A Little Advice Can Be Very Helpful

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

Abstract

Proving superpolylogarithmic lower bounds for dynamic data structures has remained an open problem despite years of research. Recently, Pătraşcu proposed an exciting new 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.

1 Introduction

Obtaining lower bounds for dynamic data structures in the cell probe model has been a challenge. In 1989, Fredman and Saks [5] 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. 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, 6, 7, 8]. In 2004, Pătraşcu and Demaine [17] introduced a beautiful information theoretic technique to prove $\Omega(\log n)$ lower bounds for the partial sums problem and dynamic connectivity in undirected graphs. More recently, Pătraşcu [16] 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. Despite these advances, it remains a longstanding open problem to prove polynomial (or even superpolylogarithmic) lower bounds for any dynamic data structure problem.

Pătraşcu [15] 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 $f : X \times Y \to \{0,1\}$, Pătraşcu 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.

*University of Toronto, email: arkadev@cs.toronto.edu
†York University, email: jeff@cs.yorku.ca
‡University of Toronto, email: faith@cs.toronto.edu
§University of Toronto, email: toni@cs.toronto.edu

Pătraşcu presented simple reductions from the problem of computing $\text{SEL}^{k\times 1}_{DISJ}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model to many dynamic problems in the cell probe model. ($DISJ$ denotes the set-disjointness problem, where $DISJ(x, y) = 0$ if and only if x and y, when viewed as subsets of $\{1, \ldots, n\}$, are disjoint, i.e. for all $i \in \{1, \ldots, n\}$, $x[i] = 0$ or $y[i] = 0$.) These reductions prove that, if $\text{SEL}^{k\times 1}_{DISJ}$ 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 $\text{SEL}^{k\times 1}_{DISJ}$ 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 [15] 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 1.1. (Pătraşcu) *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be any function. Consider a protocol π for computing $SEL^{k\times 1}_f$ 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.

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 $EQ(x, y) = 1$ 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.

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

Theorem 1.1. *There exists a Boolean function f with two-party* **randomized** *communication complexity $\Omega(n)$ such that $SEL^{k\times 1}_f$ 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 $DISJ(x_i, y)$ 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 1.2. *There is a deterministic protocol for $SEL^{k\times 1}_{DISJ}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, in which the total number of bits communicated is $O\left(\sqrt{n}(\log k)(\log k + \log^2 n)/\sqrt{\log n}\right)$.*

Note that when $k \in n^{O(1)}$, the amount of communication is $O\big(\sqrt{n}\log^{5/2} n\big)$. It is worth remarking that the above theorem 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 $\text{SEL}^{k\times 1}_{DISJ}$, for some constant $0 < \delta \leq 1/2$. Theorem 1.1 and Theorem 1.2 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 solving asymmetric problems in the $A \xrightarrow{B} (B \leftrightarrow C)$ model do not differ by much. Specifically, for any Boolean function f with two-party randomized communication complexity R, we show that $\mathrm{SEL}_f^{k\times 1}$ has deterministic communication complexity $O((R + \log n + \log k)\log k)$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ 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 $A \xrightarrow{B} (B \leftrightarrow C)$ model.

Finally, we provide lower bounds in the $A \xrightarrow{B} (B \leftrightarrow C)$ model for some restricted classes of protocols, which include those protocols used for our upper bounds in Theorem 1.1 and Theorem 1.2. 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.

These arguments suggest that, for more efficient protocols, significantly new ideas are needed. Our lower bound arguments also highlight the difficulty one faces in analyzing protocols in which Alice sends more than n bits of advice. This remains the main challenge for proving stronger lower bounds.

Overview of Paper. In Section 2, we introduce notation and define necessary concepts from communication complexity. In Section 3, we present a function f that has $\Omega(n)$ randomized 2-party complexity, but such that $\mathrm{SEL}_f^{k\times 1}$ can be solved deterministically with only $\log^{O(1)} k$ communication. We also prove that randomization does not help for solving $\mathrm{SEL}_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model. In Section 4, we prove our upper bound for set-disjointness. In Section 5, we prove lower bounds in restricted settings. We conclude in Section 6 with some open problems.

2 Preliminaries

Communication complexity was first studied for the two-party model [21], in which the input is partitioned between two players who compute a Boolean function of their inputs. For any Boolean function $f : X \times Y \to \{0,1\}$, we use $D(f)$ to denote the deterministic complexity of f, which is the minimum number of bits communicated in any deterministic protocol that computes f correctly on all inputs. If $\mu : X \times Y \to [0,1]$ is a probability distribution and $\epsilon \geq 0$, the ϵ-error complexity of f for distribution μ, which we denote by $D_\mu^\epsilon(f)$, is the minimum number of bits communicated in any deterministic protocol π that computes f and errs on a set of inputs with total weight at most ϵ. Note that $D_\mu^\epsilon(f) \leq D(f) \leq n+1$ for any f, μ and ϵ.

In a randomized two-party protocol, the two players are provided with a public (shared) sequence r of random bits. A protocol π for f has error probability ϵ if $\max\{\Pr[\pi(x,y,r) \text{ does not output } f(x,y)] \mid x \in X, y \in Y\} = \epsilon$, 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 (see [20]) proved the following relationship between randomized and distributional communication complexities.

Theorem 2.1. *For any Boolean function f, $R^\epsilon(f) = max_\mu\{D_\mu^\epsilon(f)\}$.*

Newman [14] proved that any randomized two-party communication protocol (with public randomness) 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 that holds for any nonuniform model of computation, such as communication protocols, boolean circuits, decision trees and non-uniform Turing machines:

Theorem 2.2. *If there is a randomized computation for a function with domain U and error probability at most ϵ, 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. A randomized computation of a function can be expressed as a function of both the input and the choice of random bit string. Let A be a randomized computation of a function f with domain U and error probability at most ϵ. Then for all inputs $u \in U$, the probability that $A(u,r)$ outputs $f(u)$ is at least $1-\epsilon$, where the probability is taken over all choices r for the random bit string. We show that there exist $t \in O(\log|U|)$ strings $r_1, \ldots, r_t$ such that, for each input

u, if we choose a string r at random from $r_1, \ldots, r_t$, then $A(u, r) = f(u)$ with probability at least $1 - \delta$, where $\delta \in O(\epsilon)$.

Suppose $r_1, \ldots, r_t$ are chosen independently at random from the space of random strings used by A. For any input $u \in U$, the probability that $A(u, r_i) = f(u)$ is at least $1 - \epsilon$, for all $i = 1, \ldots, t$. Hence the expected number of these random strings for which A outputs $f(u)$ on input u is at least $(1 - \epsilon)t$. By the Chernoff bound, there exists $\delta \in O(\epsilon)$ such that the probability that the number of such strings is less than $(1 - \delta)t$ is exponentially small in t. Hence, it is possible to make this probability less than $1/|U|$ by choosing $t \in \Theta(\log |U|)$. The union bound implies that, with probability less than 1, there is an input $u \in U$ for which $|\{j \in \{1, \ldots, t\} \mid A(u, r_j) = f(u)\}| < (1 - \delta)t$. Hence, there exist choices of $r_1, \ldots, r_t$ such that, for each input $u \in U$, $|\{j \in \{1, \ldots, t\} \mid A(u, r_j) = f(u)\}| \geq (1 - \delta)t$.

On any input $u \in U$, the computation A' chooses $j \in \{1, \ldots, t\}$ uniformly at random and performs $A(u, r_j)$. Then $\Pr[A'(u, j) = f(u)] \geq 1 - \delta$.

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

Proof. By THEOREM 2.2, we may assume that π uses only $O(\log n + \log k)$ random bits. Choose $t \in \Theta((\log k)/\epsilon^2)$ strings $r_1, \ldots, r_t$ independently at random from the space of random strings used by π. Let $x \in \{0,1\}^{nk}$, $y \in \{0,1\}^n$, and $i \in \{1, \ldots, k\}$. Then, for each $j \in \{1, \ldots, t\}$, $\Pr[\pi(x, y, i, r_j)$ outputs $f(x_i, y)] \geq \frac{1}{2} + \epsilon$. A simple application of the Chernoff bound shows the probability that $\pi(x, y, i, r_j)$ does not output $f(x_i, y)$ for the majority of $j \in \{1, \ldots, t\}$ is less than $1/k$. 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, y, Alice can find a sequence of $t \in \Theta((\log k)/\epsilon^2)$ strings $r_1, \ldots, r_t$ 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 $j = 1, \ldots, t$. 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.

We immediately get the following corollary:

COROLLARY 3.1. *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be any Boolean function such that $R^\epsilon(f) = (\log n)^{O(1)}$, for some constant $\epsilon < 1/2$. If $k \in n^{O(1)}$, then there exists a deterministic protocol for $SEL_f^{k \times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model where Alice sends $O(\log^2 n)$ bits to Bob and Bob and Charlie communicate $(\log n)^{O(1)}$ bits.*

There are well known functions that have very high deterministic complexity in the 2-party model, but have efficient 2-party randomized protocols. For example, the equality function, EQ, has $D(EQ) = n + 1$, but has $R^\epsilon(EQ) \in O(1)$ for any constant $\epsilon > 0$. The greater than function, GT, defined by $GT(x, y) = 1$ if and only if $x \geq y$, also has $D(GT) \in \Omega(n)$, but has $R^\epsilon(GT) \in O(\log n)$ for any constant $\epsilon > 0$. It follows from Corollary 3.1 that both $\mathrm{SEL}_{\mathrm{EQ}}^{k \times 1}$ and $\mathrm{SEL}_{\mathrm{GT}}^{k \times 1}$ have efficient deterministic protocols in the $A \xrightarrow{B} (B \leftrightarrow C)$ model. These functions witness the refutation of Patrascu's conjecture for deterministic protocols.

3.1 Alice as Teacher Next, we show that there exists a Boolean function f with very high bounded error randomized complexity in the two-party model, for which $\mathrm{SEL}_f^{k \times 1}$ has efficient deterministic protocols in the $A \xrightarrow{B} (B \leftrightarrow C)$ 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 $\Pr[h(x) \neq f(x)] \leq \epsilon$. 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 [9], shows the relevance of VC dimension to learning.

THEOREM 3.2. *Let $\mathcal{S}$ be 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*

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

Furthermore, the hypothesis that L outputs agrees with all the examples it is given as input.

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 [11] 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 3.3. *Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ be a Boolean function and let $0 \le \epsilon < 1$ be a constant. Then, there exists a randomized protocol π for $SEL_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model with error probability at most ϵ and*

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

Proof. Using Yao's min-max principle (THEOREM 2.1), 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 $x = (x_1,\ldots,x_k) \in \{0,1\}^n k$ and $y \in \{0,1\}^n$, Alice can determine the conditional distribution $\mu_{x,y}$ induced on $\{1,\ldots,k\}$. By THEOREM 3.2, 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

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

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 $A \xrightarrow{B} (B \leftrightarrow C)$ model. Recently, Sherstov [19], using earlier results of Ben-David et.al. [4] and Linial and Shraibman [13], showed that there exists a function f with high randomized communication complexity in the two-party model such that M_f has low VC-dimension. This result is implicit in the proof of Theorem 3.5 of his paper.

THEOREM 3.4. *For any constant $0 \le \epsilon < 1$, 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 1.1. *There exists a Boolean function f with two-party* **randomized** *communication complexity $\Omega(n)$ such that, for $n \le k \in 2^{(\log n)^{O(1)}}$, $SEL_f^{k\times 1}$ has a* **deterministic** *protocol in the $A \xrightarrow{B} (B \leftrightarrow C)$ model, in which Alice sends Bob $O\big(\log^2 k\big)$ bits and then Bob and Charlie communicate a total of $O\big(\log^2 k\big)$ bits.*

Proof. By THEOREM 3.4, 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 3.3 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 3.1, we derandomize π to obtain a deterministic protocol π' such that $\mathrm{CC}_{A\to B}(\pi') = O(\log^2 k)$ and $\mathrm{CC}_{B\leftrightarrow C}(\pi') = O(\log^2 k)$.

This disproves Conjecture 1.1, even for randomized protocols.

4 An Upper Bound for Set Disjointness

In this section, we construct a protocol for $\mathrm{SEL}_{DISJ}^{k\times 1}$ 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 $\{1, \ldots, n\}$.

We begin by considering some inputs for which computing $DISJ$ is easy in the standard two-party model.

LEMMA 4.1. *Let $\sigma, d \geq 1$. There is a simple randomized two-party protocol for computing $DISJ(x', y)$ with $\sigma \log n + 1$ bits of communication and with error probability at most e^{-d} for all $x', y \subseteq \{1, \ldots, n\}$ such that $|x'| \leq \sigma$ or $|x' \cap y| > d|x'|/\sigma$.*

Proof. If $|x'| \leq \sigma$, then Charlie sends each element of x' to Bob. Otherwise Charlie randomly selects σ distinct elements of x' and sends them to Bob. Since each element of x' is in $\{1, \ldots, n\}$, it can be represented using $\log n$ bits. If Bob receives any element that is in y, he sends 1, to indicate that the sets are not disjoint. Otherwise, he sends 0.

If $|x'| \leq \sigma$, then this protocol always correctly computes $DISJ(x', y)$. If $|x' \cap y| > d|x'|/\sigma$, then $\Pr[u \in y \mid u \in x'] > d/\sigma$, so $\Pr[\text{this protocol incorrectly computes } DISJ(x', y)] \leq (1 - d/\sigma)^\sigma \leq e^{-d}$.

The following protocol can be thought of as a sequence of phases in which Alice sends a carefully chosen index $r \in \{1, \ldots, k\}$ plus some of the elements of $x_r \cap y$ to Bob, who forwards the information to Charlie. This allows Charlie, who knows $x_1, \ldots, x_k$, to learn information about y, specifically, that certain elements of $\{1, \ldots, n\}$ are in y and that others are not. Charlie can also check if any of the elements of y that it was sent are in x_i and, if so, knows that $DISJ(x_i, y) = 1$. If not, Charlie can compute a set S such that $x_i \cap y \subseteq S \subseteq \{1, \ldots, n\}$, which decreases in size each phase. This continues until each of the possible problem instances $DISJ(x_1 \cap S, y), \ldots, DISJ(x_k \cap S, y)$, including $DISJ(x_i \cap S, y) = DISJ(x_i, y)$, can be computed with error probability at most e^{-d} using the simple protocol of LEMMA 4.1. This motivates the following definition. For any $S \subseteq \{1, \ldots, n\}$, let $EASY(S) = \{r \in \{1, \ldots, k\} \mid x_r \cap y \not\subseteq S \text{ or } |x_r \cap S| \leq \sigma \text{ or } |x_r \cap S \cap y| > d|x_r \cap S|/\sigma\}$. Note that $EASY(\phi) = \{1, \ldots, k\}$

Suppose that $S' \subseteq S$. If $r \notin EASY(S')$, then $x_r \cap y \subseteq S' \subseteq S$, $\sigma < |x_r \cap S'| \leq |x_r \cap S|$, and $|x_r \cap y \cap S| = |x_r \cap y \cap S'| \leq d|x_r \cap S'|/\sigma \leq d|x_r \cap S|/\sigma$, so $r \notin EASY(S)$. Thus, $EASY(S) \subseteq EASY(S')$.

THEOREM 4.1. *Let $\sigma, d \geq 1$. Then $SEL_{DISJ}^{k\times 1}$ can be computed by a randomized protocol with error probability at most e^{-d} in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, in which Alice sends at most $n(\log k)/\sigma + dn(\log n)/\sigma$ bits and at most $n(\log k)/\sigma + dn(\log n)/\sigma + \sigma \log n + 2$ bits are communicated by Bob and Charlie.*

Proof. Given $x_1, \ldots, x_k$, and y, Alice performs the following algorithm:

$S \leftarrow \{1, \ldots, n\}$
while $EASY(S) \neq \{1, \ldots, k\}$ do
  $r \leftarrow \min(\{1, \ldots, k\} - EASY(S))$
  Alice sends r and the elements of $x_r \cap S \cap y$ to Bob
  $S \leftarrow S - x_r$

In each iteration, since $r \notin EASY(S)$, $|x_r \cap S| > \sigma$, so the size of S decreases by more than σ. Thus, the number of iterations is less than n/σ and Alice sends at most n/σ indices in $\{1, \ldots, k\}$, each of which can be represented using $\log k$ bits.

Furthermore, $r \notin EASY(S)$ implies that $|x_r \cap S \cap y| \leq d|x_r \cap S|/\sigma\}$. Hence, in each iteration, the number of elements of y sent by Alice is less than d/σ times the number of elements removed from S. Since $S \subseteq \{1, \ldots, n\}$, Alice sends fewer than dn/σ elements of y altogether. Each element of y can be represented using $\log n$ bits. Thus, in total, Alice sends fewer than $n(\log k)/\sigma + dn(\log n)/\sigma$ bits to Bob.

Then Bob forwards Alice's message to Charlie. Charlie, by looking at Alice's message, learns the final S that Alice computed. Note that he also learns about many elements that are not in y. For every index $j \notin S$ that Alice does not refer to in her message, Charlie correctly infers $j \notin y$. Hence, it suffices for Charlie to check whether any of the elements in y that Alice sent are also in x_i. If so, Charlie sends 1 and the protocol terminates. Otherwise, Charlie sends 0, indicating that $x_i \cap y \subseteq S$. In this case, Bob and Charlie perform the simple randomized two-party protocol described in LEMMA 4.1 to compute $DISJ(x', y)$, where $x' = x_i \cap S$, using at most $\sigma \log n + 1$ additional bits. Since $i \in EASY(S)$, the error probability is at most e^{-d}.

Note that the protocols in LEMMA 4.1 and THEOREM 4.1 both have one-sided error: Whenever Charlie ends by sending 1, saying that the sets x_i and y are not disjoint, he has a witness to their non-disjointness. In the first of these protocols, all the communication is from Charlie to Bob, except for the final bit, which is from Bob to Charlie. In the second protocol, Alice sends information to Bob, which he simply forwards to Charlie. Then Charlie either sends information to Bob, who sends the final bit, unless Charlie sent 1, in which case Bob does not respond.

Setting $d = \log n$ and $\sigma = \sqrt{n \log n}$ and applying THEOREM 3.1 gives our upper bound for the set disjointness function.

THEOREM 1.2. *There is a deterministic protocol for $SEL^{k\times 1}_{DISJ}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model, in which the total number of bits communicated is $O\left(\sqrt{n}\log k(\log k + \log^2 n)/\sqrt{\log n}\right)$.*

5 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 anyway. Third, there is 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 5.1, we prove our upper bounds cannot be substantially improved, unless we allow Alice to give more than $\sqrt{n}$ bits of advice, even if players interact arbitrarily. In Section 5.2, we complement this by showing the upper bound for set-disjointness cannot be improved if Bob and Charlie have limited interaction.

5.1 Lower Bounds via Strong Direct Product Theorems

In this subsection, we prove lower bounds in the $A \overset{B}{\to} (B \leftrightarrow C)$ model for $\mathrm{SEL}^{k\times 1}_f$, provided that the function f has a strong direct product theorem.

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. Shaltiel [18] 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 [12] strengthened Shaltiel's result by proving a strong direct product theorem for functions where we 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 [3]. Finally, Klauck [10] the following optimal strong direct product theorem for set disjointness.

THEOREM 5.1. *There exist constants $0 < \beta < 1$ and $\alpha > 0$ such that every randomized protocol which computes $DISJ^{(k)} : \{0,1\}^{nk} \times \{0,1\}^{nk} \to \{0,1\}$ using at most βkn bits of communication has worst case success probability less than $2^{-\alpha k}$.*

Using the above theorem, we obtain the following lower bound for asymmetric set disjointness in the $A \overset{B}{\to} (B \leftrightarrow C)$ model.

THEOREM 5.2. *There exist constants $0 < \beta < 1$ and $\alpha > 0$ such that in any deterministic protocol for $SEL^{\sqrt{n}\times 1}_{DISJ}$ in the $A \overset{B}{\to} (B \leftrightarrow C)$ model where Alice sends at most $\alpha\sqrt{n}$ bits, Bob and Charlie must communicate at least $\beta\sqrt{n}$ bits.*

Proof. Let α and β be constants that satisfy THEOREM 5.1 and let $k = \sqrt{n}$.

Suppose, for sake of contradiction, that there is a deterministic protocol for $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$, 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 large success probability (w.r.t. μ) at least $2^{-\beta k}$. Using Yao's min-max principle [20] with such protocol yields a contradiction to THEOREM 5.1. Hence, all that remains is to build such protocols for every distribution μ.

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 $\mathrm{SEL}^{k\times 1}_{\mathrm{DISJ}}$ as follows: $y = y'_1 \cdots y'_k$ and, for each $i \in \{1, \ldots, k\}$, $x_i = 0^{(i-1)k} x'_i 0^{(n-i)k}$, i.e., it is all 0's except for the i'th block of k bits, which is x'_i. Let μ' be the distribution generated on $\{0,1\}^{nk} \times \{0,1\}^n$ by the above transformation, given μ on $\{0,1\}^{\sqrt{n}k} \times \{0,1\}^{\sqrt{n}k}$.

Alice's αk-bit message partitions the space $\{0,1\}^{nk} \times \{0,1\}^n$ into $2^{\alpha k}$ equivalence classes. Let

C be a largest equivalence class, w.r.t distribution μ'. Since the protocol is deterministic, for every input $((x_1, \ldots, x_k), y) \in C$ and $i \in \{1, \ldots, k\}$, the protocol answers correctly.

Since Bob and Charlie communicate at most c bits on each input, then, for any $((x_1, \ldots, x_k), y) \in C$, we can output the entire vector of answers using $ck < \beta k\sqrt{n}$ bits of communication. Thus, the protocol outputs a correct answer for $\text{DISJ}^{(k)} : \{0,1\}^{k\times k} \times \{0,1\}^{k\times k} \to \{0,1\}^k$ on all inputs from C. Since C was chosen to be the largest equivalence class w.r.t distribution μ', the protocol is correct with probability at least $2^{-\alpha k}$ over distribution μ. This, along with Yao's min-max theorem [20], contradicts THEOREM 5.1, the direct product theorem for set disjointness.

This lower bound matches our upper bound to within factors of $\log n$ and $\log k$. A similar lower bound can also be obtained for any Boolean function that has a strong direct product theorem.

5.2 Lower Bounds via Compression In all our upper bounds, there is limited interaction between Bob and Charlie. We formalize this as follows: First, as usual, Alice sends m bits $A = A(x_1, \ldots, x_k, y)$ to Bob. Then, π is a 1.5 round (m, ℓ, q)-protocol if there are two rounds of communication between Bob and Charlie that satisfy the following: In the first round, Bob communicates ℓ bits $B = B(y, A)$ to Charlie that does not depend on i (equivalently Bob could forward ℓ bits of Alice's message to Charlie). In the second round, Charlie communicates q bits $C = C(x_1, \ldots, x_k, i, B)$ back to Bob. Finally, Bob, determines the answer $f(x_i, y)$, from his knowledge of y, i, A, and C. Note that there is no restriction on Alice's advice, except that she sends $o(k)$ bits. The crucial restriction, beyond the fact that there are only two rounds of communication after Alice's advice, is that Bob's communication to Charlie is independent of i. In that sense, he is engaging in only half a round of communication.

Interestingly, 1.5 round protocols have non-trivial power. The proof of THEOREM 3.3, that refutes Pătraşcu's conjecture in a strong way, employs just a 1.5 round $(O(\log^2 n), O(\log^2 n), 1)$-protocol. For set-disjointness, we gave a 1.5 round $\big(\tilde{O}(\sqrt{n}), \tilde{O}(\sqrt{n}), \tilde{O}(\sqrt{n})\big)$-protocol, where $\tilde{O}$ ignores polylogarithmic factors. 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 5.3. *For every 1.5 round (m, ℓ, q)-protocol for computing $SEL^{k\times 1}_{DISJ}$, we have $\ell \cdot q \geq \frac{n}{35}$ provided $k \geq 300n(n+m)$.*

Theorem 5.3 is tight as it matches the upper bound provided by Theorem 1.2. The lower bound here is incomparable to Theorem 5.2 as we do not restrict Alice's advice like it does. Our next lower bound is for the well known inner-product function, denoted by IP, where $\text{IP}(x, y) = \sum_{i=1}^{n} x_i y_i \bmod 2$. The inner product function is one of the hardest functions in the standard two-party communication model.

THEOREM 5.4. *Let $\alpha > 2$ be some constant. For every 1.5 round (m, ℓ, q)-protocol for computing $SEL^{k\times 1}_{IP}$, we have $(\ell + q) \geq \frac{n(\alpha-2)}{\alpha}$ provided $k \geq \alpha(n+m)$.*

The main idea in proving both of the above theorems is to find an encoding of Charlie's input $x_1, \ldots, x_k$ using 1.5 round protocols. If the cost of the protocol is small, our encoding *compresses* kn bits of information to fewer bits. On the other hand, this is impossible as $x_1, \ldots, x_k$ has entropy kn, yielding a contradiction. This idea can be quite cleanly carried out for the inner-product function and so we begin by proving Theorem 5.4. Implementing the compression idea for disjointness is more involved and so we show it later.

Proof. [of Theorem 5.4] We assume for now that we are given a deterministic protocol π contradicting the theorem. Our goal is to give a scheme for encoding $x_1, \ldots, x_k$, where each x_i is a uniformly chosen n bit vector. Consider any input $x_1, \ldots, x_k$ of Charlie. By averaging, there exists a message M_{fixed} that Bob sends Charlie 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 M_{fixed} on each of them. Our encoding of $x_1, \ldots, x_k$ then contains the following.

- The set $\mathcal{Y}$, using $(n - \ell) \cdot n$ bits.
- Bob's fixed message M_{fixed}, using ℓ bits.
- Alice's message $A(x_1, \ldots, x_k, y)$ for each $y \in \mathcal{Y}$ (using $(n - \ell) \cdot m$ bits).
- For each index $i \in [k]$ and each $y \in \mathcal{Y}$, the q bit message $C(x_1, \ldots, x_k, M_{fixed}, i)$ sent from Charlie to Bob (Uses only kq bits, because $x_1, \ldots, x_k$ to be encoded is fixed and M_{fixed} is the same for each $y \in \mathcal{Y}$.)
- Extra information E consisting of the inner product of each x_i with a set $\mathcal{Y}'$ of ℓ more linearly independent vectors such that $\mathcal{Y} \cup \mathcal{Y}'$ forms a basis of the

vector space $\{0,1\}^n$. Note that for each partial basis $\mathcal{Y}$, we choose $\mathcal{Y}'$ in some pre-determined way. Encoding E thus takes only $k \cdot \ell$ bits.

The decoding follows simulating the player Bob in protocol π for each i and each $y \in \mathcal{Y}$. Given any such encoding of $x_1, \ldots, x_k$, the decoder can simulate Bob since he knows y, i, Alice's message $A(x_1, \ldots, x_k, y)$ and Charlie's message $C(x_1, \ldots, x_k, M_{fixed}, i)$. Hence, he learns the answer $IP(x_i, y)$ because π gives the correct answer. From this and the inner products in E, the decoder learns all of $x_1, \ldots, x_k$ by solving a simple system of linear equations of full rank. Because no encoding of $x_1, \ldots, x_k$ can use less than its entropy $H(x_1, \ldots, x_k)$, we have the following: $(n-\ell) \cdot n + \ell + (n-\ell) \cdot m + qk + \ell k \geq nk$. Thus, $k(q+\ell) \geq k\big(n - \frac{(n-\ell)(n+m)+\ell}{k}\big)$ Recalling that the theorem requires $k \geq \alpha(n+m)$, we are done.

The idea above can be used to obtain lower bounds for set-disjointness as well but requires more work. The main source of complication is the following: knowing the inner-product of x_i with a non-zero known vector y provides 1 bit of information about x_i. This is not quite true for set-disjointness. If $DISJ(x_i, y) = 0$, then not much can be learned about x_i. On the other hand, if $DISJ(x_i, y) = 1$, then we learn that indices at which y has a 1, are indices where x_i has zero. Thus, in order to encode $x_1, \ldots, x_k$ efficiently, we would like to choose a convenient set of y's such that for many i's, $DISJ(x_i, y) = 1$. Unfortunately, if we choose a vector x and y at random, with very high probability $DISJ(x, y) = 0$. Hence, we work with a restricted set of vectors. Let Γ_x be the set of n bit vectors that have exactly σ_x many 1's. Similarly, let Γ_y be the set of n bit vectors with σ_y many 1's. It will be convenient for us to set $\sigma_x \times \sigma_y \leq 0.4n$ and $\sigma_y = m/0.2$. Our setting of these parameters, together with the following fact, ensures that a 0.85 fraction of vectors in Γ_y intersect with any given vector in Γ_x.

FACT 5.1. *For each $x \in \Gamma_x$, if y is chosen at random from Γ_y, then $\Pr_y\big[DISJ(x,y) = 1\big] \geq exp\big(-\frac{4\sigma_x\sigma_y}{n}\big)$, if $\sigma_x, \sigma_y \leq \frac{1}{4}n$.*

Let $Cover(x_i, y) \subseteq [k] \times [n]$ denote the indices of x_i that one learns are zero from learning $DISJ(x_i, y)$, namely if $DISJ(x_i, y) = 1$, then $Cover(x_i, y) = \{(i,j) \,|\, y_j = 1\}$ and if $DISJ(x_i, y) = 0$, then $Cover(x_i, y) = \emptyset$. Similarly, for any $\mathcal{Y} \subseteq \Gamma_y$, let $Cover((x_1, \ldots, x_k), \mathcal{Y}) = \cup_{i \in [k]} \cup_{y \in \mathcal{Y}} Cover(x_i, y)$.

The main lemma that enables efficient encoding of $x_1, \ldots, x_k$ is the following:

LEMMA 5.1. *Consider any deterministic 1.5 round (ℓ, q)-protocol, such that $\ell \leq .2\sigma_y$. For each $x_1, \ldots, x_k$, there exists a message B_{fixed} of Bob and a set $\mathcal{Y} \subseteq \Gamma_y$ of size at most $30n$ such that:*
1) $\forall y \in \mathcal{Y}$, Bob's message on y is B_{fixed}.
2) $|Cover((x_1, \ldots, x_k), \mathcal{Y})| \geq \frac{1}{2}nk$.

Proving this lemma needs technical work. Before we do that, let us see how the lower bound for disjointness follows from Lemma 5.1 via compression:

Proof. [of Theorem 5.3] Given any $x = x_1, \ldots, x_k$ and any 1.5 round (ℓ, q)- protocol for $\text{SEL}^{k \times 1}_{DISJ}$ in which Alice provides m bits of advice, our encoding of $x_1, \ldots, x_k$ contains the following.

- The set $\mathcal{Y}$ from Lemma 5.1 (using $30n \cdot n$ bits).
- Bob's fixed message B_{fixed} (using $\ell = .2\sigma_y$ bits).
- Alice's messages $A(x_1, \ldots, x_k, y)$ for each $y \in \mathcal{Y}$ (using $30n \cdot m$ bits).
- For each $i \in [k]$ and each $y \in \mathcal{Y}$, the q bit message $C(x_1, \ldots, x_k, B_{fixed}, i)$ sent from Charlie to Bob. (Using only kq bits, because $x_1, \ldots, x_k$ to be encoded is fixed and B_{fixed} is the same for each $y \in \mathcal{Y}$.)
- The remaining information E about $x_1, \ldots, x_k$ that is not learned from $Cover((x_1, \ldots, x_k), \mathcal{Y})$. As $|Cover((x_1, \ldots, x_k), \mathcal{Y})| > 0.5nk$, there are at most $\binom{0.5nk}{\sigma_x k}$ possibilities left for placing the ones in $x_1, \ldots, x_k$. This can be transmitted in $e = \log\big(\binom{0.5nk}{\sigma_x k}\big)$ bits.

The decoding follows Bob's protocol for each i and each $y \in \mathcal{Y}$. Given any such encoding of $x_1, \ldots, x_k$, the decoder can simulate Bob as needed he knows y, i, Alice's message $A(x_1, \ldots, x_k, y)$ and Charlie's response. Hence, he learns the answer $DISJ(x_i, y)$ from the protocol for each $y \in \mathcal{Y}$ and thus determines the indices in $Cover((x_1, \ldots, x_k), \mathcal{Y})$ where $x_1, \ldots, x_k$ has zeroes. By definition, E communicates the remaining information about $x_1, \ldots, x_k$ and so we correctly decode.

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

$$30n \cdot n + .2\sigma_y + 30n \cdot m + qk + H(E) \geq H(x_1, \ldots, x_k).$$

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

$$H(x_1, \ldots, x_k) - H(E) = \log\left(\binom{n}{\sigma_x}^k \Big/ \binom{0.5nk}{\sigma_x k}\right).$$

Using Sterling's approximation for $\sigma_x < n/4$, it is not hard to verify $H(x_1, \ldots, x_k) - H(E) \sim \sigma_x k$. Also, the

theorem requires $k \geq 300n(n+m)$, so that $30n \cdot n + .2\sigma_y + 30n \cdot m \leq .1\sigma_x k$. This leaves, $qk \geq .9\sigma_x k$, giving the result $q \geq .9\sigma_x = .9 \cdot \frac{.16n}{\sigma_y} \geq$. Now recalling, that we set $\sigma_y = m/0.2$, we get $q \geq .9 \cdot \frac{.16n \cdot .2}{m} \geq \frac{n}{35m}$.

All that remains is to prove that the set $\mathcal{Y}(x_1, \ldots, x_k)$ promised by Lemma 5.1, exists for each $x_1, \ldots, x_k$. We will do it using the probabilistic method, in two stages. 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 5.2. *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 $y \in \mathcal{Y}_0(x_1, \ldots, x_k)$ satisfying the following conditions:*

- $|\mathcal{Y}_0(x_1, \ldots, x_k)| \geq 10 \cdot .8^{\sigma_y} \cdot |\Gamma_y|$.
- *Let $I(x_1, \ldots, x_k) = \{i \mid \Pr_{y \in \mathcal{Y}_0}\left[DISJ(x_i, y) = 1\right] > 0.2\}$. Then, $|I(x_1, \ldots, x_k)| > 0.7k$.*

Before we prove Lemma 5.2, let us show why such a $\mathcal{Y}_0$ 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 of the elements of this subset is fairly well spread out among indices in $[n]$. For any such set $S \subset \Gamma_y$, let $C(S) = \left\{i \mid \Pr_{y \in S}[y_i = 1] \leq \frac{1}{2n}\right\}$.

LEMMA 5.3. *Let $|S| > 2 \cdot .8^{\sigma_y} \cdot |\Gamma_y|$. Then, $\left|C(S)\right| \leq .2n$.*

Proof. Suppose that $.2n < |C(S)| \leq n$. Choose a random $y \in C(S)$. By definition, the probability that y has a one in some index in $C(S)$ is at most $|C(S)| \cdot \frac{1}{2n} \leq \frac{1}{2}$. Hence, $|S'| \geq \frac{1}{2}|S|$, where S' denotes the set of ys in S such that $y_j = 0$, for all $j \in C(S)$. This leaves at most $n - |C(S)| < .8n$ locations for the σ_y ones that are in each such y. Hence, $|S'| \leq \binom{.8n}{\sigma_y} \leq (.8)^{\sigma_y} \cdot \binom{n}{\sigma_y}$ giving the contra-positive of the result.

We now show that Lemma 5.3 and Lemma 5.2 can be combined to get our desired set $\mathcal{Y}(x_1, \ldots, x_k)$, proving Lemma 5.1.

Proof. [of Lemma 5.1] We pick $\mathcal{Y}_0(x_1, \ldots, x_k)$ according to Lemma 5.2. Construct a subset $\mathcal{Y}$ by independently choosing $30n$ elements at random from $\mathcal{Y}_0$. We show that $\text{Exp}_{\mathcal{Y}}[|Cover((x_1, \ldots, x_k), \mathcal{Y})|] \geq \frac{1}{2}nK$. For any $i \in [k]$, let $\mathcal{Y}^i$ be the set of $y \in \mathcal{Y}_0$ such that $DISJ(x_i, y) = 1$. Recall, from Lemma 5.2, for each $i \in I(x_1, \ldots, x_k)$, we have $|\mathcal{Y}^i| > 0.2|\mathcal{Y}_0| \geq 2 \cdot .8^{\sigma_y} \cdot |\Gamma_y|$. Hence, by Lemma 5.3, $\left|C(\mathcal{Y}^i)\right| \leq .2n$. Hence, for any $i \in I(x_1, \ldots, x_k)$ and $j \notin C(\mathcal{Y}^i)$, $\Pr_{y \in \mathcal{Y}_0}\left[(i,j) \in Cover(x_i, y)\right] = \Pr\left[y \in \mathcal{Y}^i\right] \cdot \Pr\left[y_j = 1 \,|\, y \in \mathcal{Y}^i\right] \geq 0.2 \cdot \frac{1}{2n} = \frac{.1}{n}$. But we independently choose $30n$ different y to be in $\mathcal{Y}$. Hence, $\Pr_{\mathcal{Y}}[(i,j) \notin Cover(x_i, \mathcal{Y})] \leq \left(1 - \frac{.1}{n}\right)^{30n} \leq \frac{1}{e^3}$. We conclude that $\text{Exp}_{\mathcal{Y}}[|Cover((x_1, \ldots, x_k), \mathcal{Y})|] \geq \sum_{i \in I(X)} \sum_{j \notin C(\mathcal{Y}^i} \Pr_{\mathcal{Y}}[(i,j) \in Cover(x_i, \mathcal{Y})] \geq .9 \cdot (1 - .2)k \cdot (1 - .2)n \cdot \left(1 - \frac{1}{e^3}\right) \geq .5nK$. Thus, there exists such a set $\mathcal{Y}$.

All that remains is to establish Lemma 5.2, which we do below.

Proof. [of Lemma 5.2] Choose a random message B_{fixed} sent by Bob by choosing a random $y \in \Gamma_y$ and letting $B_{fixed} = B(A(x_1, \ldots, x_k, y), y)$. Let $\mathcal{Y}_0$ denote the set of y's leading Bob to send B_{fixed} when Charlie has $x_1, \ldots, x_k$. The first thing to show is the following:

LEMMA 5.4. $Exp_{B_{fixed}}\left[\left|\{i \in [k] : \Pr_{y \in \mathcal{Y}_0}[DISJ(x_i, y) = 1] \leq .2\}\right|\right] \leq .2k$.

Proof. Consider any $i \in [k]$. Choose a random B_{fixed}, where the probability of choosing each message is proportional to the number of y's for which this message is sent by Bob, when Charlie gets $x_1, \ldots, x_k$. Let $\mathcal{Y}_0$ be the set of y's for which B_{fixed} is sent by Bob. Let D be the event that $\Pr_{y \in \mathcal{Y}_0}\left[DISJ(x_i, y) = 1\right] \leq .2$. Let $\Pr[D] = a$. We will upper bound a by computing $\Pr_{y \in \Gamma_y}\left[DISJ(x_i, y) = 1\right]$ in two ways. First, note that choosing a y at random from Γ_y is the same as first choosing a random B_{fixed} and then choosing $y \in \mathcal{Y}_0$ at random. Hence, $\Pr_{y \in \mathcal{Y}_0}\left[DISJ(x_i, y) = 1\right] = \Pr[D] \times \Pr_{y \in \mathcal{Y}_0}[DISJ(x_i, y) = 1 \mid D] + \Pr[\neg D] \times \Pr_{y \in \mathcal{Y}_0}[DISJ(x_i, y) = 1 \mid \neg D] \leq a \times .2 + (1 - a) \times 1$. On the other hand, directly computation using Fact 5.1 and recalling $\sigma_x \cdot \sigma_y = 0.4n$, yields $\Pr_{y \in \Gamma_y}\left[DISJ(x_i, y) = 1\right] = \geq \frac{1}{e^{.16}} = .85$.

Combining these two bounds for the same thing gives $0.85 \leq a \times .2 + (1 - a) \times 1$ giving that $a \leq .2$.

This is true for all $i \in [k]$. The lemma now follows from the linearity of expectation.

Thus, using Markov's inequality with the lemma above, $\Pr_{B_{fixed}}\left[|I(x_1, \ldots, x_k)| < 0.7k\right] < 2/3$. We are almost done now. The total number of Bob's messages is at most 2^{ℓ}. Hence, $\Pr_{B_{fixed}}\left[|\mathcal{Y}_0| < \frac{1}{4}2^{-\ell}|\Gamma_y|\right] < 1/4$. As $2/3 + 1/4 < 1$, with non-zero probability neither $\mathcal{Y}_0$ nor $I(x_1, \ldots, x_k)$ have smaller than their desired sizes respectively. Recall, $\ell = .2\sigma_y$. Hence, $\frac{1}{4}2^{-\ell}|\Gamma_y| > \frac{1}{4}(0.87)^{\sigma_y}|\Gamma_y|$. Observing that there exists some constant c, such that for $\sigma_y > c$, $\frac{1}{4}(0.87)^{\sigma_y}|\Gamma_y| > 10 \cdot (0.8)^{\sigma_y}|\Gamma_y|$, we are done.

This completes the proof of the lower bound for set-disjointness, i.e Theorem 5.3.

6 Open Problems and Conclusions

The $A \xrightarrow{B} (B \leftrightarrow C)$ model is a new variant of the communication complexity model that may be useful for studying the complexity of many dynamic data structure problems. Pătraşcuconjectured that for any hard two-player function f, the asymmetric version of f is hard in the $A \xrightarrow{B} (B \leftrightarrow C)$ model. In this paper, we have obtained surprising counterexamples to this conjecture: we have exhibited a function with maximal two-player complexity that is easy in the $A \xrightarrow{B} (B \leftrightarrow C)$ model, and have also shown nontrivial upper bounds for set disjointness in the $A \xrightarrow{B} (B \leftrightarrow C)$ model.

The most important unresolved question is the exact complexity of asymmetric set disjointness in the $A \xrightarrow{B} (B \leftrightarrow C)$ model. It is still possible that $\mathrm{SEL}_{DISJ}^{k\times 1}$ 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 $\mathrm{SEL}_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ model are presently known for any function, even via a non-constructive argument.

One intuition that we have relates the complexity of $\mathrm{SEL}_f^{k\times 1}$ 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 $\mathrm{SEL}_{DISJ}^{k\times 1}$.

Motivated by this intuition, we conjecture that for any function f, the worst-case instances of $\mathrm{SEL}_f^{k\times 1}$ 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 $\mathrm{SEL}_f^{k\times 1}$ in the $A \xrightarrow{B} (B \leftrightarrow C)$ game. Thus, for asymmetric set disjointness, we conjecture a $\sqrt{n}$ lower bound in the $A \xrightarrow{B} (B \leftrightarrow C)$ 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 is also supported by a postdoctoral fellowship of the Ontario Ministry of Research and Innovation.

References

[1] S. Alstrup, T. Husfeldt, and T. Rauhe. Marked ancestor problems. In *39th IEEE Annual 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 *27th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 337–347, 1986.

[3] 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.

[4] S. Ben-David, N. Eiron, and H. U. Simon. Limitations of learning via embeddings in Euclidean half spaces. *J. Mach. Learn. Res.*, 3:441–461, 2003.

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

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

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

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

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

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

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

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

[13] N. Linial and A. Shraibman. Learning complexity vs. communication complexity. In *23rd IEEE Conf. on Computational Complexity*, pages 53–63, 2008.

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

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

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

[17] M. Pătraşcu and E. D. Demaine. Logarithmic lower bounds in the cell-probe model. *SIAM J. Comput.*, 35(4):932–9–63, 2006.

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

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

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

[21] A. C. C. Yao. Some complexity questions related to distributed computing. In *11th ACM Symposium on Theory of Computing (STOC)*, pages 209–213, 1979.