 3.2 and 3.3 have limited interaction between Bob and Charlie. We now discuss the limitations that these restrictions place on the power of the $A \overset{B}{\to} (B \leftrightarrow C)$ model. In Section 4.1, we prove our upper bound for set-disjointness cannot be substantially improved, unless we allow Alice to send more than $\sqrt{n}$ bits of advice, even if players interact arbitrarily. In Section 4.2, we complement this by showing the upper bound for set-disjointness cannot be improved if Bob and Charlie have limited interaction.

4.1 Lower Bounds via Strong Direct Product Theorems

For any Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, let $f^{(k)} : \{0,1\}^{nk} \times \{0,1\}^{nk} \to \{0,1\}^k$ denote the function such that, for all $x_1, \ldots, x_k, y_1, \ldots, y_k \in \{0,1\}^n$, $f^{(k)}(x_1, \ldots, x_k, y_1, \ldots, y_k) = (f(x_1, y_1), \ldots, f(x_k, y_k))$. Suppose that every c-bit communication protocol for f has probability of success $\sigma < 1$. Then a *strong direct product theorem* for f states that any ck-bit protocol for $f^{(k)}$ has success probability that is exponentially small in k.

There is a rich history of both positive and negative results for strong direct product theorems in complexity theory, including Yao's famous XOR Lemma (see for example [11]). Shaltiel [32] initiated the study of strong direct product theorems in communication complexity and proved a strong direct product theorem for functions where we have lower bounds via the discrepancy method over product distributions. This includes functions such as the inner product function. Lee, Shraibman, and Spalek [23] strengthened Shaltiel's result by proving a strong direct product theorem for functions that have lower bounds via the discrepancy method over *any* distribution. There is no known lower bound for set-disjointness via the discrepancy method, although a weaker form of a strong direct product theorem (with suboptimal parameters) was obtained by Beame, Pitassi, Segerlind and Wigderson [4]. Finally, Klauck [17] proved the following optimal strong direct product theorem for set-disjointness.

Theorem 11 *There exist constants $0 < \beta < 1$ and $\alpha > 0$ such that, for all $k \geq 1$, every randomized protocol which computes $\mathrm{DISJ}^{(k)} : \{0,1\}^{nk} \times \{0,1\}^{nk} \to \{0,1\}^k$ using at most βkn bits of communication has error probability greater than $1 - 2^{-\alpha k}$.*

Using this theorem, we obtain the following lower bound for asymmetric set-disjointness in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. A similar lower bound can also be obtained for any Boolean function that has a strong direct product theorem.

Theorem 12 *There exist constants $0 < \beta < 1$ and $\alpha > 0$ such that, in any deterministic protocol for $\mathrm{SEL}_{\mathrm{DISJ}}^{\sqrt{n} \times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, if Alice sends at most $\alpha\sqrt{n}$ bits, then Bob and Charlie must communicate at least $\beta\sqrt{n}$ bits.*

Proof: Let α and β be constants that satisfy Theorem 11 and let $k = \sqrt{n}$.

To obtain a contradiction, suppose that there is a deterministic protocol for $\mathrm{SEL}_{\mathrm{DISJ}}^{k \times 1}$, where Alice sends αk bits of advice to Bob, and then Bob and Charlie communicate $c < \beta k$ bits. Using this protocol, for every distribution μ' on $\{0,1\}^{k \times k} \times \{0,1\}^{k \times k}$, we construct a deterministic ck-bit protocol for $\mathrm{DISJ}^{(k)} : \{0,1\}^{k \times k} \times \{0,1\}^{k \times k} \to \{0,1\}^k$ with error probability at most $\epsilon = 1 - 2^{-\alpha k}$, i.e., $D_{\mu'}^{\epsilon}(\mathrm{DISJ}^{(k)}) \leq ck$. Then Yao's min-max principle (Theorem 4) implies that $R^{\epsilon}(\mathrm{DISJ}^{(k)}) \leq ck$. This contradicts Theorem 11, the direct product theorem for set-disjointness.

Consider any distribution $\mu' : \{0,1\}^{k \times k} \times \{0,1\}^{k \times k} \to [0,1]$. Given inputs $x'_1, \ldots, x'_k, y'_1, \ldots, y'_k \in \{0,1\}^k$, we create inputs $x_1, \ldots, x_k, y \in \{0,1\}^n$ for $\mathrm{SEL}_{\mathrm{DISJ}}^{k \times 1}$ as follows: $y = y'_1 \cdots y'_k$ and, for each $i \in \{1, \ldots, k\}$, $x_i = 0^{(i-1)k} x'_i 0^{(n-i)k}$. In particular, x_i is all 0's except for its i'th block of k bits, which is x'_i. Let μ be the resulting distribution on $\{0,1\}^{nk} \times \{0,1\}^n$.

Alice's αk-bit message partitions the space $\{0,1\}^{nk} \times \{0,1\}^n$ into $2^{\alpha k}$ equivalence classes. Let C be an equivalence class with maximal weight under distribution μ. Then $\mu(C) \geq 2^{-\alpha k}$. Since the $A \overset{B}{\to} (B \leftrightarrow C)$ protocol is deterministic, it answers correctly for every input $((x_1, \ldots, x_k), y) \in C$ and $i \in \{1, \ldots, k\}$.

Consider the following two-party protocol for $\mathrm{DISJ}^{(k)}$: Given $x'_1, \ldots, x'_k, y'_1, \ldots, y'_k \in \{0,1\}^k$, Bob and Charlie compute the answers to $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}(i, (x_1, \ldots, x_k), y)$, for $i = 1, \ldots, k$, pretending, each time, that Alice sent the message for equivalence class C. They concatenate these k one-bit answers to get a k-bit answer to $\mathrm{DISJ}^{(k)}(x'_1, \ldots, x'_k, y'_1, \ldots, y'_k)$. Since Bob and Charlie communicate at most c bits to compute each one-bit answer, they communicate at most $ck < \beta k^2$ bits in total.

Let $C' \subseteq \{0,1\}^{k\times k} \times \{0,1\}^{k\times k}$ consist of all inputs $(x'_1, \ldots, x'_k, y'_1, \ldots, y'_k)$ from which inputs $((x_1, \ldots, x_k), y)$ in C are produced. Then the two-party protocol correctly computes $\mathrm{DISJ}^{(k)}$ for all inputs in C' and $\mu'(C') \geq 2^{-\alpha k}$. Thus $D^{\epsilon}_{\mu'}(\mathrm{DISJ}^{(k)}) \leq ck$. ■

This lower bound matches our upper bound to within factors of $\log n$ and $\log k$.

4.2 Lower Bounds via Compression

In all our upper bounds, there is limited interaction between Bob and Charlie. We say that π is a 1.5 round (m, ℓ, q)-protocol if it proceeds in the following way: First, as usual, Alice sends m bits $A = A(X, y)$ to Bob. Then there are two rounds of communication between Bob and Charlie. In the first round, Bob communicates ℓ bits $B = B_1(y, A)$ to Charlie that do not depend on i. (For example, Bob could forward ℓ bits of Alice's message to Charlie.) In the second round, Charlie communicates q bits $C(x_1, \ldots, x_k, i, B)$ back to Bob. Finally, using his knowledge of i, Bob computes the answer $B_2(y, i, A, C) = f(X_i, y)$. Note that there is no restriction on Alice's advice. The crucial restriction, beyond the fact that there are only two rounds of communication after the advice, is that Bob's communication to Charlie is independent of i. We say that this is only half a round of communication.

Interestingly, 1.5 round protocols have non-trivial power. The proof of Theorem 2, which refutes Pătraşcu's conjecture, employs a 1.5 round $(O((\log n)^2, O((\log n)^2, 1)$-protocol. For set-disjointness, Theorem 3 gives a 1.5 round $\big(\sqrt{n}\log k,\ \sqrt{n}\log k,\ \sqrt{n}\log n\big)$-protocol. It is fun to verify that functions like equality and greater-than can all be solved cheaply, without even using the 0.5 round communication from Bob to Charlie, i.e. they both have deterministic $(O(\log n),\ 0,\ O(\log n))$-protocols.

In this section, we show the following limitations of 1.5 round protocols.

Theorem 13 *Let $k/\log k \in \omega(n)$ and $m \geq n$. For $1 \leq q$ and $1 \leq \ell \leq n$, every 1.5 round (m, ℓ, q)-protocol for computing $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$ has $\ell\big(\frac{61nm}{k} + q\big) \geq 0.008n$. If the protocol is randomized with error probability at most $\frac{1}{2} - \epsilon$, then it has $\big(\ell + \log k\big) \cdot \big(\frac{61nm}{k} + q\big) \in \Omega\big(\frac{\epsilon^4}{\log^2 k} n\big)$.*

Note that $\ell\big(\frac{61nm}{k} + q\big) \geq 0.008n$ implies that either $m \geq k/61n$ or $\ell \cdot (q{+}1) \geq 0.008n$. Since every 1.5 round $(m, 0, q_0)$-protocol immediately gives a 1.5 round $(m, 1, q_0)$-protocol, it follows that every 1.5 round $(m, 0, q_0)$-protocol for computing $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$ has $\frac{61nm}{k} + q \geq 0.008n$, so either $m \in \Omega(k)$ or $q \in \Omega(n)$. If the protocol is randomized with error probability at most $\frac{1}{2} - \epsilon$, then either $m \in \Omega\big(\frac{\epsilon^4}{\log^2 k} k\big)$ or $q \in \Omega\big(\frac{\epsilon^4}{\log^2 k} n\big)$. The upper bound in Theorem 3 shows that Theorem 13 is tight to within logarithmic factors, when $k = n^{2+\delta}$ for any constant $\delta > 0$. The lower bound in Theorem 12 is incomparable, since it restricts the amount of advice Alice can send.

Our next lower bound is for the well known inner-product function, $\mathrm{IP}(x, y) = \sum_{i=1}^{n} x_i y_i \bmod 2$. The inner product function is one of the hardest functions in the standard two-party communication model. For example, Chor and Goldreich [8, 20] showed that, even for protocols with an inverse-subexponential advantage over random guessing, inner product requires $\Omega(n)$ bits of communication in the two party model. It is thus also a natural target for proving lower bounds on the cost of 1.5 round protocols in the $A \overset{B}{\to} (B \leftrightarrow C)$ model.

Theorem 14 *Let $k \in [\omega(n), 2^{o(\sqrt{n})}]$. Consider any 1.5 round (m, ℓ, q)-protocol for computing $\mathrm{SEL}_{\mathrm{IP}}^{k\times 1}$. If Alice communicates $m \leq (1-\delta)k$ bits for some $\delta > 0$, then $q + \ell \geq (\delta - o(1))n$. If the protocol is randomized with error probability at most $\frac{1}{2} - \epsilon$, then either $m \in \Omega\big(\frac{\epsilon^2}{\log k}k\big)$ or $q + \ell \in \Omega\big(\frac{\epsilon^2}{\log k}n\big)$.*

Note that, if Alice communicates k bits, she can send the answer for each $i \in \{1, \ldots, k\}$.

Our lower bound for inner-product, $\ell + q \in \Omega(n)$, is stronger than our lower bound for set-disjointness, $\ell q \in \Omega(n)$. Recall that ℓ and q are the number of bits communicated by Bob and Charlie, respectively. The naive algorithm in which Bob sends y to Charlie has $\ell + q = n$, which matches our lower bound for inner product. The algorithm for set-disjointness in Section 3.3 has $\ell q \in O(n \log k \log n)$, which is only logarithmic factors more than our lower bound.

The main idea for proving both of these theorems is to find an encoding of $x_1, \ldots, x_k$ using 1.5 round protocols. If the cost of the protocol is small, our encoding *compresses* kn bits of information to fewer bits. However, this is impossible, since $x_1, \ldots, x_k$ has entropy kn, We begin by proving Theorem 14, since compression can be done cleanly for the inner-product function. Implementing compression for set-disjointness is more involved.

Proof of Theorem 14: Assume that π is a deterministic 1.5 round (m, ℓ, q)-protocol for computing $\mathrm{SEL}_{\mathrm{IP}}^{k\times 1}$, in which Alice communicates $m \leq (1-\delta)k$ bits. Our goal is to give a scheme for encoding $x_1, \ldots, x_k$, where each x_i is chosen uniformly from $\{0,1\}^n$.

Fix $x_1, \ldots, x_k$. Because Alice does not know i and Bob's message $B = B_1(y, A(x_1, \ldots, x_k, y))$ to Charlie cannot depend on i, this message depends only on y. Hence, by averaging, there exists a message B_{fixed} that Bob sends for at least $2^{n-\ell}$ many y's. Thus, there exists a set, $\mathcal{Y}$, of $n - \ell$ many linearly independent vectors such that Bob sends B_{fixed} on each of them. Let $\mathcal{Y}'$ be a set of ℓ additional linearly independent vectors such that $\mathcal{Y} \cup \mathcal{Y}'$ forms a basis of the vector space $\{0,1\}^n$. For each partial basis $\mathcal{Y}$, we choose $\mathcal{Y}'$ in some fixed way.

Our encoding of $x_1, \ldots, x_k$ contains the following:
(a) the set $\mathcal{Y}$, which can be represented using $(n-\ell) \cdot n$ bits;
(b) Alice's message $A(x_1, \ldots, x_k, y)$ for each $y \in \mathcal{Y}$, which can be represented using $(n-\ell) \cdot m$ bits;
(c) for each index $i \in \{1, \ldots, k\}$, the q bit message $C(x_1, \ldots, x_k, B_{fixed}, i)$ sent from Charlie to Bob; and
(d) extra information E consisting of the inner product of each x_i with each $y' \in \mathcal{Y}'$, which can be represented using $k \cdot \ell$ bits.
A key point is that Charlie's message does not depend on $y \in \mathcal{Y}$. This is because Charlie does not see y and, for all $y \in \mathcal{Y}$, Charlie receives the same message, B_{fixed}, from Bob.

Given any such encoding of $x_1, \ldots, x_k$, decoding can be done as follows: First, simulate Bob in protocol π, for each $i \in \{1, \ldots, k\}$ and each $y \in \mathcal{Y}$. This is possible because y, i, Alice's message $A(x_1, \ldots, x_k, y)$, Bob's message B_{fixed}, and Charlie's message $C(x_1, \ldots, x_k, B_{fixed}, i)$ are all known. Because π is a correct protocol for $\mathrm{SEL}_{\mathrm{IP}}^{k\times 1}$, the output of the protocol is $\mathrm{IP}(x_i, y)$. From $\{\mathrm{IP}(x_i, y) \mid i = 1, \ldots, k\}$ and the inner products in E, $x_1, \ldots, x_k$ can be obtained by solving a system of linear equations of full rank. Because no encoding of $x_1, \ldots, x_k$ can use less than nk bits, it follows that $(n-\ell) \cdot n + (n-\ell) \cdot m + qk + \ell k \geq nk$. Thus, $q + \ell \geq n - \frac{(n-\ell)(n+m)}{k}$. By assumption,

$m \leq (1-\delta)k$. Therefore $q + \ell \geq n - (n-\ell)(\frac{n}{k} + 1 - \delta) \geq (\delta - \frac{n}{k})n$, which is in $(\delta - o(1))n$, since $k \in \omega(n)$.

We now extend the result to handle randomized protocols. Theorem 6 converts a randomized (m_r, ℓ_r, q_r)-protocol into a deterministic (m_d, ℓ_d, q_d)-protocol, with $m_d \in O((m_r + \log k + \log n)(\log k)/\epsilon^2)$, $\ell_d \in O((\ell_r + \log k + \log n)(\log k)/\epsilon^2)$, and $q_d \in O(q_r(\log k)/\epsilon^2)$. Setting $\delta = 0.5$ gives that $m_d \geq 0.5k$ or $q_d + \ell_d \geq (0.5 - o(1))n$. In the first case, $m_r \in \Omega\big(\frac{\epsilon^2}{\log k} m_d - \log k - \log n\big)$, which is $\Omega\big(\frac{\epsilon^2}{\log k} k\big)$, since $k \in \omega(n)$. In the second case, $q_r + \ell_r \in \Omega\big(\frac{\epsilon^2}{\log k}(q_d + \ell_d) - \log k - \log n\big)$, which is $\Omega\big(\frac{\epsilon^2}{\log k} n\big)$, since $k \in \omega(n)$ and $(\log k)^2 \in o(n)$. ■

This proof can be easily adapted to prove a similar lower bound for the indexing function, $\text{IN} : \{0,1\}^n \times \{1, \ldots, n\} \to \{0,1\}$, where $\text{IN}\big(x, y\big) = x_y$. This function outputs the y'th bit of string x.

Theorem 15 *Consider any $(m, 0, q)$ protocol for computing $\text{SEL}_{\text{IN}}^{k\times 1}$. If $m \leq (1-\delta)k$, for some $\delta > 0$, then $q \geq \delta n$. If the protocol is randomized with error probability at most $\frac{1}{2} - \epsilon$, then either $m \in \Omega\big(\frac{\epsilon^2}{\log k} k\big)$ or $q \in \Omega\big(\frac{\epsilon^2}{\log k} n\big)$.*

Proof: The indexing function can be treated as a Boolean function with domain $\{0,1\}^n \times \{0,1\}^n$, with the promise that y is an n-bit string with Hamming weight 1. In this case, $\text{IN}\big(x, y\big)$ is the inner product of x and y. Then the proof is a simple specialization of the proof of Theorem 14. Since $\ell = 0$, Bob sends the same message (which consists of no bits) for every string $y \in \{0,1\}^n$ with Hamming weight 1. These n strings are linearly independent. Let $\mathcal{Y}$ denote this set of strings. The encoding of $x_1, \ldots, x_k$ is the same, except for (a) and (e), which are not needed, since $\mathcal{Y}$ is fixed and $\mathcal{Y}'$ is empty. It follows that $nm + qk \geq nk$, so $m \leq (1-\delta)k$ implies that $q \geq n - nm/k \geq n - (1-\delta)n = \delta n$. The remainder of the proof, including the extension to randomized protocols, is the same as in the proof of Theorem 14. ■

The inner-product of x_i with any known non-zero vector y provides 1 bit of information about x_i. However, the fact that $\text{DISJ}(x_i, y) = 0$ does not provide much information about x_i. This is the main source of complication when trying to apply the same approach to prove a lower bound for set-disjointness.

On the other hand, the fact that $\text{DISJ}(x_i, y) = 1$ provides a lot of information about x_i: all indices at which y is 1 are indices at which x_i is 0. Therefore, to encode $x_1, \ldots, x_k$ efficiently, we would like to choose a convenient set $\mathcal{Y}$ of y's such that $\text{DISJ}(x_i, y) = 1$ for many $i \in \{1, \ldots, k\}$. Unfortunately, if we choose vectors x and y uniformly at random from $\{0,1\}^n$, then $\text{DISJ}(x, y) = 0$ with very high probability. Hence, we have to work with a restricted set of vectors. This makes it delicate to find the set $\mathcal{Y}$.

Let Γ_x consist of the vectors in $\{0,1\}^n$ with Hamming weight σ_x and let Γ_y consist of the vectors in $\{0,1\}^n$ with Hamming weight σ_y. We will consider only $x_1, \ldots, x_k \in \Gamma_x$ and $y \in \Gamma_y$.

The following fact will be used in the proof of Claim 20 to show that, when $\sigma_x \cdot \sigma_y$ is appropriately set, 85% of the vectors in Γ_y do not intersect with any given vector in Γ_x.

Fact 16 *Suppose $\sigma_x, \sigma_y \leq n/4$. For each $x \in \Gamma_x$, if y is chosen at random from Γ_y, then $\Pr_y\big[\text{DISJ}(x, y) = 1\big] \geq exp\big(-\frac{4\sigma_x\sigma_y}{n}\big)$.*

Proof: Fix $x \in \Gamma_x$. If y is chosen at random from Γ_y, the probability that x and y are disjoint is

$$\frac{\binom{n-\sigma_x}{\sigma_y}}{\binom{n}{\sigma_y}} = \left(1 - \frac{\sigma_x}{n}\right) \cdot \left(1 - \frac{\sigma_x}{n-1}\right) \cdots \left(1 - \frac{\sigma_x}{n-\sigma_y+1}\right) \geq \left(1 - \frac{\sigma_x}{n-\sigma_y+1}\right)^{\sigma_y} > \left(1 - \frac{4\sigma_x}{3n}\right)^{\sigma_y},$$

since $\sigma_y \leq n/4$. Now $1 - t \geq \exp\big(-3t\big)$ for $0 \leq t \leq 1/3$ and $4\sigma_x/3n \leq 1/3$, since $\sigma_x \leq n/4$. Therefore $\big(1 - \frac{4\sigma_x}{3n}\big)^{\sigma_y} \geq (exp(-\frac{4\sigma_x}{n})^{\sigma_y} = exp(-\frac{4\sigma_x\sigma_y}{n})$. ■

Let $Cover(x_i, y) \subseteq \{1, \ldots, n\}$ denote the indices of x_i that one learns are zero from learning $\mathrm{DISJ}(x_i, y)$. If $\mathrm{DISJ}(x_i, y) = 1$, then $Cover(x_i, y) = \{j \,|\, y_j = 1\}$ and, if $\mathrm{DISJ}(x_i, y) = 0$, then $Cover(x_i, y) = \emptyset$. For any $\mathcal{Y} \subseteq \Gamma_y$, let $Cover((x_1, \ldots, x_k), \mathcal{Y}) = \{(i, j) \mid \text{there exists } y \in \mathcal{Y} \text{ such that } \mathrm{DISJ}(x_i, y) = 1 \text{ and } y_j = 1\} = \cup_{i \in \{1,\ldots,k\}} \cup_{y \in \mathcal{Y}} \{i\} \times Cover(x_i, y)$.

Next, we present the main lemma that enables $x_1, \ldots, x_k$ to be encoded efficiently.

Lemma 17 *Consider any deterministic 1.5 round (m, ℓ, q)-protocol, where $1 \leq \ell \leq n/20$. Let $\sigma_y = 5\ell$, let $\sigma_x = 0.008n/\ell$, and fix $x_1, \ldots, x_k \in \Gamma_x$. Then there exists a message B_{fixed} and a set $\mathcal{Y} \subseteq \Gamma_y$ of size at most $30n$ such that, for each $y \in \mathcal{Y}$, Bob sends B_{fixed} to Charlie and $|Cover((x_1, \ldots, x_k), \mathcal{Y})| \geq \frac{1}{2}nk$.*

Proving this lemma needs technical work. Before we do that, let us see how the lower bound for disjointness follows from Lemma 17 via compression. We will need also the following simple fact:

Fact 18 *Let $v_1, \ldots, v_k, \sigma$ be positive integers such that $v_1 + \cdots + v_k \leq r$ and $\sigma \leq \min\{v_i \mid 1 \leq i \leq k\}$. Then,*

$$\prod_{i=1}^{k} \binom{v_i}{\sigma} \leq \binom{\lceil r/k \rceil}{\sigma}^k.$$

Proof: We will make use of the inequality of arithmetic and geometric means: If $a_1, \ldots, a_k$ are non-negative real numbers, then

$$a_1 a_2 \cdots a_k \leq \left(\frac{a_1 + \cdots + a_k}{k}\right)^k.$$

Let j be any integer such that $0 \leq j < \sigma$ and let $a_i = v_i - j$ for all $i \in \{1, \ldots, k\}$. Then,

$$\prod_{i=1}^{k} (v_i - j) \leq \left(\frac{r}{k} - j\right)^k \leq (\lceil r/k \rceil - j)^k,$$

so, by the definition of binomial coefficients,

$$\prod_{i=1}^{k} \binom{v_i}{\sigma} = \frac{1}{(\sigma!)^k} \prod_{i=1}^{k} \prod_{j=0}^{\sigma-1} (v_i - j) \leq \frac{1}{(\sigma!)^k} \left(\prod_{j=0}^{\sigma-1} (\lceil r/k \rceil - j)\right)^k = \binom{\lceil r/k \rceil}{\sigma}^k.$$

■

Proof of Theorem 13: Let $\sigma_y = 5\ell$ and $\sigma_x = 0.008n/\ell$. Fix any 1.5 round deterministic (m, ℓ, q)-protocol π for $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$ and any $x = x_1, \ldots, x_k \in {\Gamma_x}^k$. If $\ell > n/20$, then $\ell\big(\frac{61nm}{k} + q\big) \geq \ell > 0.05n \geq 0.008n$. Thus, we may assume that $1 \leq \ell \leq n/20$. Then it is easy to verify that $\sigma_x, \sigma_y \leq n/4$. Let B_{fixed} and $\mathcal{Y}$ be the message and subset of Γ_y guaranteed by Lemma 17.

Our encoding of $x_1, \ldots, x_k$ contains the following:
(a) the set $\mathcal{Y}$, which can be represented using $30n \cdot n$ bits;
(b) Bob's fixed message B_{fixed}, which can be represented using $\ell = 0.2\sigma_y$ bits;
(c) Alice's message $A(x_1, \ldots, x_k, y)$ for each $y \in \mathcal{Y}$, which can be represented using $30n \cdot m$ bits;
(d) for each $i \in \{1, \ldots, k\}$, the q bit message $C(x_1, \ldots, x_k, B_{fixed}, i)$ sent from Charlie to Bob; and
(e) the remaining information E about $x_1, \ldots, x_k$ that is not learned from $\mathit{Cover}((x_1, \ldots, x_k), \mathcal{Y})$.

For $i \in \{1, \ldots, k\}$, let $S_i = \{j : (i, j) \notin \mathit{Cover}((x_1, \ldots, x_k), \mathcal{Y})$. Observe that the indices for which x_i has value one lie in S_i. Let $v_i = |S_i|$. Thus, the number of possibilities for the locations of the ones of $x_1, \ldots, x_k$ is $\binom{v_1}{\sigma_x} \cdot \binom{v_2}{\sigma_x} \cdots \binom{v_k}{\sigma_x}$. As $|\mathit{Cover}((x_1, \ldots, x_k), \mathcal{Y})| > 0.5nk$, we have $v_1 + \cdots + v_k < 0.5nk$. Invoking Fact 18, information E can be transmitted in at most $\log \binom{0.5n}{\sigma_x}^k$ bits.

Given any such encoding of $x_1, \ldots, x_k$, decoding can be done as follows: First, simulate Bob in protocol π, for each $i \in \{1, \ldots, k\}$ and each $y \in \mathcal{Y}$. This is possible because y, i, Alice's message $A(x_1, \ldots, x_k, y)$, Bob's message B_{fixed}, and Charlie's message $C(x_1, \ldots, x_k, B_{fixed}, i)$ are all known. Because π is a correct protocol for $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$, the output of the protocol is $\mathrm{DISJ}(x_i, y)$. From this, compute the indices in $\mathit{Cover}((x_1, \ldots, x_k), \mathcal{Y})$ where $x_1, \ldots, x_k$ has zeroes. By definition, E communicates the remaining information about $x_1, \ldots, x_k$, so it is possible to decode correctly.

Because no encoding of $x_1, \ldots, x_k$ can use less than its entropy $H(x_1, \ldots, x_k)$, we have:

$$30n \cdot n + \ell + 30n \cdot m + qk + H(E) \geq H(x_1, \ldots, x_k).$$

Note that $H(x_1, \ldots, x_k) = k \cdot \log \binom{n}{\sigma_x}$. Thus,

$$H(x_1, \ldots, x_k) - H(E) \geq \log\left(\frac{\binom{n}{\sigma_x}^k}{\binom{0.5n}{\sigma_x}^k}\right).$$

Using the observation that $\binom{n}{\sigma_x}/\binom{0.5n}{\sigma_x} \geq 2^{\sigma_x}$, we obtain $H(x_1, \ldots, x_k) - H(E) \geq \sigma_x k$. Thus we have

$$30n \cdot n + \ell + 30n \cdot m + qk \geq \sigma_x k.$$

Then, using the fact that $\ell \leq n \leq nm$, we have

$$\frac{30n(n+m) + nm}{k} + q \geq \sigma_x.$$

Since $m \geq n$, $\ell = 0.2\sigma_y$, and $\sigma_x \sigma_y = 0.04n$, it follows that

$$\ell\big(\frac{61nm}{k} + q\big) \geq \ell\big(\frac{30n(n+m) + nm}{k} + q\big) \geq \ell\sigma_x = 0.2\sigma_x\sigma_y = 0.008n.$$

As in the proof of Theorem 14, Theorem 6 can be used to extend the lower bound to randomized protocols, giving $(\ell + \log k + \log n)\left(\frac{61n}{k}(m + \log k + \log n) + q\right) \in \Omega(\frac{\epsilon^4}{\log^2 k} n)$. Since $k/\log k \in \omega(n)$, it follows that $\log k + \log n \in \Theta(\log k)$ and $\frac{61n}{k}(\log k + \log n) \in o(1)$. But $q \geq 1$, so $\frac{61n}{k}(\log k + \log n) + q \in \Theta(q)$. Hence $(\ell + \log k)\left(\frac{61nm}{k} + q\right) \in \Omega(\frac{\epsilon^4}{\log^2 k} n)$. ■

All that remains is to prove the existence of the set, $\mathcal{Y}(x_1, \ldots, x_k)$, promised by Lemma 17, for each $x_1, \ldots, x_k$. We will do so in two stages, using the probabilistic method. First, we will construct an intermediate set, $\mathcal{Y}_0(x_1, \ldots, x_k)$, with some nice properties. This will allow us to obtain our final desired set by picking elements from $\mathcal{Y}_0$ at random.

Lemma 19 *Consider any deterministic 1.5 round (m, ℓ, q)-protocol, where $1 \le \ell \le n/20$. Let $\sigma_y = 5\ell$ and $\sigma_x = 0.008n/\ell$. Then, for each $x_1, \ldots, x_k$, there exists a set $\mathcal{Y}_0(x_1, \ldots, x_k)$ such that:*

- *Bob sends the same message for each $y \in \mathcal{Y}_0(x_1, \ldots, x_k)$,*
- *$|\mathcal{Y}_0(x_1, \ldots, x_k)| \ge 10 \cdot \big(0.8\big)^{\sigma_y} \cdot |\Gamma_y|$, and*
- *$|\{i \in \{1, \ldots, k\} \mid \Pr_{y \in \mathcal{Y}_0}\big[DISJ(x_i, y) = 1\big] > 0.2\}| > 0.7k.$*

Proof: Fix $x_1, \ldots, x_k$. Choose B_{fixed} at random, where the probability of choosing each message is proportional to the number of y's for which Bob sends the message. Let $\mathcal{Y}_0$ denote the set of messages $y \in \Gamma_y$ for which Bob sends B_{fixed}. Bob sends at most 2^ℓ different messages. Hence, $\Pr_{B_{fixed}}\big[|\mathcal{Y}_0| < \frac{1}{4}2^{-\ell}|\Gamma_y|\big] < 1/4$. It follows that, for ℓ sufficiently large and, hence, for $\sigma_y = 5\ell$ sufficiently large, $\frac{1}{4}2^{-\ell}|\Gamma_y| > \frac{1}{4}(0.87)^{\sigma_y}|\Gamma_y| > 10 \cdot (0.8)^{\sigma_y}|\Gamma_y|$. Therefore, $\Pr_{B_{fixed}}\big[|\mathcal{Y}_0| < 10 \cdot (0.8)^{\sigma_y}|\Gamma_y|\big] < 1/4$.

The following claim shows that, for a typical random message B_{fixed}, for most messages $y \in \mathcal{Y}_0$ and for most $i \in \{1, \ldots, k\}$, the sets represented by y and x_i are disjoint.

Claim 20 $\mathbb{E}_{B_{fixed}}\Bigg[\Big|\{i \in \{1, \ldots, k\} \mid \Pr_{y \in \mathcal{Y}_0}[DISJ(x_i, y) = 1] \le 0.2\}\Big|\Bigg] \le 0.2k.$

To prove the claim, consider any $i \in \{1, \ldots, k\}$. Let D_i be the event that $\Pr_{y \in \mathcal{Y}_0}\big[\mathrm{DISJ}(x_i, y) = 1\big] \le 0.2$. Let $a = \Pr[D_i]$. We will bound a from above by computing $\Pr_{y \in \Gamma_y}\big[\mathrm{DISJ}(x_i, y) = 1\big]$ in two ways. First, note that choosing y at random from Γ_y is the same as first choosing B_{fixed} at random and then choosing $y \in \mathcal{Y}_0$ at random. Hence,

$$\begin{aligned}&\Pr_{y \in \Gamma_y}\big[\mathrm{DISJ}(x_i, y) = 1\big] \\ &= \Pr[D_i] \times \Pr_{y \in \mathcal{Y}_0}[\mathrm{DISJ}(x_i, y) = 1 \mid D_i] + \Pr[\neg D_i] \times \Pr_{y \in \mathcal{Y}_0}[\mathrm{DISJ}(x_i, y) = 1 \mid \neg D_i] \\ &\le a \times 0.2 + (1 - a) \times 1 = 1 - 0.8a.\end{aligned}$$

Since $\sigma_x, \sigma_y \le n/4$ and $\sigma_x \cdot \sigma_y = 0.04n$, Fact 16 implies that $\Pr_{y \in \Gamma_y}\big[\mathrm{DISJ}(x_i, y) = 1\big] \ge e^{-.16} > 0.85$. Hence $1 - 0.8a \ge 0.85$, which implies that $a < 0.1875 < 0.2$. This is true for all $i \in \{1, \ldots, k\}$. The claim now follows from the linearity of expectation.

Applying Markov's inequality to this claim gives $\Pr_{B_{fixed}}\Bigg[\Big|\{i \in \{1, \ldots, k\} \mid \Pr_{y \in \mathcal{Y}_0}[\mathrm{DISJ}(x_i, y) = 1] \le 0.2\}\Big| \ge 0.3k\Bigg] \le 0.2k/0.3k \le 2/3.$

As $1/4 + 2/3 < 1$, there is a non-zero probability that both $|\mathcal{Y}_0| \ge 10 \cdot (0.8)^{\sigma_y}|\Gamma_y|$ and $|\{i \in \{1, \ldots, k\} \mid \Pr_{y \in \mathcal{Y}_0}\big[\mathrm{DISJ}(x_i, y) = 1\big] > 0.2\}| > 0.7k.$ ■

Let us now show why choosing $\mathcal{Y}_0$ using Lemma 19 helps us construct $\mathcal{Y}$. We will need one more fact that formalizes the following natural intuition: if we take a sufficiently large subset of Γ_y, then the distribution of the ones in the vectors of this subset is fairly well spread out among $\{1, \ldots, n\}$. For any such set $S \subseteq \Gamma_y$, let $C(S) = \big\{i \in \{1, \ldots, n\} \mid \Pr_{y \in S}[y_i = 1] \le \frac{1}{2n}\big\}$.

Lemma 21 *Let $|S| > 2 \cdot \big(0.8\big)^{\sigma_y} \cdot |\Gamma_y|$. Then, $\big|C(S)\big| \le 0.2n$.*

Proof: Suppose that $0.2n < |C(S)| \le n$. Let S' denote the set of y's in S such that $y_j = 0$ for all $j \in C(S)$. Choose a random $y \in \Gamma_y$. By definition, the probability that y has a one in some index in $C(S)$ is at most $|C(S)| \cdot \frac{1}{2n} \le \frac{1}{2}$. Hence, $|S'| \ge \frac{1}{2}|S|$. This leaves at most $n - |C(S)| < 0.8n$ locations for the σ_y ones that are in each such y. Hence, $|S'| \le \binom{0.8n}{\sigma_y} \le (0.8)^{\sigma_y} \cdot \binom{n}{\sigma_y}$, giving the contrapositive of the result. ■

We now prove Lemma 17 by showing that Lemma 19 and Lemma 21 can be combined to get our desired set $\mathcal{Y}(x_1, \ldots, x_k)$.

Proof of Lemma 17: Pick $\mathcal{Y}_0(x_1, \ldots, x_k)$ according to Lemma 19. Construct $\mathcal{Y} \subseteq \Gamma_y$ by independently choosing $30n$ elements at random from $\mathcal{Y}_0(x_1, \ldots, x_k)$. By definition of $\mathcal{Y}_0$, Bob sends the same message to Charlie for all $y \in \mathcal{Y}$. It remains to show that

$$\mathrm{Exp}_{\mathcal{Y}}[|Cover((x_1, \ldots, x_k), \mathcal{Y})|] \ge \frac{1}{2}nk.$$

For any $i \in \{1, \ldots, k\}$, let $\mathcal{Y}^i$ be the set of $y \in \mathcal{Y}_0$ such that $\mathrm{DISJ}(x_i, y) = 1$. Suppose that $\Pr_{y \in \mathcal{Y}_0}\big[\mathrm{DISJ}(x_i, y) = 1\big] > 0.2$. Then, from Lemma 19, we have $|\mathcal{Y}^i| > 0.2|\mathcal{Y}_0| \ge 2 \cdot \big(0.8\big)^{\sigma_y} \cdot |\Gamma_y|$. Hence, by Lemma 21, $\big|C(\mathcal{Y}^i)\big| \le 0.2n$. Thus, for any j not in $C(\mathcal{Y}^i)$,

$$\Pr_{y \in \mathcal{Y}_0}\big[j \in Cover(x_i, y)\big] = \Pr\big[y \in \mathcal{Y}^i\big] \cdot \Pr\big[y_j = 1 \,|\, y \in \mathcal{Y}^i\big] \ge 0.2 \cdot \frac{1}{2n} = \frac{0.1}{n}.$$

But we independently choose $30n$ different y's to be in $\mathcal{Y}$. Hence,

$$\Pr_{\mathcal{Y}}[j \notin Cover(x_i, y) \text{ for all } y \in \mathcal{Y}] \le \big(1 - \frac{0.1}{n}\big)^{30n} \le \frac{1}{e^3}.$$

We conclude that

$$\begin{aligned} \mathrm{Exp}_{\mathcal{Y}}[|Cover((x_1, \ldots, x_k), \mathcal{Y})|] &\ge \sum_{i \in I(X)} \sum_{j \notin C(\mathcal{Y}^i)} \Pr_{\mathcal{Y}}[j \in Cover(x_i, y) \text{ for some } y \in \mathcal{Y}] \\ &> (0.7)k \cdot (1 - 0.2)n \cdot \big(1 - \frac{1}{e^3}\big) > 0.5nk. \end{aligned}$$

■

This completes the proof of Theorem 13, the lower bound for set-disjointness.

5 Non-Adaptive Data Structures

We now show that our lower bounds on restricted protocols in the previous section can be used to obtain lower bounds for restricted data-structures. In particular, let us recall the key three phase data-structure problem originally considered by Pǎtraşcu. We state the problem slightly more generally in the cell-probe model, with each cell containing $w \in O(\log n)$ bits. Let f be a function that takes a pair (X, Y) as input and produces a Boolean output. In Phase 1, the data structure algorithm gets $X_1, \ldots, X_k$, with each $X_i \in \{0,1\}^n$. Thus, each X_i is specified by $O(n/w)$ words.

The algorithm pre-processes these items and stores them using s cells in total. In Phase 2, the data structure algorithm gets Y, which is specified using n' words. The update time, i.e., the number of cells probed (read from or written to) by the algorithm in this phase, is expressed as $n't_u$, where t_u is the amortized update time. Finally, in Phase 3, the data structure algorithm receives a query $i \in \{1, \dots, k\}$ and the algorithm must output $f(X_i, Y)$. The query time, i.e., the number of cells probed in this phase, is denoted by t_q. For example, when f is the inner product or disjointness function, then $X, Y \in \{0,1\}^n$ and $n' \in O(n/\log n)$. For the indexing function, $X \in \{0,1\}^n$, but $Y \in \{1, \dots, n\}$, which can be stored in one word. The indexing function can also be viewed a special case of disjointness, where Y is restricted to contain exactly one 1.

A natural goal is to understand the relationship between s, t_u, t_q and w for the best algorithm for f. Then, we would like to show that for a hard function f, unless s is very large (super-polynomial in n), at least one of t_u and t_q is polynomially large, i.e. $n^{\Omega(1)}$.

Pătraşcu's reduction shows that proving such lower bounds for suitable functions f results in polynomial lower bounds for various (dynamic) data-structure problems. In particular, he showed that strong lower bounds for a host of natural dynamic problems follow from a strong lower bound for set-disjointness. It is worth noting that such a lower bound for the inner product function, which we believe will be easier to prove, implies breakthrough lower bounds for several dynamic problems, for example, the problem of determining the parity of the length of the shortest path.

We can prove very strong lower bounds when the three phase algorithm has the following restriction: the third phase of the algorithm is nonadaptive, i.e. for each i, there is a fixed subset S_i of cells that get probed. Just to be clear, every other phase is entirely unrestricted. In fact, our bounds hold even when there are no restrictions on the pre-processing done in Phase 1, i.e., the total number of cells, sm is unrestricted. Such bounds were first proved by [6]. Their arguments worked directly on the three-phase data-structure problem.

Here we show that their lower bounds for nonadaptive query algorithms can also be obtained by viewing them as restricted protocols in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. More precisely, nonadaptive query algorithms are $(m, 0, q)$-protocols in the language of Section 4.2. Alice sends $m = wn't_u$ bits $A = A(X, Y)$ and Charlie sends $q = wt_q$ bits $C = C(X, i)$ both to Bob. Bob answers with $B(Y, i, A, C)$. Thus, it is a further restriction of the 1.5 round protocols that we considered before. The following lemma makes these connections precise.

Lemma 22 *Let D be a dynamic (randomized) algorithm for the three phase problem corresponding to function $f(X, Y)$, where Y is specified by n' words. If D uses only non-adaptive queries, then there exists a (randomized) $(wn't_u, 0, wt_q)$-protocol for $SEL_f^{k\times 1}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, where w is the word-size, and t_u, t_q are respectively the amortized update and query times of D. Specifically when the problem is the indexing function (so $n' = 1$), the reduction is to a $(wt_u, 0, wt_q)$-protocol.*

Proof: Let U_{XY} be the set of cells that were touched in the second phase of the run of D. Clearly $|U| \leq n't_u$. Further, let $Q[i]$ be the set of cells that are read in the third phase by D when queried with index i. Note that $Q[i]$ is independent of $X_1, \dots, X_k$ and Y as D uses non-adaptive query algorithm. Also, $|Q[i]| \leq t_q$ because i fits into one cell word.

Now consider the $A \overset{B}{\to} (B \leftrightarrow C)$ model. Alice, having access to all of $X_1, \dots, X_k$ and Y sends just the contents (without addresses) of the cells in U before they were updated in Phase 2, to Bob. This costs $wn't_u$ bits of advice. Charlie, knows $X_1, ..., X_k$ and i. Knowing i, he knows $Q[i]$. He sends only the un-updated contents (no addresses) of the cells in $Q[i]$. Now, we argue that Bob is in a good position to effectively simulate D. First he knows Y. So he runs the update algorithm. For each of his reads, he can get the contents from the original data structure via Alice's message. The cells that he has to update/write, he can do so himself. Thus, using just Alice's message and Y, he is able to recover which addresses got updated (i.e. written) and what their new contents are. Further, he knows i and hence the addresses of all cells in set $Q[i]$. The contents of the updated cells in $Q[i]$ he has recovered with the help of Y and Alice's message already, as we argued. For the contents of the unupdated, he gets them from Bob's message. All in all he now can simulate the query phase of D and give the answer. We remark that, if D is randomized, then the simulation in the $A \overset{B}{\to} (B \leftrightarrow C)$ model will also be randomized. ■

Combining Lemma 22 with Theorem 15 immediately gives the following result. Note that our result matches the lower bounds in [6] up to logarithmic factors.

Theorem 23 *Let D be a dynamic (randomized) algorithm for the three phase problem corresponding to the indexing function. Either $t_u \in \Omega\big(\frac{\epsilon^4}{w \log^2 k} k\big)$ or $t_q \in \Omega\big(\frac{\epsilon^4}{w \log^2 k} n\big)$.*

6 Open Problems and Conclusions

The $A \overset{B}{\to} (B \leftrightarrow C)$ model is a new variant of the communication complexity model that may be useful for studying the complexity of many dynamic data structure problems. Pătraşcu conjectured that for any hard two-player function f, the asymmetric version of f is hard in the $A \overset{B}{\to} (B \leftrightarrow C)$ model when the length of Alice's advice is $o(k)$. Suppose that k is polynomial in n.

In this paper, we have obtained surprising counterexamples to this conjecture: we have exhibited a function with maximal two-player randomized complexity that is easy in the $A \overset{B}{\to} (B \leftrightarrow C)$ model using very little advice from Alice. We have also shown nontrivial upper bounds for set-disjointness in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, when the length of Alice's advice is $O\big(\sqrt{n}\log n\big)$.

The most important unresolved question is the exact complexity of asymmetric set-disjointness in the $A \overset{B}{\to} (B \leftrightarrow C)$ model. It is still possible that $\mathrm{SEL}^{k\times 1}_{DISJ}$ requires polynomial complexity (n^ϵ for some $\epsilon > 0$), which would yield polynomial lower bounds for a large collection of dynamic data structure problems. More generally, no superpolylogarithmic lower bounds for $\mathrm{SEL}^{k\times 1}_{f}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model are presently known for any function, even via a non-constructive argument.

One intuition that we have relates the complexity of $\mathrm{SEL}^{k\times 1}_{f}$ to the two-party complexity of f under *product distributions*. More specifically, if y is independent of each x_i, then Bob and Charlie can solve $f(x_i, y)$ on their own (without the help of Alice) using the best product distribution algorithm. On the other hand, if y depends on some x_i then Alice should be able to use x_i to teach Charlie a lot about y by telling him the differences and similarities between y and x_i. This was precisely the intuition used in our upper bound for $\mathrm{SEL}^{k\times 1}_{DISJ}$.

Motivated by this intuition, we conjecture that for any function f, the worst-case instances of $\mathrm{SEL}^{k\times 1}_{f}$ are obtained by some product distribution, where each x_i is chosen independently of y, to ensure that the x_i's do not contain information about y that can be exploited by Alice. We conjecture, further, that any lower bound for the two-player game for f under product distributions (x_i and y are chosen independently) acts as a lower bound for $\mathrm{SEL}^{k\times 1}_{f}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ game. Thus, for asymmetric set-disjointness, we conjecture a $\sqrt{n}$ lower bound in the $A \overset{B}{\to} (B \leftrightarrow C)$

model, which matches the tight $\sqrt{n}$ lower bound for set-disjointness over product distributions in the 2-party model [2].

Acknowledgements

This work was supported by the Natural Science and Engineering Research Council of Canada. The first author was also supported by a Ramanujan Fellowship from the Department of Science and Technology, India, and a Postdoctoral Fellowship from the Ontario Ministry of Research and Innovation.

References

[1] S. Alstrup, T. Husfeldt, and T. Rauhe. Marked ancestor problems. In *Proceedings of the 39th IEEE Symposium on Foundations of Computer Science(FOCS)*, pages 534–544, 1998.

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

[3] Z. Bar-Yossef, T. Jayram, R. Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput and Syst. Sci*, 68(4):702–732, 2004.

[4] P. Beame, T. Pitassi, N. Segerlind, and A. Wigderson. A strong direct product theorem for corruption and the multiparty communication complexity of disjointness. *Computational Complexity*, 15(4):391–432, 2006.

[5] S. Ben-David, N. Eiron, and H. U. Simon. Limitations of learning via embeddings in Euclidean half spaces. *J. Mach. Learn. Res.*, pages 441–461, 2003.

[6] J. Brody and K. G. Larsen. Adapt or die: Polynomial lower bounds for non-adaptive dynamic data structures. *Theory of Computing*, 11(882):751–768, 2015.

[7] A. Chattopadhyay, J. Edmonds, F. Ellen, and T. Pitassi. A little advice can be very helpful. In *Proceedings of the 23rd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 615–625, 2012.

[8] B. Chor and O. Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. In *IEEE 26th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 429–442, 1985.

[9] M. L. Fredman. Observations on the complexity of generating quasi-gray codes. *SIAM J. Comput.*, 7(2):134–146, 1978.

[10] M. L. Fredman and M. E. Saks. The cell probe complexity of dynamic data structures. In *Proceedings of the 21st ACM Symposium on Theory of Computing (STOC)*, pages 345–354, 1989.

[11] O. Goldreich, N. Nisan, and A. Wigderson. On yao's xor-lemma. *Studies in Complexity and Cryptography*, pages 273–301, 2011.

[12] M. R. Henzinger and M. L. Fredman. Lower bounds for fully dynamic connectivity problems in graphs. *Algorithmica*, 22(3):351–362, 1998.

[13] T. Husfeldt and T. Rauhe. New lower bound techniques for dynamic partial sums and related problems. *SIAM J. Comput.*, 32(3):736–753, 2003.

[14] T. Husfeldt, T. Rauhe, and S. Skyum. Lower bounds for dynamic transitive closure, planar point location, and parentheses matching. In *Proceedings of the 5th Scandinavian Workshop on Algorithm Theory (SWAT)*, volume 1097 of *Lecture Notes in Computer Science*, pages 198–211, 1996.

[15] B. Kalyanasundaram and G. Schnitger. The probabilistic communication complexity of set intersection. *SIAM J.Discrete Math*, 5(4):545–557, 1992.

[16] M. J. Kearns and U. V. Vazirani. *An introduction to computational learning theory.* MIT Press, 1994.

[17] H. Klauck. A strong direct product theorem for disjointness. In *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*, pages 77–86, 2010.

[18] J. Kleinberg and E. Tardos. *Algorithm Design.* Addison Wesley, 2006.

[19] I. Kremer, N. Nisan, and D. Ron. On randomized one-round communication complexity. *Computational Complexity*, 8(1):21–49, 1999.

[20] E. Kushilevitz and N. Nisan. *Communication complexity.* Cambridge University Press, 1997.

[21] K. G. Larsen. The cell probe complexity of dynamic range counting. In *Proceedings of the 44th ACM Symposium on Theory of Computing (STOC)*, pages 85–94, 2012.

[22] K. G. Larsen. Higher cell probe lower bounds for evaluating polynomials. In *Proceedings of the 53rd IEEE Symposium on Foundations of Computer Science(FOCS)*, pages 293–301, 2012.

[23] T. Lee, A. Shraibman, and R. Spalek. A direct product theorem for discrepancy. In *23rd IEEE Conference on Computational Complexity*, pages 71–80, 2008.

[24] N. Linial and A. Shraibman. Learning complexity vs. communication complexity. In *Proceedings of the 23rd IEEE Conference on Computational Complexity*, pages 53–63, 2008.

[25] M. Minsky and S. Papert. *Perceptrons.* MIT Press, 1969.

[26] I. Newman. Private vs. common random bits in communication complexity. *Information Processing Letters*, 39(2):67–71, 1991.

[27] N. Nisan. The communication complexity of Threshold gates. In V. S. D. Miklós and T. Szönyi, editors, *Combinatorics, Paul Erdös is eighty*, volume 1, pages 301–315. Janos Bolyai Math Society, 1993.

[28] M. Pătraşcu. Towards polynomial lower bounds for dynamic problems. In *Proceedings of the 42nd ACM Symposium on Theory of Computing(STOC)*, pages 603–610, 2010.

[29] M. Pătraşcu. Unifying the landscape of cell-probe lower bounds. *SIAM J. Comput.*, 40(3):827–847, 2011.

[30] M. Pătraşcu and E. D. Demaine. Logarithmic lower bounds in the cell-probe model. *SIAM J. Comput.*, 35(4):932–963, 2006.

[31] A. Razborov. On the distributional complexity of Disjointness. *Theor.Comput.Sci.*, 106(2):385–390, 1992.

[32] R. Shaltiel. Towards proving strong direct product theorems. *Computational Complexity*, 12(1-2):1–22, 2003.

[33] A. A. Sherstov. Communication complexity under product and nonproduct distributions. *Computational Complexity*, 19(1):135–150, 2010.

[34] A. C. Yao. Should tables be sorted? *J.ACM*, 28:615–628, 1981.

[35] A. C. Yao. Lower bounds by probabilistic arguments. In *Proceedings of the 24th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 420–428, 1983.

[36] A. C.-C. Yao. Some complexity questions related to distributive computing (preliminary report). In *Proceedings of the 11th ACM Symposium on Theory of Computing (STOC)*, pages 209–213, 1979.