Electronic Colloquium on Computational Complexity, Report No. 197 (2023)

Optimal Multi-Pass Lower Bounds for MST in Dynamic Streams

Sepehr Assadi[*] Gillat Kol[†] Zhijun Zhang[‡]

Abstract

The seminal work of Ahn, Guha, and McGregor in 2012 introduced the graph sketching technique and used it to present the first streaming algorithms for various graph problems over *dynamic* streams with both insertions and deletions of edges. This includes algorithms for cut sparsification, spanners, matchings, and minimum spanning trees (MSTs). These results have since been improved or generalized in various directions, leading to a vastly rich host of efficient algorithms for processing dynamic graph streams.

A curious omission from the list of improvements has been the MST problem. The best algorithm for this problem remains the original AGM algorithm that for every integer $p \geqslant 1$, uses $n^{1+O(1/p)}$ space in p passes on n-vertex graphs, and thus achieves the desired semi-streaming space of $\widetilde{O}(n)$ at a relatively high cost of $O(\frac{\log n}{\log\log n})$ passes. On the other hand, no lower bounds beyond a folklore one-pass lower bound is known for this problem.

We provide a simple explanation for this lack of improvements:

The AGM algorithm for MSTs is optimal for the entire range of its number of passes!

We prove that even for the simplest *decision* version of the problem—deciding whether the weight of MSTs is at least a given threshold or not— any p-pass dynamic streaming algorithm requires $n^{1+\Omega(1/p)}$ space. This implies that semi-streaming algorithms do need $\Omega(\frac{\log n}{\log\log n})$ passes.

Our result relies on proving new **multi-round** communication complexity lower bounds for a variant of the *universal relation* problem that has been instrumental in proving prior lower bounds for *single-pass* dynamic streaming algorithms. The proof also involves proving new composition theorems in communication complexity, including majority lemmas and multi-party XOR lemmas, via information complexity approaches.

[*] (`sepehr@assadi.info`) Cheriton School of Computer Science, University of Waterloo, and Department of Computer Science, Rutgers University.

[†] (`gillat.kol@gmail.com`) Department of Computer Science, Princeton University.

[‡] (`zhijunz@princeton.edu`) Department of Computer Science, Princeton University.

ISSN 1433-8092

Contents

1 Introduction

In the **dynamic graph streaming** model, we have a (possibly edge-weighted) graph $G = (V, E)$ with vertices $V := \{1, 2, \ldots, n\}$, whose edges and their weights are being defined by a sequence of insertions and deletions in a stream $\sigma := (\sigma_1, \sigma_2, \ldots, \sigma_N)$; here, N is the length of the stream which is typically assumed to be $\text{poly}(n)$. Each entry σ_i is either of the form $(u_i, v_i, w_i, +)$ for $u_i, v_i \in V$ and $w_i \in \mathbb{N}$ and is interpreted as the edge (u_i, v_i) with weight $w(u_i, v_i) = w_i$ being inserted to E, or $(u_i, v_i, w_i, -)$ which means the edge (u_i, v_i) with the given weight w_i is being deleted. We are guaranteed that the stream does not delete an edge which is not inserted, does not insert an edge more than once before deleting it in the middle, and that the weight of a deleted edge matches its weight at the time of insertion[1]. The goal is to make one or a few sequential passes over the stream σ, use a limited memory—ideally, $\widetilde{O}(n) := O(n \cdot \text{polylog}(n))$ bits, referred as the **semi-streaming space**—and compute an answer to the given problem on G at the *end* of the last pass.

Dynamic streams (not necessarily for graphs) have been studied extensively in the streaming literature since the introduction of the model in [AMS96], e.g., for statistical estimation problems [CCF02] or geometric problems [FIS05]. However, despite the significant attention graph streams have received since their introduction in [FKM+05], *dynamic* graph streams were not studied for quite some time due to lack of any techniques for addressing problems in this domain.

This state-of-affairs was entirely changed by a seminal work of Ahn, Guha, and McGregor (henceforth, AGM) [AGM12a] who introduced the *graph sketching* technique and used it to devise dynamic graph streaming algorithms for several fundamental problems, including connectivity, minimum spanning trees, cut sparsifiers, and matchings. This immediately led to a flurry of results on dynamic graph streaming algorithms, *all* using the graph sketching technique[2], that either improved upon [AGM12a] or extended its results to various other problems; see, e.g., [AGM12b, AGM13, KLM+14, BHNT15, CCHM15, MTVV15, GMT15, ACG+15, BS15, AKLY16, CCE+16, HP16, FKN21] and references therein.

One of the very few problems that saw *zero* improvement since [AGM12a] is the *minimum spanning tree (MST)* problem. [AGM12a] designed a dynamic streaming algorithm that for every integer $p \geqslant 1$, with high probability, finds an MST of the input graph using $n^{1+O(1/p)}$ space and p passes. Specifically, this leads to an $O(\frac{\log n}{\log\log n})$-pass semi-streaming algorithm. No better algorithms have been designed for this problem yet, despite the fact that in *insertion-only* streams, a simple single-pass semi-streaming algorithm has already been known since [FKM+05].

We provide a simple explanation for this lack of improvements:

The AGM algorithm for MSTs is optimal for the entire range of its number of passes!

Specifically, semi-streaming algorithms for MSTs require $\Omega(\frac{\log n}{\log\log n})$ passes. Beside settling the complexity of the fundamental MST problem in the semi-streaming model, this also constitutes one of the strongest separations between the power of insertion-only streams and dynamic graph streams; see, e.g. [AKLY16, DK20] that prove such separations only between single-pass algorithms (for the approximate matching problem).

[1] In particular, no "partial updates" to the edge weights are allowed and the stream needs to delete the edge "fully" first (and provide its weight) and then re-inserts it possibly with another weight; see [CKL22] for more details on this.

[2] The results in [LNW14, AHLW16] show that this is not a coincidence: any dynamic graph streaming algorithms that can handle triply-exponential long streams and doubly-exponential edge-multiplicities (in the middle of the stream), can be turned into a graph sketch. While these restrictions seem quite strong, almost all known graph streaming algorithms can handle such inputs as well. However, in this work, we will *not* rely on this characterization.

1.1 Our Contributions

We now discuss our contributions in more detail. Our main result establishes the optimality of the MST algorithm of [AGM12a].

Result 1. *For any integer $p = o(\frac{\log n}{\log\log n})$, any p-pass dynamic streaming algorithm on n-vertex graphs requires $\tilde{\Omega}(n^{1+\frac{1}{2p-1}})$ space to solve the minimum spanning tree problem with constant probability. The lower bound applies even if the edge weights and the length of the stream are both at most $O(n^2)$ and the algorithm only needs to decide whether the weight of minimum spanning trees is at least a given threshold.*

Prior to our work, no lower bounds were known for the MST problem in dynamic streams beside a single-pass lower bound of $\Omega(n^2)$ space[3]. Another immediate corollary of our result is a strong limitation on the power of the graph sketching technique. While graph sketching has been extremely successful for problems such as cut- or spectral-sparsification [AGM12b, AGM13, KLM$^+$14], it appears to be quite weak for the MST problem, even when allowed "many" rounds of adaptive sketching.

It is worth mentioning that our lower bound indeed only holds for *exact* MSTs. For the relaxed version of the problem, wherein the goal is to obtain a $(1+\varepsilon)$-approximation instead, [AGM12a] already presents a single-pass semi-streaming algorithm. On the other hand, we prove our lower bound for exact MSTs for the algorithmically easiest *decision* version of the problem: given a threshold at the beginning of the stream, decide whether the weight of MSTs is at least as large as this threshold or not. It is also worth mentioning that many problems admit provable separations between their search versus decision variants in the dynamic streaming model; see, e.g. [AKL17] for an example of a separation for finding approximate matchings versus estimating the size of the largest matchings via single-pass algorithms (or in [AKL16] for the streaming set cover problem).

Our techniques. Result 1 relies on proving a new *multi-round* communication complexity lower bound for a *non-standard composition* of a variant of the **Universal Relation (UR)** problem. UR has been instrumental in proving prior lower bounds for *single-pass* dynamic streaming algorithms [JST11, KNP$^+$17, NY19] (see also [Yu21]). In this problem, there is a universe U of m elements; Alice receives a set $A \subseteq U$ and Bob receives a proper subset $B \subset A$. The communication is only from Alice to Bob. Prior work has shown that in order for Bob to output *any* element from $A \backslash B$, Alice needs to communicate $\Omega(\log^2 m)$ bits to succeed with constant probability [JST11] or $\Omega(\log^3 m)$ bits for high probability [KNP$^+$17].

We start by proving that any r-round protocol—wherein Alice and Bob can communicate back and forth at most r times—for outputting the *smallest* element in $A \backslash B$ (as opposed to outputting any one) requires $\Omega_r(m^{1/r})$ communication. We can then combine this with standard direct-sum arguments in communication complexity (see, e.g. [BBCR13]) to obtain that solving m *independent* copies of this problem requires $\Omega_r(m^{1+1/r})$ communication. We then show how to reduce this to the problem of *finding* MSTs in dynamic streams and prove a lower bound for the latter problem as well. This lower bound however does not extend to the decision problem (which is a common occurrence for other "direct-sum UR-type" reductions, e.g., in [NY19] and [Yu21]).

As we will explain in Section 3, to be able to extend the lower bound to the decision problem, the key ingredients used in our proof are:

[3] To our knowledge, this lower bound appears to have been folklore and we do not know a reference for it.

Direct sum with "hint". At a high level, we will be dealing with a direct sum of a carefully defined variant of pointer chasing problems on trees. It differs from typical direct-sum arguments in that the reduction to MST demands knowing the sum of outputs of all copies, which *correlates* the copies. Our direct-sum result is obtained by directly carrying this extra bit of knowledge, named *hint*, throughout the proof.

Majority vs. XOR. It turns out the most straightforward approach, which guesses the hint and conducts a typical direct-sum argument without the hint, can never work as it involves lower bounding majority computation of multiple copies with super low advantage. Simple coin toss examples will show that such a result is impossible. We work around this by a connection between majority computation with high advantage and XOR computation with low advantage. It enables us to utilize direct-sum results for XOR computation instead.

Multi-party XOR lemma. As known results are not strong enough for proving the optimal pass lower bound, we devise a multi-party XOR lemma, mimicking the 2-party version of [Yu22], that improves the dependence of communication in the number of rounds, while leading to a worse advantage decay. In particular, suppose each of k pairs of 2 parties are given n/k instances of a boolean function f, and they want to jointly solve the n-fold XOR of all n instances. We prove the following result which may be of independent interest.

Result 2. *If any r-round, 2-party protocol that solves f with constant probability, requires C communication, then any r-round, $2k$-party protocol that solves the n-fold XOR of f with probability $\frac{1}{2} + (\frac{1}{2})^{\Omega(\frac{k}{r})}$, requires $\Omega(\frac{n}{k} \cdot (\frac{C}{r} - O(r)))$ communication.*

The rest of this paper is organized as follows. Section 3 provides a sketch of our proof in more detail. Then we prove a suboptimal pass lower bound in Section 4 using the 2-party XOR lemma of [Yu22]. Our multi-party XOR lemma is presented in Section 5 and used to obtain the full version of our main result.

2 Preliminaries

Notation. For an integer $n \in \mathbb{N}$, $[n]$ is used as a shorthand for the set $\{1, \ldots, n\}$. For a tuple $X = (X_1, \ldots, X_n)$, we write $X_{\leqslant i} = (X_1, \ldots, X_i)$. Similarly, we have $X_{\geqslant i}$ and $X_{<i}, X_{>i}$. We also use $X_{-i} = (X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$. The XOR operation is denoted by $\oplus$.

Throughout this paper, sans-serif letters are reserved for random variables (e.g. X) while normal letters are used for realizations of the corresponding random variables (e.g. x, X). For random variables X, Y, we denote the *Shannon entropy* of X by $\mathbb{H}(\mathsf{X})$, the *mutual information* between X, Y by $\mathbb{I}(\mathsf{X}\,;\mathsf{Y})$, the *KL-divergence* between X, Y by $\mathbb{D}(\mathsf{X} \,||\, \mathsf{Y})$, and the *total variation distance* between X, Y by $\|\mathsf{X} - \mathsf{Y}\|_{\mathrm{tvd}}$. Appendix A provides necessary background on information theory, including the basic tools used in this paper.

Dynamic graph streaming. For a dynamic graph streaming problem, the input is a sequence of insertions and deletions of edges in an underlying graph, initially empty. In every pass of an algorithm, it processes the operations, one at a time, in the given order. At the end of the algorithm, it answers some query about the constructed graph resulting from all insertions and deletions. Only the space requirement between operations is considered in this paper (i.e., unlimited memory is allowed while processing each operation). We are interested in the problem $\mathbf{MST}_n$, which asks whether the weight of minimum spanning trees of an n-vertex graph is at least a given threshold.

Communication model. For the standard 2-party communication model, we assume Alice sends the first message and the receiver of the last message returns the output. Let $\mathbf{CC}(\pi)$ denote the *communication complexity* of a protocol π, and $\mathbf{CC}^{(i)}(\pi)$ the communication complexity of the i-th round of π. We also use $\mathbf{IC}(\pi)$ to denote the *internal information cost* of π. The *distributional complexity* of f, denoted by $\mathbf{D}^{(r)}_{\mu,\epsilon}(f)$, is defined as the infimum communication complexity of any r-round protocol solving f with probability ϵ over μ.

The multi-party communication model we use in this paper is formally defined as follows. There are $2k$ parties named Alice $1, \ldots, k$ and Bob $1, \ldots, k$. Each Alice has an input from $\mathcal{X}$ and each Bob has an input from $\mathcal{Y}$. There is a *blackboard*, initially empty, visible to all parties. The parties proceed in the *circular* order of Alice $1, \ldots, k$ and Bob $1, \ldots, k$, starting with Alice 1. In one's turn, it computes a message given its input as well as the current blackboard, and posts the message to the blackboard. At the end of the protocol, the last party returns an output (and does not post a message to the blackboard). The communication complexity is defined as the length of the final blackboard. The number of rounds is defined as the total number of times Alice k and Bob k post messages to the blackboard. (So, e.g., a 1-round protocol in general consists of Alice $1, \ldots, k$ and Bob $1, \ldots, k-1$ posting one message each, and Bob k returning an output.) In a randomized protocol, each party is allowed to use both public randomness, shared by all parties, and private randomness, known only to itself. The goal is to compute a function g over $\mathcal{X}^k \times \mathcal{Y}^k$. We similarly define the distributional complexity of g in the $2k$-party model and denote it by $\mathbf{D}^{(r),k}_{\mu,\epsilon}(g)$, where μ is a distribution over $\mathcal{X}^k \times \mathcal{Y}^k$. It can be verified that the multi-party model for $k = 1$ coincides with the standard 2-party model. Moreover, $\mathbf{D}^{(r),1}_{\mu,\epsilon}(\cdot) = \mathbf{D}^{(r)}_{\mu,\epsilon}(\cdot)$.

In this paper, we are interested in the *k-fold XOR* of a function $f : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$, defined as $f^{\oplus k}(x_1, \ldots, x_k, y_1, \ldots, y_k) = \bigoplus_{i \in [k]} f(x_i, y_i)$. We also consider the *$k$-fold majority*, denoted by $f^{\# k}$, which evaluates to 1 if $f(x_i, y_i) = 1$ for more than $\lfloor k/2 \rfloor$ indices $i \in [k]$, and 0 otherwise.

3 Technical Overview

This section serves as an outline of our proof. As a starting point, in Section 3.1, we first tackle the easier problem of proving a lower bound for the task of finding an MST solution, i.e., outputting the edges of an MST. We then proceed to identify the primary challenges in extending our technique to give a lower bound for the algorithmically easier task of computing the weight of MSTs or even for the task of deciding whether it exceeds a specified threshold. In Section 3.2, we discuss some of our initial attempts and their inherent limitations. Finally, we present the ultimate solution in Section 3.3.

3.1 The Search Version

Our hard instance. We start by outlining our lower bound for the easier task of lower bounding the space complexity of steaming algorithms that output the edges of an MST. To prove our lower bound, we design hard instances inspired by that of [NY19, Yu21], that were used to prove lower bounds for the Spanning Forest and Connectivity problems. See Figure 1 for an illustration of our hard instances. Our construction starts with a clique of size $n/2$. Edges in the clique all have the minimum possible weight, say 0. Another $n/2$ vertices are added, one at a time, as follows. For each non-clique vertex v, it is randomly connected to some vertices in the clique, with distinct, positive edge weights. Later in the stream, we remove a proper subset of the edges incident on v. Both the inserted and deleted edges follow some (non-uniform) hard distributions. The concatenation of the clique edges (of weight 0), followed by the edge insertions for all non-clique vertices, and then the edge deletions for all non-clique vertices, constitutes the entire stream.

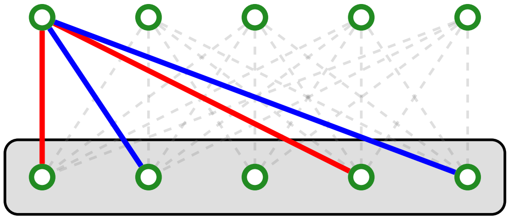


Figure 1: An illustration of hard instances for the search version of MST. Bottom vertices are fully connected. Each top vertex is connected to some bottom vertices via red edges (inserted and deleted) and blue edges (inserted but not deleted) – to avoid clutter, only edges for the first vertex are drawn.

Observe that any MST of the constructed graph must have the following structure: a spanning tree connecting the clique, plus, for each non-clique vertex, the minimum weight edge that is not deleted connecting this vertex to the clique. As a consequence, the problem of finding an MST essentially reduces to the direct sum (i.e., solving multiple copies) of the following subproblem, which we denote by $\mathbf{UR}^{\subset}_{\min}$: find the minimum element in the difference $A \backslash B$ of two sets A, B, where B is promised to be a proper subset of A.

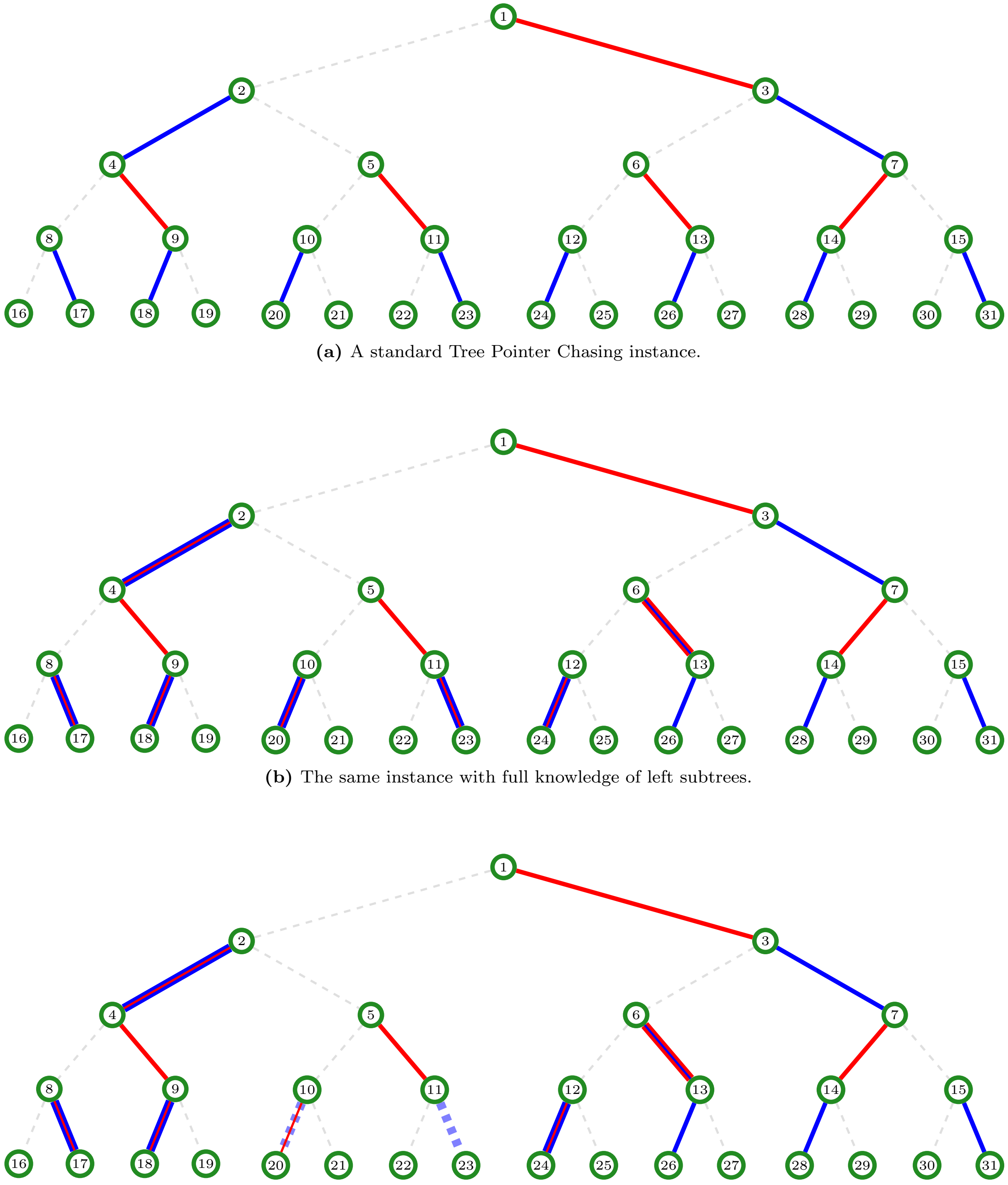
The problem $\mathbf{UR}^{\subset}_{\min}$ can be viewed as an addition to the well-studied family of *Universal Relation problems* [KRW95]. The work of [NY19] proves optimal lower bounds for Spanning Forest via one of its variants, $\mathbf{UR}^{\subset}$, in which it is sufficient to find *any* element in the difference $A \backslash B$, as opposed to finding the minimum element. In particular, [NY19] use tight results from [KNP+17] for the *one-way* communication complexity of $\mathbf{UR}^{\subset}$. However, this bound is only poly-logarithmic and therefore is too weak for our purposes. We prove that $\mathbf{UR}^{\subset}_{\min}$ is hard even with multiple rounds of communication. More specifically, we show that it admits an r vs. $\Omega_r(m^{1/r})$ round-communication tradeoff, where m is the size of the universe. Given the canonical reduction from communication to streaming, this means any direct sum/product result for bounded-round two-party communication (e.g., [JPY12, BRWY13]) suffices for lower bounding the search version of MST.

Augmented Tree Pointer Chasing. We prove the round-communication tradeoff for $\mathbf{UR}^{\subset}_{\min}$ by reduction from an "augmented" version of Pointer Chasing on trees[4]. The starting point is the well-known *Augmented Index* problem [MNSW98], in which Alice holds $x \in \{0,1\}^n$ while Bob is required to output x_i given $i \in [n]$ and $x_{<i}$. It is an "augmented" version of Index in that Bob additionally knows $x_{<i}$, i.e., everything to the left of the pointer i.

Note that Index can be viewed as Pointer Chasing on single-level trees. To generalize it to multi-level trees, recall that in the standard Tree Pointer Chasing problem, one party owns all pointers in odd levels (that is, the first party gets as input an edge going out of each node in an odd level) and the other party owns all pointers in even levels. The parties' goal is to output the unique leaf node that can be reached using the parties' pointers. See Figure 2a for an example.

A natural attempt is to additionally give the owner of each pointer full knowledge of all the left subtrees, or equivalently all pointers owned by the other party in those subtrees. In other words, if a party has, as part of its input, the pointer connecting vertex v to its i-th child, then the same party also gets all the pointers in the other party's input for the subtrees rooted at the first $i-1$ children of v. See Figure 2b for an illustration. For example, in the illustration, since Bob has the pointer connecting the root to its second child, Bob also knows all Alice's pointers in the entire left

[4] We note that $\mathbf{UR}^{\subset}_{\min}$ is introduced here only for the purpose of illustration and to provide a better context. Our proofs in Sections 4 and 5 directly deal with the augmented version of Pointer Chasing with no reference to $\mathbf{UR}^{\subset}_{\min}$. For completeness and since the lower bound for this problem may be of independent interest, we include its proof; see Corollary 4.6.



(a) A standard Tree Pointer Chasing instance.

(b) The same instance with full knowledge of left subtrees.

(c) The same instance with full knowledge of left subtrees and no knowledge of right subtrees.

Figure 2: An illustration of ATPC instances. Solid, blue edges are known to Alice and solid, red edges are known to Bob. Thick edges are owned in standard Tree Pointer Chasing while thin edges are known via augmentation. (For example, in Figure 2c, there are two *overlapping* edges from node 6 to node 13. One is red and thick, meaning that Bob owns this edge in standard Tree Pointer Chasing, and the other is blue and thin, meaning that Alice knows this edge via augmentation.) Dashed, light-colored edges are forgotten during augmentation.

subtree of the root.

Forgetting pointers. We wish to prove a lower bound for the augmented Pointer Chasing problem on trees as described above. However, we next show that there is a subtle issue. Suppose we want to prove the lower bound using the, by now standard, embedding arguments, showing that a protocol for instances with r levels implies a protocol with one less message for instances with $r-1$ levels. To do so, we sample an instance with r levels as follows. We denote by (A_j, B_j) the subinstance corresponding to the j-th subtree (of the root) of the r-level instance we are sampling. We also denote by (A', B') the input instance with $r-1$ levels that we attempt to solve. Alice and Bob publicly sample an index i and do the embedding by setting $(A_i, B_i) = (A', B')$. $A_{<i}$ is also publicly sampled (note that this standard sampling respects our augmentation). To eliminate the first round of communication, Alice and Bob publicly sample Alice's first message M_1 (conditioned on $A_{<i}$). In order to continue the simulation, the standard embedding argument would have the parties privately sample all remaining parts, namely $A_{>i}, B_{<i}, B_{>i}$. Unfortunately, $A_{>i}$ and $B_{>i}$ may not be privately sampled (roughly following their original distributions) at the same time, due to possible high correlation.

To rectify the situation, we "eliminate" $B_{>i}$ by defining the *Augmented Tree Pointer Chasing* (ATPC) problem as follows: for each pointer, the party that owns it, also (i) knows *everything* that the other party knows in subtrees to its left; and (ii) knows *nothing* in subtrees to its right. See Figure 2c for an illustration. For example, in the illustration, Alice "forgets" the pointer from node 10 because it is in a subtree to the right of the pointer from node 2.

We also emphasize that "everything that the other party knows" may not be equivalent to "all pointers owned by the other party", exactly because the other party may forget some of its originally owned pointers. To see this, consider the pointers from nodes 10 and 11 in the illustration. Before the augmentation, Alice knows both of them and Bob knows neither. As we perform the augmentation bottom-up, Bob knows the one from node 10 since it is in a subtree to the left of the pointer from node 5. Another level up, Alice forgets both of them due to the pointer from node 2. Note, however, that Bob still keeps his knowledge of the pointer from node 10. As a result, finally at the top level, Bob has the combined knowledge of both parties, including the pointer from node 10, but not the one from node 11. In other words, Bob does not know the latter even though it is also in a subtree to the left of the pointer from the root. Moreover, Bob's knowledge of the former is actually coming from himself in lower levels, but not from Alice.

A formal definition of ATPC is given in Section 4. Intuitively, the augmentation neither helps nor hurts the parties that attempt to solve an ATPC instance, as both parties should always follow the correct pointers. Indeed, we are able to prove an r vs. $\Omega_r(n^{1/r})$ round-communication tradeoff for trees with n leaf nodes, using standard information-theoretic tools.

Reducing ATPC to $\mathbf{UR}^{\subset}_{\mathbf{min}}$ and the role of augmentation. Next, we wish to show a lower bound for $\mathbf{UR}^{\subset}_{\min}$ by proving that $\mathbf{UR}^{\subset}_{\min}$ is even harder than ATPC. The reduction is as follows. Given an ATPC instance, Alice is constructing the larger set A (corresponding to insertions for MST) and Bob is constructing the smaller set B (corresponding to deletions for MST). The universe contains all the leaf nodes of the ATPC instance, sequentially ordered from left to right, and the goal is to have $\min(A \backslash B)$ being the leaf node induced by the pointers in the ATPC problem. Imagine the parties perform the construction of the sets A and B "bottom-up" in the following sense. Suppose the current pointer i is known to Alice (a similar argument applies to the case in which Bob knows the current pointer). Also, assume that the parties have already constructed $A_1, \ldots, A_w$ and $B_1, \ldots, B_w$ where w is the arity of the ATPC tree and (A_j, B_j) is the $\mathbf{UR}^{\subset}_{\min}$ instance constructed for the j-th subtree of the ATPC tree, and they want to combine all these sets to obtain A, B.

Now, we may wish for Bob to set $B = B_1 \cup \cdots \cup B_i$ and for Alice to set $A = B_1 \cup \cdots \cup B_{i-1} \cup A_i$, as this would imply $A \backslash B = A_i \backslash B_i$, while the promise that $B \subseteq A$ in the definition of $\mathbf{UR}^{\subset}_{\min}$ is satisfied. Note that Alice can indeed compute this set A *thanks to the augmentation* that gives her $B_1, \cdots, B_{i-1}$. In fact, this is the exact reason for the augmentation. Unfortunately, though, Bob cannot compute B as he does not know i. Nevertheless, it can be easily remedied by setting $A = B_1 \cup \cdots \cup B_{i-1} \cup A_i \cup A'$ and $B = B_1 \cup \cdots \cup B_w$, where A' is the set of all leaf nodes in subtrees $i+1, \ldots, w$.

Weights. Since the number of weights in our MST instances is essentially the number of leaf nodes in the ATPC instances, our MST construction only uses polynomially many integer weights. We note that this is necessary due to the result of [AGM12a], as otherwise there is a single pass streaming algorithm that finds an MST in $n^{1+o(1)}$ space. Specifically, an MST can be incrementally found by considering all edges of weight i and applying the Spanning Forest algorithm of [AGM12a] at the i-th step. This can be implemented in a single pass by maintaining W independent copies of the sketch used for the Spanning Forest algorithm, resulting in an $\tilde{O}(nW)$-space algorithm.

Computing the MST weight with large edge weights. So far, we are able to lower bound the search version of MST. We note that the construction shown in Figure 1 can be readily adapted for computing the weight of MSTs if *exponential* edge weights were allowed[5]: edges incident on the j-th non-clique vertex have weights in the order of n^j, so that the minimum weight edge that is not deleted, for each non-clique vertex, can be uniquely recovered from the MST weight alone. However, exponential edge weights would lead to a polynomial overhead in space requirement, which is unaffordable for streaming algorithms. So, we explore the decision version of MST in the following, while keeping the edge weights polynomial.

3.2 The Decision Version

Decisional $\mathbf{UR}^{\subset}_{\mathbf{min}}$. We next proceed to outline our lower bound for the algorithmically-easier decision version of the MST problem. Since there exist efficient algorithms, even with a single pass, for *approximating* the weight of MSTs (e.g. [AGM12a]), we should expect hard instances for the decision version to have MST weights concentrated within a small range. So the following attempt seems plausible. Let e_j be the minimum edge weight for the j-th non-clique vertex, and z_j the parity of e_j. Also let $T = \sum_j e_j - \sum_j z_j$. Then the weight of MSTs is always between T and $T+k$, where $k = n/2$ is the number of non-clique vertices. In the above, we have argued that finding e_j is hard for a fixed j. With little additional effort we can show that computing z_j is also hard.[6] We denote by $\mathbf{UR}^{\subset}_{\text{min,dec}}$ the corresponding decisional universal relation problem, where one needs to compute the parity of the minimum element in $A \backslash B$. We remark that this attempt is in line with [Yu21], in which a decision version of Universal Relation, $\mathbf{UR}^{\subset}_{\text{dec}}$, is utilized to obtain optimal lower bounds for Connectivity.

A majority lemma? One may hope that our final result would again follow from a direct-sum (or, more accurately, "*majority lemma*") type argument: hardness of some boolean function f implies hardness of computing the *majority* of k copies of f[7]. This is because given such a majority

[5] This is not an issue for the search version of MST as even linear edge weights are sufficient to ensure a unique MST, up to edges in the clique.

[6] Recall that the lower bound on $\mathbf{UR}^{\subset}_{\min}$ is derived via ATPC. Roughly speaking, we may view the bottom level as a composition of two sublevels, one of which is binary.

[7] For simplicity, we may assume throughout this section that f is "balanced" in the sense that it evaluates to 0 on exactly half of possible inputs and to 1 on the other half.

lemma, we can simply set the threshold to be $T + k/2$. It is easy to see that the weight of MSTs exceeds $T + k/2$ if and only if the majority of the k parity bits z_j is 1.

Fixing the threshold at the price of correlating the $\mathbf{UR}^{\subset}_{\mathbf{min,dec}}$ instances. To our disappointment, this approach has major problems. One notable issue is that T is "instance dependent", and is not a predetermined value, and therefore the threshold $T + k/2$ is also instance dependent. This is indeed a problem as, in the reduction in Figure 1, the parties would not know the threshold value required for the streaming MST instance. In other words, we don't even have a well-defined input for the decision version of MST! To circumvent this, we add one special edge of weight $T' = C - T$ to the graph, where C is a sufficiently large number to ensure T' is positive. This way, we are always comparing the weight of the MSTs with a *fixed* number $C + k/2$.

In the communication setting, this addition is equivalent to *revealing* T to both parities (implemented as an extra part of input), which *correlates all k copies of $\boldsymbol{UR}^{\subset}_{min,dec}$*. Since a direct-sum style argument typically deals with independent copies, we now need to "get rid" of T. Note that $T = \text{poly}(n) = \text{poly}(k)$. This renders it impossible to brute force over all possible values of T due to the communication constraint.

Another way of getting rid of T would be to make a random guess at T and output randomly if the guess is wrong (with very small communication overhead for verifying the guess). However, this approach has the following major shortcoming: the random guessing reduces the advantage (over 1/2) by a factor of $T = \text{poly}(k)$ but *a majority lemma can never hold in such a low advantage regime!* Specifically, the following may not be true:

> *If computing f with success probability $3/4$ requires C communication, then computing the majority of k copies of f with success probability $1/2 + 1/k$ requires $\tilde{\Omega}(kC)$ communication.*

What's even worse, is that C communication is sufficient to achieve success probability $1/2 + \Theta(1/\sqrt{k})$. To see this, suppose f evaluates to 0 on exactly half of the first $k - 1$ copies and 1 on the other half, and then the majority is solely determined by the output of the last copy. Now consider the protocol that simply computes the value of the last copy and outputs it as the majority. It succeeds whenever the single copy protocol succeeds and thus has constant advantage ($3/4 - 1/2 = 1/4$ to be exact) in the above case, which occurs with probability $\Theta(1/\sqrt{k})$ due to properties of binomial distributions, and is equivalent to a random guess in all other cases as the majority is already determined by the first $k - 1$ copies (recall that we assume f to be balanced). So we cannot hope for a majority lemma that works with advantage well below $\Theta(1/\sqrt{k})$. This dooms our attempt as we are requiring even much lower advantage.

Majority Lemma with hint via XOR Lemma with hint. We work around the above limitation by a different approach. Instead of directly getting rid of T and seeking a majority lemma with low advantage (which turns out to be nonexistent), we convert majority computation into XOR computation by a simple process (with T revealed). Only after that, we again guess T and then utilize an XOR lemma with low advantage which indeed exists. As will be seen later, this alternative approach can be viewed as a majority lemma with high advantage (close to 1/2).

To prove this latter majority lemma, we start from the beautiful recent work [Yu22] that provides a strong XOR lemma in which advantage decreases exponentially in k. We then consider the following process for computing XOR from majority. If the number of 1's is at most $k/2$ (so the majority is 0), return the parity of $k/2$ (assume that k is even), and otherwise return the parity of $k/2 + 1$. Intuitively, the probability of having exactly i 1's is slightly larger than having $i - 1$, for $i \leqslant k/2$. So this process should have certain advantage over 1/2. Indeed, again by properties of binomial distributions, this advantage can be shown to be $\Theta(1/\sqrt{k})$, assuming that the computation of majority is perfect. In general, we can prove that a protocol for computing majority with success probability $1-\epsilon$ implies a protocol for computing XOR with success probability $1/2-\epsilon+\Theta(1/\sqrt{k})$. Since the XOR lemma of [Yu22] proves that a protocol for computing the XOR with success probability $1/2-\epsilon+\Theta(1/\sqrt{k})$ (or even $1/2+\exp(-k)$) is costly, it also implies that the computation of the majority with success probability $1-\epsilon$ is costly. Our entire proof now works as follows.

1. Prove a lower bound on $\mathbf{UR}^{\subset}_{\text{min,dec}}$.
2. Apply the XOR lemma of [Yu22] to show it is also hard to compute the XOR of k copies of $\mathbf{UR}^{\subset}_{\text{min,dec}}$, with success probability $1/2+1/\text{poly}(k)$. This hardness continues to hold with T revealed, which we call a "hint" in our proof.
3. Using the above process, we get a lower bound for computing the majority of k copies of $\mathbf{UR}^{\subset}_{\text{min,dec}}$, with success probability $1-1/\text{poly}(k)$, and also with hint T.
4. Finally, a streaming lower bound for the decision version of MST is derived by our reduction (up to logarithmic factors resulted from boosting the success probability).

All the above ideas are formalized in Section 4. At a high level, what we really use, is roughly a majority lemma of the following form, which has a very weak probability guarantee that is enough for us:

> *If computing f with success probability $3/4$ requires C communication, then computing the majority of k copies of f with success probability $1-1/\text{poly}(k)$ requires $\tilde{\Omega}(kC)$ communication.*

We note, however, that we need such a lemma that also works when T is revealed. As we claimed before, to prove a majority lemma that works when T is revealed, we can guess T, but then need to prove a majority lemma with a very small advantage. Likewise, to show an XOR lemma that works when T is revealed, we can guess T and prove an XOR lemma for very small advantages. Luckily, unlike the case for majority, such an XOR lemma can be proved. Indeed, the XOR lemma of [Yu22] has a strong enough probability guarantee.

Unfortunately, all the above has not yet led to an optimal pass lower bound. Specifically, the XOR lemma of [Yu22] shows that computing the XOR of k copies requires roughly $k/r^{O(r)}$ times the communication for computing a single copy, where r is the number of communication rounds. This loss of an $r^{O(r)}$ factor is problematic as the final lower bound that can be obtained is roughly $n^{1+1/r}/r^{O(r)}$, which only works for r up to $\sqrt{\log n/\log\log n}$. For comparison, [AGM12a] presents a semi-streaming algorithm of $O(\log n/\log\log n)$ passes. So, there is still a gap between the lower and the upper bounds. Furthermore, the loss in communication turns out to be the sole barrier for closing this gap, in the sense that we can prove a tight lower bound if the $r^{O(r)}$ factor could be reduced to $\text{poly}(r)$. We address this challenge in the rest of this section, by proving a *multi-party* XOR lemma (rather than a two-party one) with a better dependence on the number of rounds.

3.3 Multi-Party XOR Lemma

The XOR lemma we need. As indicated above, an ideal XOR lemma (in the standard two-party setting) that is sufficient for our purpose is of the following form: computing the XOR of k copies requires $k/\text{poly}(r)$ times the communication for computing a single copy, to achieve $1/\text{poly}(k)$ advantage. Note that such an XOR lemma does not necessarily improve upon [Yu22] as it only requires a polynomial advantage decay. Nevertheless, to the best of our knowledge, the existence of such an XOR lemma is still unknown.

We prove such a lemma in the multi-party setting. We note that we opt not to restrict ourselves in the two-party setting as our ultimate goal is to prove streaming lower bounds and multi-party settings are usually easier to work with. Nevertheless, our multi-party XOR lemma may be of independent interest as well since it works entirely in the communication setting, with no reference to streaming.

Separating amplification of communication and of advantage. To get our multi-party XOR lemma, we decompose it into two *independent* parts: *amplification of communication* and *amplification of advantage*. More specifically, up to $\mathrm{poly}(r)$ factors, amplification of communication means:

> *If computing f with success probability $1/2 + \delta$ requires C communication, then computing the XOR of k_1 copies of f with success probability $1/2 + \delta + \epsilon$ requires $\tilde{\Omega}_\epsilon(k_1 C)$ communication,*

and amplification of advantage means:

> *If computing f with success probability $3/4$ requires C communication, then computing the XOR of k_2 copies of f with success probability $1/2 + \exp(-\Omega(k_2))$ requires $\tilde{\Omega}(C)$ communication.*

Intuitively, this decomposition is possible because the XOR of many XOR computations is equivalent to a single XOR computation. Furthermore, our desired XOR lemma, up to logarithmic factors, follows from combining amplification of communication with $k_1 = \Theta(k/\log k)$ and amplification of advantage with $k_2 = \Theta(\log k)$.

Amplification of communication. As to amplification of communication, it can be accomplished using known (round-preserving) compression schemes (e.g., [JPY12, BRWY13]) in the standard two-party setting. However, we do emphasize that known compression schemes all seem to have a linear (or even polynomial) dependence on $1/\epsilon$ in the communication if we want an ϵ-simulation. This essentially means amplification of communication has to be performed before amplification of advantage. Otherwise, communication would suffer a polynomial blowup in order to preserve the already amplified advantage.

Amplification of advantage. For amplification of advantage, inspiration is drawn from the streaming XOR lemma by [AN21]. Their result shows that computing the XOR of k copies in the streaming setting with the same space constraint as for a single copy, can only achieve advantage exponentially small in k. Moreover, streaming algorithms are viewed as multi-party communication protocols in their proof. This enables us to adapt their techniques to prove a multi-party communication version: computing the XOR of k copies with ($2k$ parties and) the same total communication as for a single copy (with two parties), can only achieve advantage exponentially small in k. Combined with amplification of communication, it finally yields a multi-party XOR lemma with the desired parameters.

We also remark that the streaming XOR lemma of [AN21] applies to streams in which k copies arrive *sequentially*, i.e., one complete stream followed by another. For our MST construction, this means insertions of the first non-clique vertex is followed by deletions of the same vertex, and then insertions and deletions of the second non-clique vertex and so on. In contrast, our version for multi-party communication has an "interleaved" input order in the sense that part of the first copy (insertions for the first non-clique vertex) is followed by part of the second copy (insertions for the second non-clique vertex) and so on for all other copies, and the remaining part of the first copy (deletions for the first non-clique vertex) only comes after that. Put it another way, all the Alices communicate before all the Bobs. Consequently, the streams resulted from our proof have the *simplest form*: all insertions arrive before all deletions.

4 A Lower Bound in Few Passes

As a warmup, we first prove the following weaker version of Result 1 for only few passes. It already contains many of the critical ideas for fully proving Result 1, while also identifying the key barrier in getting a proof for even more passes.

Theorem 1 (Weaker version of Result 1)**.** *For* $p = o(\sqrt{\frac{\log n}{\log\log n}})$*, any* p*-pass dynamic streaming algorithm for solving* $\boldsymbol{MST}_n$ *with probability* $2/3$ *requires* $\Omega(\frac{n^{1+\frac{1}{2p-1}}}{p^{O(p)}\log n})$ *space.*

We remark that the upper bound on edge weights in Result 1 will be seen in the proof of Claim 4.4.

4.1 Augmented Tree Pointer Chasing

The proof of Theorem 1 is via a communication problem named Augmented Tree Pointer Chasing.

Definition 4.1. *For* $d, w \geqslant 1$*, the two-party problem* $\boldsymbol{ATPC}_{d,w}$ *is defined recursively as follows.*

1. *For* $d = 1$*, Alice is given as input* $A^{(1)} \in \{0,1\}^w$ *and Bob is given as input* $B^{(1)} = (i^{(1)}, A^{(1)}_{<i^{(1)}})$*, where* $i^{(1)} \in [w]$*. They are required to output* $A^{(1)}_{i^{(1)}}$*.*

2. *For* $d > 1$*, Alice is given as input* $A^{(d)} = b^{(d-1)}_{\leqslant w}$ *and Bob is given as input* $B^{(d)} = (i^{(d)}, a^{(d-1)}_{\leqslant i^{(d)}}, b^{(d-1)}_{<i^{(d)}})$*, where* $i^{(d)} \in [w]$ *and* $(a^{(d-1)}_j, b^{(d-1)}_j)$ *for* $j \in [w]$ *is an instance of* $\boldsymbol{ATPC}_{d-1,w}$[8]*. They are required to output the answer to* $(a^{(d-1)}_{i^{(d)}}, b^{(d-1)}_{i^{(d)}})$ *as an instance of* $\boldsymbol{ATPC}_{d-1,w}$*.*

For $k \geqslant 1$*,* $\boldsymbol{ATPC}^{\oplus k}_{d,w}$ *denotes the* k*-fold XOR version of* $\boldsymbol{ATPC}_{d,w}$*, and similarly* $\boldsymbol{ATPC}^{\# k}_{d,w}$ *denotes the* k*-fold majority version of* $\boldsymbol{ATPC}_{d,w}$*.*

Each $\boldsymbol{ATPC}_{d,w}$ *instance can be naturally visualized on a depth-*d*,* w*-ary tree with* i*'s being pointers of corresponding levels. Suppose the leaf nodes are numbered from* 1 *to* w^d*. Starting from the root and following the pointers will lead to a unique leaf node* $t \in [w^d]$*, which is called the* target *of this instance. For either of the* k*-fold versions of* $\boldsymbol{ATPC}_{d,w}$*, both parties may additionally be given the* hint $T = \sum_{j\in[k]} t_j$ *as part of their input, where* t_j *is the target of the* j*-th* $\boldsymbol{ATPC}_{d,w}$ *instance. The resulting problems are denoted by* $\textbf{\textit{Hint-ATPC}}^{\oplus k}_{d,w}$ *and* $\textbf{\textit{Hint-ATPC}}^{\# k}_{d,w}$*, respectively.*

At a high level, in an instance of $\mathbf{ATPC}_{d,w}$, Alice owns all pointers at even levels while Bob owns all pointers at odd levels. It only differs from the stardard Tree Pointer Chasing problem by performing the following modification to each internal node: the owner of a pointer is additionally given the other party's knowledge of subtrees to the left of the pointer, while losing any knowledge of subtrees to the right of the pointer. The modification is performed bottom-up. In other words, the effect of ancestors supersedes that of descendants. Intuitively, the extra information about

[8] For $j > i^{(d)}$, $a_j^{(d-1)}$ is imaginary and given to neither party.

subtrees to the left cannot help while the lost information about subtrees to the right cannot hurt, as the owner of the current node should always follow the pointer. Also note that $\mathbf{ATPC}_{1,w}$ is exactly the same as the well-studied Augmented Index problem. For $d > 1$, $\mathbf{ATPC}_{d,w}$ can be naturally viewed as its multi-round generalization.

The hard input distributions and corresponding lower bounds are as follows.

Distribution 1. For $d, w \geqslant 1$, the hard input distribution $\mathcal{D}_{d,w}$ is defined recursively as follows.

1. For $d = 1$, Alice is given as input $A^{(1)}$ and Bob is given as input $B^{(1)} = (i^{(1)}, A^{(1)}_{<i^{(1)}})$, where $i^{(1)}$ is sampled from $[w]$ uniformly at random and $A^{(1)}$ is independently sampled from $\{0,1\}^w$ uniformly at random.
2. For $d > 1$, Alice is given as input $A^{(d)} = b^{(d-1)}_{\leqslant w}$ and Bob is given as input $B^{(d)} = (i^{(d)}, a^{(d-1)}_{\leqslant i^{(d)}}, b^{(d-1)}_{<i^{(d)}})$, where $i^{(d)}$ is sampled from $[w]$ uniformly at random and $(a^{(d-1)}_j, b^{(d-1)}_j)$ for $j \in [w]$ is independently sampled from $\mathcal{D}_{d-1,w}$.

Lemma 4.2. *For $d, w \geqslant 1$ and $\epsilon \in [0, 1/2]$, it holds that*

$$\mathbf{D}^{(d)}_{\mathcal{D}_{d,w},\frac{1}{2}+\epsilon}(\boldsymbol{ATPC}_{d,w}) \geqslant \frac{\epsilon^2 w}{d}.$$

Distribution 2. For $k, d, w \geqslant 1$, the hard input distribution $\mathcal{D}^k_{d,w}$ is defined as follows. Alice is given as input $A = a^{(d)}_{\leqslant k}$ and Bob is given as input $B = b^{(d)}_{\leqslant k}$, where $(a^{(d)}_j, b^{(d)}_j)$ for $j \in [k]$ is independently sampled from $\mathcal{D}_{d,w}$.

Lemma 4.3. *For $w \geqslant 1$, $d = o(\log w/\log\log w)$, and $k = \omega(d \log w)$, it holds that*

$$\mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w},\frac{2}{3}}(\boldsymbol{Hint\text{-}ATPC}^{\#k}_{d,w}) = \Omega\left(\frac{kw}{d^{O(d)}\log k}\right).$$

We first prove Theorem 1 in Section 4.2, assuming the lower bounds for Augmented Tree Pointer Chasing. Proofs of the above lower bounds are shown in Sections 4.3 and 4.4, respectively.

4.2 Proof of Theorem 1

In this section, we present a proof of Theorem 1 via the following claim.

Claim 4.4. *For $k, p, w, S \geqslant 1$ and $\epsilon \in [0,1]$, if there exists a p-pass, S-space dynamic streaming algorithm for solving $\boldsymbol{MST}_{k+w^{2p-1}+1}$ with probability ϵ, then there also exists a $(2p-1)$-round, $(2p-1)S$-communication protocol for solving $\boldsymbol{Hint\text{-}ATPC}^{\#k}_{2p-1,w}$ with probability ϵ over $\mathcal{D}^k_{2p-1,w}$.*

Before proving Claim 4.4, we show that it indeed implies Theorem 1.

Proof of Theorem 1. Fix a p-pass dynamic streaming algorithm for solving $\mathbf{MST}_n$ with probability 2/3 that has space S. Let $k = (n-1)/2$, $d = 2p-1$, $w = (n-k-1)^{1/d}$, and $C = dS$. Applying the reduction of Claim 4.4, we get a d-round protocol for solving $\textbf{Hint-ATPC}^{\#k}_{d,w}$ with probability $2/3$ over $\mathcal{D}^k_{d,w}$ that has communication C. On the other hand, Lemma 4.3 implies

$$C = \Omega\left(\frac{kw}{d^{O(d)}\log k}\right),$$

or equivalently,

$$S = \Omega\left(\frac{n^{1+\frac{1}{2p-1}}}{p^{O(p)}\log n}\right),$$

as claimed. ∎

We remark that the assumption $p = o(\sqrt{\log n/\log\log n})$ of Theorem 1 is nessesary in the above proof for satisfying the condition $d = o(\log w/\log\log w)$ of Lemma 4.3. Moreover, a closer look at the proofs in Sections 4.3 and 4.4 will reveal that this constraint comes solely from the $r^{O(r)}$-fold decrease in communication when using the XOR lemma of [Yu22]. If the loss factor were reduced to $\text{poly}(r)$ (meaning a better XOR lemma), the constraint would then be relaxed to $d = w^{o(1)}$. In turn, this would be sufficient for proving a lower bound for up to $p = o(\log n/\log\log n)$ passes (and thus the full verion of our main result). Nevertheless, as will be seen in Section 5, we actually take a two-step approach in the absence of such an ideal XOR lemma. At a high level, we will perform the amplification of communication and error probability separately.

The rest of this section constitutes a proof of Claim 4.4. Let $d = 2p-1$, $C = dS$, and $n = k + w^d + 1$. Fix a dynamic streaming algorithm π as described in the claim. In the following, we construct a protocol τ for solving $\textbf{Hint-ATPC}^{\#k}_{d,w}$ with the desired properties. On input $((A = a^{(d)}_{\leqslant k}, T), (B = b^{(d)}_{\leqslant k}, T))$, τ simulates π on the following dynamic stream, where `sender` and `receiver` are defined in Algorithm 1 and Algorithm 2, respectively ($p = 0$ represents Alice and $p = 1$ represents Bob); see Figure 3 for an illustration of the functions and Figure 4 for an illustration of the reduction. The threshold given to π is to be determined.

1. Insert an edge $(1, n)$ with weight $2kw^d - 2T + 1$.

2. For $u < v \in [w^d]$, insert an edge $(k+u, k+v)$ with weight 1.

3. For $j \in [k]$ and $t \in \texttt{sender}(d, w, a^{(d)}_j, 0)$, insert an edge $(j, k + \lceil t/2 \rceil)$ with weight $t + 1$.

4. For $j \in [k]$ and $t \in \texttt{receiver}(d, w, b^{(d)}_j, 1)$, delete the edge $(j, k + \lceil t/2 \rceil)$ with weight $t + 1$.

Algorithm 1. The function $\texttt{sender}(d, w, A^{(d)}, p)$.

- $d = 1$: We have $A^{(1)} \in \{0,1\}^w$.
    - $p = 0$: Return
    $$\left\{2j - A_j^{(1)} \mid j \in [w]\right\}.$$
    - $p = 1$: Return
    $$[2w] \backslash \left\{2j - A_j^{(1)} \mid j \in [w]\right\}.$$
- $d > 1$: We have $A^{(d)} = b_{\leqslant w}^{(d-1)}$, where $b_j^{(d-1)}$ for $j \in [w]$ is a valid input to Bob for $\mathbf{ATPC}_{d-1,w}$. Return
$$\bigcup_{j \in [w]} \left\{2(j-1) \cdot w^{d-1} + t \mid t \in \texttt{receiver}(d-1, w, b_j^{(d-1)}, p)\right\}.$$

Algorithm 2. The function $\texttt{receiver}(d, w, B^{(d)}, p)$.

- $d = 1$: We have $B^{(1)} = (i^{(1)}, A_{<i^{(1)}}^{(1)})$, where $i^{(1)} \in [w]$ and $A_j^{(1)} \in \{0,1\}$ for $j \in [i^{(1)} - 1]$.
    - $p = 0$: Return
    $$[2w] \backslash \left\{2j - A_j^{(1)} \mid j \in [i^{(1)} - 1]\right\}.$$
    - $p = 1$: Return
    $$\left\{2j - A_j^{(1)} \mid j \in [i^{(1)} - 1]\right\}.$$
- $d > 1$: We have $B^{(d)} = (i^{(d)}, a_{\leqslant i^{(d)}}^{(d-1)}, b_{<i^{(d)}}^{(d-1)})$, where $i^{(d)} \in [w]$, $(a_j^{(d-1)}, b_j^{(d-1)})$ for $j \in [i^{(d-1)} - 1]$ is a valid instance of $\mathbf{ATPC}_{d-1,w}$, and $a_{i^{(d)}}^{(d-1)}$ is a valid input to Alice for $\mathbf{ATPC}_{d-1,w}$. If $p = 0$, return
$$\begin{aligned}&\left(\bigcup_{j \in [i^{(d)}-1]} \left\{2(j-1) \cdot w^{d-1} + t \mid t \in \texttt{receiver}(d-1, w, b_j^{(d-1)}, 1)\right\}\right)\\ &\quad \cup \left\{2(i^{(d)} - 1) \cdot w^{d-1} + t \mid t \in \texttt{sender}(d-1, w, a_{i^{(d)}}^{(d-1)}, 0)\right\}\\ &\quad \cup [2i^{(d)} \cdot w^{d-1} + 1, 2w^d],\end{aligned}$$
and if $p = 1$, return
$$\begin{aligned}&\left(\bigcup_{j \in [i^{(d)}-1]} \left\{2(j-1) \cdot w^{d-1} + t \mid t \in \texttt{receiver}(d-1, w, b_j^{(d-1)}, 0)\right\}\right)\\ &\quad \cup \left\{2(i^{(d)} - 1) \cdot w^{d-1} + t \mid t \in \texttt{sender}(d-1, w, a_{i^{(d)}}^{(d-1)}, 1)\right\}.\end{aligned}$$

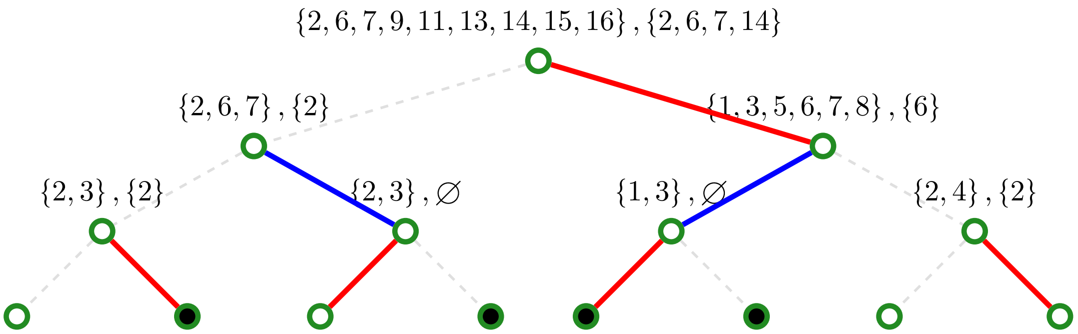


Figure 3: An illustration of functions `sender` and `receiver`, for $d = 3$ and $w = 2$. Blue edges are the pointers owned by Alice (not via augmentation) while red edges are the pointers owned by Bob (not via augmentation). Unfilled leaf nodes have value 0 while filled leaf nodes have value 1. Each internal node is labeled by the set of insertions (for Alice), followed by the set of deletions (for Bob), with respect to the subinstance represented by its subtree, where the owner of the pointer computes `receiver` and the other party computes `sender`.

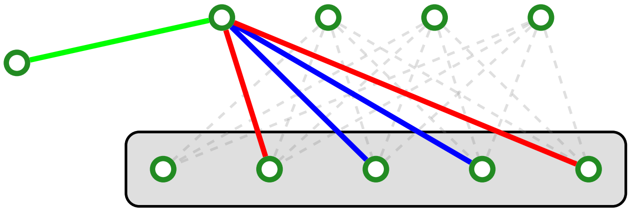


Figure 4: An illustration of the reduction from $\textbf{Hint-ATPC}^{\#k}_{d,w}$ to $\textbf{MST}_n$, for $k = 4$, $w^d = 5$, and $n = 10$. Bottom vertices (encircled in gray) represent the elements of $[w^d]$, which are fully connected as a clique, while each of the top vertices represents an $\textbf{ATPC}_{d,w}$ instance. Red edges correspond to the deletions while each of the blue edges is inserted but not deleted – to avoid clutter, only the edges for $j = 1$ are drawn. The green edge is $(1, n)$.

At a high level, Alice and Bob jointly encode the target of each $\textbf{ATPC}_{d,w}$ instance as the minimum weight edge incident on a unique vertex. To do this, the sender of a message (who does not own the current pointer) has no choice but to collect and merge its insertions/deletions from all subtrees (offset properly to make them disjoint). On the other hand, the receiver owns the current pointer and thus has full knowledge of the subtrees to the left of the pointer, enabling perfect simulation of both parties in all these subtrees. Suppose the receiver performs opposite operations on exactly the same subset of elements as the sender, meaning effectively no edge is inserted/deleted, in each of these subtrees. As a result, the output of the larger instance always corresponds to the output of the smaller instance determined by the current pointer. Besides, to ensure a proper inclusion, Alice as the receiver will insert everything to the right of the pointer while Bob as the receiver will delete nothing to the right of the pointer, as can be seen in the second case of Algorithm 2.

It can be verified that Alice is able to compute all insertions on her own and Bob is able to compute all deletions on his own. So the p-pass dynamic streaming algorithm π can be simulated by the protocol τ, using d rounds and C communication, in the canonical way of exchanging memory states. Now, it remains to formally show the correctness of the reduction. This is done with the help of the following technical claim.

Claim 4.5. *For any $\textbf{ATPC}_{d,w}$ instance $(A^{(d)}, B^{(d)})$, it holds that*

$$\texttt{sender}(d, w, A^{(d)}, 0) \supsetneq \texttt{receiver}(d, w, B^{(d)}, 1),$$

and

$$\texttt{receiver}(d, w, B^{(d)}, 0) \supsetneq \texttt{sender}(d, w, A^{(d)}, 1).$$

Furthermore, it also holds that

$$\min(\texttt{sender}(d, w, A^{(d)}, 0) \backslash \texttt{receiver}(d, w, B^{(d)}, 1)) = 2t - z,$$

and

$$\min(\texttt{receiver}(d, w, B^{(d)}, 0) \backslash \texttt{sender}(d, w, A^{(d)}, 1)) = 2t - z,$$

where t is the target of the instance, and z the output.

Assume the above claim for now. Applying it to $(a_j^{(d)}, b_j^{(d)})$ for $j \in [k]$, we know that the constructed dynamic stream is well-defined and any minimum spanning tree of the constructed graph must consist of the following edges.

1. The edge $(1, n)$ with weight $2kw^d - 2T + 1$.
2. $w^d - 1$ edges connecting $[k + 1, k + w^d]$, each with weight 1.
3. The edge $(j, k + t_j)$ with weight $2t_j - z_j + 1$, where t_j is the target of $(a_j^{(d)}, b_j^{(d)})$ and z_j is the output of the same instance, for $j \in [k]$.

Therefore, the weight of minimum spanning trees is

$$2kw^d - 2T + 1 + w^d - 1 + \sum_{j \in [k]} (2t_j - z_j + 1) = 2kw^d + w^d + k - \sum_{j \in [k]} z_j.$$

In other words, τ will output 1 (i.e., more than $\lfloor k/2 \rfloor$ out of the k instances of $\mathbf{ATPC}_{d,w}$ output 1) if and only if π outputs 0 given threshold $2kw^d + w^d + k - \lfloor k/2 \rfloor$ (i.e., the weight of minimum spanning trees is less than the given threshold). So the success probability remains the same.

We conclude this section with a proof of Claim 4.5.

Proof of Claim 4.5. The proof is by induction on d. The base case of $d = 1$ is a direct consequence of the first cases of Algorithm 1 and Algorithm 2. For $d > 1$, we can get

$$\texttt{sender}(d, w, A^{(d)}, 0) \supsetneq \texttt{receiver}(d, w, B^{(d)}, 1),$$

because by the inductive hypothesis, we have that

$$\texttt{receiver}(d - 1, w, b_{i^{(d)}}^{(d-1)}, 0) \supsetneq \texttt{sender}(d - 1, w, a_{i^{(d)}}^{(d-1)}, 1).$$

Furthermore, we actually know

$$\begin{aligned}
&\min(\texttt{sender}(d, w, A^{(d)}, 0) \backslash \texttt{receiver}(d, w, B^{(d)}, 1)) \\
&\quad = 2(i^{(d)} - 1) \cdot w^{d-1} + \min(\texttt{receiver}(d - 1, w, b_{i^{(d)}}^{(d-1)}, 0) \backslash \texttt{sender}(d - 1, w, a_{i^{(d)}}^{(d-1)}, 1)) \\
&\quad = 2(i^{(d)} - 1) \cdot w^{d-1} + 2t' - z' \\
&\quad = 2t - z,
\end{aligned}$$

where t' is the target of $(a_{i^{(d)}}^{(d-1)}, b_{i^{(d)}}^{(d-1)})$ and z' is the output of the same instance.

Similarly, we can also get

$$\texttt{receiver}(d, w, B^{(d)}, 0) \supsetneq \texttt{sender}(d, w, A^{(d)}, 1),$$

because by the inductive hypothesis, we have that

$$\texttt{sender}(d-1, w, a^{(d-1)}_{i^{(d)}}, 0) \supsetneq \texttt{receiver}(d-1, w, b^{(d-1)}_{i^{(d)}}, 1).$$

Furthermore, we actually know

$$\begin{aligned}
&\min(\texttt{receiver}(d, w, B^{(d)}, 0)\backslash\texttt{sender}(d, w, A^{(d)}, 1)) \\
&\quad = 2(i^{(d)} - 1) \cdot w^{d-1} + \min(\texttt{sender}(d-1, w, a^{(d-1)}_{i^{(d)}}, 0)\backslash\texttt{receiver}(d-1, w, b^{(d-1)}_{i^{(d)}}, 1)) \\
&\quad = 2(i^{(d)} - 1) \cdot w^{d-1} + 2t' - z' \\
&\quad = 2t - z,
\end{aligned}$$

where t' is the target of $(a^{(d-1)}_{i^{(d)}}, b^{(d-1)}_{i^{(d)}})$ and z' is the output of the same instance. This concludes the proof. ∎

We remark that Claim 4.5 can also be viewed as a reduction from $\mathbf{ATPC}_{d,w}$ to $\mathbf{UR}^{\subset}_{\text{min,dec}}$ over a universe of size $m = 2w^d$, by Alice computing the set $\texttt{sender}(d, w, A^{(d)}, 0)$ and Bob computing the set $\texttt{receiver}(d, w, B^{(d)}, 1)$. So the following corollary follows immediately from the lower bound for $\mathbf{ATPC}_{d,w}$ (Lemma 4.2).

Corollary 4.6. *For $m, r \geqslant 1$, and $\epsilon \in [0, 1/2]$, any r-round (randomized) protocol that solves $\boldsymbol{UR}^{\subset}_{min,dec}$ with probability $1/2 + \epsilon$ over a universe of size m, requires $\Omega(\epsilon^2 m^{1/r}/r)$ communication.*

4.3 Lower Bound for $\mathbf{ATPC}_{d,w}$

We derive Lemma 4.2 by round elimination in this section. Claims 4.7 and 4.8 take care of the base case and each round elimination step, respectively.

Claim 4.7 (Base case)**.** *For $w \geqslant 1$, any one-way protocol π for solving $\boldsymbol{ATPC}_{1,w}$ succeeds with probability at most $1/2 + \sqrt{\mathbf{CC}(\pi)/w}$ over $\mathcal{D}_{1,w}$.*

Proof. Throughout the proof, all superscripts on random variables will be temporarily omitted for conciseness. Let M be the message sent from Alice to Bob. We can get

$$\begin{aligned}
\mathbb{I}(\mathsf{M}, \mathsf{A}_{<\mathsf{I}}, \mathsf{I} \,;\, \mathsf{A}_{\mathsf{I}}) &= \mathbb{I}(\mathsf{A}_{<\mathsf{I}}, \mathsf{I} \,;\, \mathsf{A}_{\mathsf{I}}) + \mathbb{I}(\mathsf{M} \,;\, \mathsf{A}_{\mathsf{I}} \mid \mathsf{A}_{<\mathsf{I}}, \mathsf{I}) \\
&\qquad \text{(by chain rule of mutual information (Fact A.1-(6)))} \\
&= \mathbb{I}(\mathsf{M} \,;\, \mathsf{A}_{\mathsf{I}} \mid \mathsf{A}_{<\mathsf{I}}, \mathsf{I}) && \text{(as } \mathsf{A}_{<\mathsf{I}}, \mathsf{I} \perp \mathsf{A}_{\mathsf{I}}) \\
&= \frac{1}{w} \cdot \sum_{i \in [w]} \mathbb{I}(\mathsf{M} \,;\, \mathsf{A}_i \mid \mathsf{A}_{<i}, \mathsf{I} = i) && \text{(as } \mathsf{I} \text{ is uniform)} \\
&= \frac{1}{w} \cdot \sum_{i \in [w]} \mathbb{I}(\mathsf{M} \,;\, \mathsf{A}_i \mid \mathsf{A}_{<i}) && \text{(as } \mathsf{M}, \mathsf{A}_{\leqslant i} \perp \mathsf{I} = i) \\
&= \frac{\mathbb{I}(\mathsf{M} \,;\, \mathsf{A})}{w} && \text{(by chain rule of mutual information (Fact A.1-(6)))} \\
&\leqslant \frac{\mathbf{CC}(\pi)}{w}.
\end{aligned}$$

Furthermore, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}} \|\mathrm{dist}(\mathsf{A}_\mathsf{I} \mid \mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}) - \mathrm{dist}(\mathsf{A}_\mathsf{I})\|_{\mathrm{tvd}} \\
&\quad\leqslant \mathop{\mathbb{E}}_{\mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}} \sqrt{\mathbb{D}(\mathrm{dist}(\mathsf{A}_\mathsf{I} \mid \mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}) \mid\mid \mathrm{dist}(\mathsf{A}_\mathsf{I}))} && \text{(by Pinsker's inequality (Fact A.6))} \\
&\quad\leqslant \sqrt{\mathop{\mathbb{E}}_{\mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}} \mathbb{D}(\mathrm{dist}(\mathsf{A}_\mathsf{I} \mid \mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}) \mid\mid \mathrm{dist}(\mathsf{A}_\mathsf{I}))} && \text{(by concavity of } \sqrt{\cdot}\text{)} \\
&\quad= \sqrt{\mathbb{I}(\mathsf{A}_\mathsf{I}\,;\mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I})} && \text{(by Fact A.3)} \\
&\quad\leqslant \sqrt{\frac{\mathbf{CC}(\pi)}{w}}.
\end{aligned}$$

Since A_I is initially uniformly random, the probability that Bob can correctly guess A_I from his perspective (i.e., given $\mathsf{M},\mathsf{A}_{<\mathsf{I}},\mathsf{I}$) is then at most $1/2 + \sqrt{\mathbf{CC}(\pi)/w}$, as claimed. ∎

Claim 4.8 (Round elimination)**.** *For $d, w \geqslant 1$ and $\epsilon \in [0,1]$, if there exists a $(d+1)$-round protocol π for solving $\boldsymbol{ATPC}_{d+1,w}$ with probability ϵ over $\mathcal{D}_{d+1,w}$, then there also exists a d-round protocol τ for solving $\boldsymbol{ATPC}_{d,w}$ with probability $\epsilon - \sqrt{\mathbf{CC}^{(1)}(\pi)/w}$ over $\mathcal{D}_{d,w}$ and communication $\mathbf{CC}(\tau) = \mathbf{CC}^{(>1)}(\pi)$.*

Proof. Throughout the proof, all superscripts on random variables will be temporarily omitted for conciseness. When there is ambiguity, unprimed quantities represent the d-round versions while primed ones are reserved for $d+1$ rounds. Let M be the first-round message of π. On input (A, B), τ will simulate π as shown in Algorithm 3, where all random variables are with respect to π.

Algorithm 3. The d-round protocol τ for solving $\mathbf{ATPC}_{d,w}$ on input (A, B).

1. Alice and Bob publicly sample $\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}$.
2. Alice sets $\mathsf{A}_\mathsf{I} = A$ and Bob sets $\mathsf{B}_\mathsf{I} = B$.
3. Alice privately samples $\mathsf{A}_{<\mathsf{I}}$ conditioned on $\mathsf{M},\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I}$, and sets $\mathsf{B}' = (\mathsf{I},\mathsf{A}_{\leqslant\mathsf{I}},\mathsf{B}_{<\mathsf{I}})$.
4. Bob privately samples $\mathsf{B}_{>\mathsf{I}}$ conditioned on $\mathsf{M},\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}$, and sets $\mathsf{A}' = \mathsf{B}$.
5. Alice and Bob simulate π on input $(\mathsf{B}',\mathsf{A}')$ from the second round, assuming the first-round message is M, with Alice playing the role of Bob and Bob playing the role of Alice.
6. Return the output of π.

It can be verified that $(\mathsf{A}',\mathsf{B}')$ is a valid instance of $\mathbf{ATPC}_{d+1,w}$ and thus τ is indeed a d-round protocol for solving $\mathbf{ATPC}_{d,w}$, with the claimed communication complexity. We also want to emphasize that with Alice and Bob playing the roles of each other, M is the first-round message of π from Bob to Alice, so it is a function of $\mathsf{A}' = \mathsf{B}$. Now, it remains to calculate the success probability of τ. To this end, the following two technical claims show that all the random variables as sampled in τ almost perfectly follow their distribution in π.

Claim 4.9. *It holds that*

$$\mathbb{I}(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\,;\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \leqslant \frac{\mathbf{CC}^{(1)}(\pi)}{w}.$$

Proof. Observe that

$$\begin{aligned}
\mathbb{I}(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\,;\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}) &= \mathbb{I}(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\,;\mathsf{B}_{<\mathsf{I}},\mathsf{I}) + \mathbb{I}(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \\
&\qquad \text{(by chain rule of mutual information (Fact A.1-(6)))} \\
&= \mathbb{I}(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \qquad \text{(as } \mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I}\perp\mathsf{B}_{<\mathsf{I}},\mathsf{I}\text{)} \\
&= \mathbb{I}(\mathsf{B}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{<\mathsf{I}},\mathsf{I}) + \mathbb{I}(\mathsf{A}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}) \\
&\qquad \text{(by chain rule of mutual information (Fact A.1-(6)))} \\
&\leqslant \mathbb{I}(\mathsf{B}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{<\mathsf{I}},\mathsf{I}) + \mathbb{I}(\mathsf{A}_\mathsf{I}\,;\mathsf{B}_{>\mathsf{I}}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}) \\
&\qquad \text{(by data processing inequality (Fact A.1-(7)) as } \mathsf{M} \text{ is a function of } \mathsf{B}_{>\mathsf{I}},\mathsf{B}_{\leqslant\mathsf{I}}\text{)} \\
&= \mathbb{I}(\mathsf{B}_\mathsf{I}\,;\mathsf{M}\mid\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \qquad \text{(as } \mathsf{A}_\mathsf{I}\perp\mathsf{B}_{>\mathsf{I}}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}\text{)} \\
&= \frac{1}{w}\cdot\sum_{i\in[w]}\mathbb{I}(\mathsf{B}_i\,;\mathsf{M}\mid\mathsf{B}_{<i},\mathsf{I}=i) \qquad \text{(as } \mathsf{I} \text{ is uniform)} \\
&= \frac{1}{w}\cdot\sum_{i\in[w]}\mathbb{I}(\mathsf{B}_i\,;\mathsf{M}\mid\mathsf{B}_{<i}) \qquad \text{(as } \mathsf{M},\mathsf{B}_{\leqslant i}\perp\mathsf{I}=i\text{)} \\
&= \frac{\mathbb{I}(\mathsf{B}\,;\mathsf{M})}{w} \qquad \text{(by chain rule of mutual information (Fact A.1-(6)))} \\
&\leqslant \frac{\mathbf{CC}^{(1)}(\pi)}{w},
\end{aligned}$$

as claimed. ∎

Claim 4.10. *It holds that*

$$\mathsf{A}_{<\mathsf{I}}\perp\mathsf{B}_\mathsf{I}\mid\mathsf{M},\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I},$$

and

$$\mathsf{B}_{>\mathsf{I}}\perp\mathsf{A}_{\leqslant\mathsf{I}}\mid\mathsf{M},\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}.$$

Proof. Observe that

$$\begin{aligned}
\mathbb{I}(\mathsf{A}_{<\mathsf{I}}\,;\mathsf{B}_\mathsf{I}\mid\mathsf{M},\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I}) &\leqslant \mathbb{I}(\mathsf{A}_{<\mathsf{I}}\,;\mathsf{M},\mathsf{B}_\mathsf{I}\mid\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \\
&\leqslant \mathbb{I}(\mathsf{A}_{<\mathsf{I}}\,;\mathsf{B}_{\geqslant\mathsf{I}}\mid\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I}) \\
&\qquad \text{(by data processing inequality (Fact A.1-(7)) as } \mathsf{M} \text{ is a function of } \mathsf{B}_{\geqslant\mathsf{I}},\mathsf{B}_{<\mathsf{I}}\text{)} \\
&= 0, \qquad \text{(as } \mathsf{A}_{<\mathsf{I}}\perp\mathsf{B}_{\geqslant\mathsf{I}}\mid\mathsf{A}_\mathsf{I},\mathsf{B}_{<\mathsf{I}},\mathsf{I}\text{)}
\end{aligned}$$

and that

$$\begin{aligned}
\mathbb{I}(\mathsf{B}_{>\mathsf{I}}\,;\mathsf{A}_{\leqslant\mathsf{I}}\mid\mathsf{M},\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}) &\leqslant \mathbb{I}(\mathsf{M},\mathsf{B}_{>\mathsf{I}}\,;\mathsf{A}_{\leqslant\mathsf{I}}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}) \\
&\leqslant \mathbb{I}(\mathsf{B}_{>\mathsf{I}}\,;\mathsf{A}_{\leqslant\mathsf{I}}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}) \\
&\qquad \text{(by data processing inequality (Fact A.1-(7)) as } \mathsf{M} \text{ is a function of } \mathsf{B}_{>\mathsf{I}},\mathsf{B}_{\leqslant\mathsf{I}}\text{)} \\
&= 0. \qquad \text{(as } \mathsf{B}_{>\mathsf{I}}\perp\mathsf{A}_{\leqslant\mathsf{I}}\mid\mathsf{B}_{\leqslant\mathsf{I}},\mathsf{I}\text{)}
\end{aligned}$$

This concludes the proof. ∎

Note that τ succeeds on $(\mathsf{A}_\mathsf{I},\mathsf{B}_\mathsf{I})$ so long as π does on $(\mathsf{A}',\mathsf{B}')$. Intuitively, τ samples $(\mathsf{A}',\mathsf{B}')$ closely following $\mathcal{D}_{d+1,w}$ so its success probability should also be close to the success probability of π. This can be formally argued as follows. By chain rule of total variation distance (Fact A.8), together with Claims 4.9 and 4.10, we can get that the distribution sampled by τ of all random variables $(\mathsf{M}, \mathsf{A}_{\leqslant \mathsf{I}}, \mathsf{B}, \mathsf{I})$ and their distribution in π have a total variation distribution upper bounded by

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}} \|\mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I} \mid \mathsf{M}, \mathsf{B}_{<\mathsf{I},\mathsf{I}}) - \mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I})\|_{\mathrm{tvd}} \\
&\qquad \leqslant \mathop{\mathbb{E}}_{\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}} \sqrt{\mathbb{D}(\mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I} \mid \mathsf{M}, \mathsf{B}_{<\mathsf{I},\mathsf{I}}) \;||\; \mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I}))} && \text{(by Pinsker's inequality (Fact A.6))} \\
&\qquad \leqslant \sqrt{\mathop{\mathbb{E}}_{\mathsf{M},\mathsf{B}_{<\mathsf{I}},\mathsf{I}} \mathbb{D}(\mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I} \mid \mathsf{M}, \mathsf{B}_{<\mathsf{I},\mathsf{I}}) \;||\; \mathrm{dist}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I}))} && \text{(by concavity of } \sqrt{\cdot}\text{)} \\
&\qquad = \sqrt{\mathbb{I}(\mathsf{A}_\mathsf{I}, \mathsf{B}_\mathsf{I}\,;\mathsf{M}, \mathsf{B}_{<\mathsf{I},\mathsf{I}})} && \text{(by Fact A.3)} \\
&\qquad \leqslant \sqrt{\frac{\mathbf{CC}^{(1)}(\pi)}{w}}.
\end{aligned}$$

As a result, the overall success probability of τ is less than that of π by at most $\sqrt{\mathbf{CC}^{(1)}(\pi)/w}$ due to Fact A.5, concluding the proof. ∎

Now, we are ready to prove Lemma 4.2.

Proof of Lemma 4.2. Fix a d-round protocol π for solving $\mathbf{ATPC}_{d,w}$ with probability $1/2 + \epsilon$ over $\mathcal{D}_{d,w}$ that has optimal communication. By applying the round elimination step (Claim 4.8) repeatedly for $d - 1$ times, we get a one-way protocol for solving $\mathbf{ATPC}_{1,w}$ with probability $1/2 + \epsilon - \sum_{r\in[d-1]} \sqrt{\mathbf{CC}^{(r)}(\pi)/w}$ over $\mathcal{D}_{1,w}$ and communication $\mathbf{CC}^{(d)}(\pi)$. On the other hand, Claim 4.7 implies that

$$\begin{aligned}
\epsilon &\leqslant \sum_{r\in[d]} \sqrt{\frac{\mathbf{CC}^{(r)}(\pi)}{w}} \\
&= d \cdot \sum_{r\in[d]} \frac{1}{d} \cdot \sqrt{\frac{\mathbf{CC}^{(r)}(\pi)}{w}} \\
&\leqslant d \cdot \sqrt{\sum_{r\in[d]} \frac{1}{d} \cdot \frac{\mathbf{CC}^{(r)}(\pi)}{w}} && \text{(by concavity of } \sqrt{\cdot}\text{)} \\
&= \sqrt{\frac{d}{w} \cdot \mathbf{CC}(\pi)}.
\end{aligned}$$

The lower bound is derived by rearranging the terms. ∎

4.4 Lower Bound for Hint-ATPC$^{\#k}_{d,w}$

We generalize the lower bound for $\mathbf{ATPC}_{d,w}$ to its k-fold verisons in this section. The hardness of $\textbf{Hint-ATPC}^{\#k}_{d,w}$ will be proved by a series of reductions from $\mathbf{ATPC}^{\oplus k}_{d,w}$. Indeed, Claim 4.11 deals with the hint, namely the sum of targets, while Claim 4.12 relates the XOR and the majority versions of the problem. The hardness of $\mathbf{ATPC}^{\oplus k}_{d,w}$ will in turn be derived from the hardness of $\mathbf{ATPC}_{d,w}$ using the XOR lemma of [Yu22] (Lemma 4.13).

Claim 4.11. *For $k, d, w \geqslant 1$ and $\epsilon \in [0, 1/2]$, it holds that*

$$\mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w}, \frac{1}{2}+\frac{\epsilon}{kw^d}}(\boldsymbol{ATPC}^{\oplus k}_{d,w}) \leqslant \mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w}, \frac{1}{2}+\epsilon}(\boldsymbol{Hint\text{-}ATPC}^{\oplus k}_{d,w}) + kd\log w.$$

Proof. Fix a d-round protocol π for solving $\mathbf{Hint\text{-}ATPC}^{\oplus k}_{d,w}$ with probability $1/2+\epsilon$ over $\mathcal{D}^k_{d,w}$ that has optimal communication. In the following, we construct a protocol τ for solving $\mathbf{ATPC}^{\oplus k}_{d,w}$ with the claimed properties. On input (A, B), τ starts by publicly guessing T' from $[kw^d]$ uniformly at random and simulating π on input $((A, T'), (B, T'))$. Then, Alice sends the first message of π to Bob. For $r \in [2, d]$, the sender of the r-th round will transmit the r-th message of π, which can be simulated using the current transcript, as well as the correct depth-$(r-1)$ pointers for each of the k instances of $\mathbf{ATPC}_{d,w}$. So the receiver of the r-th round will know what the correct depth-r pointers are. At the end of the protocol, the receiver of the last message has full knowledge required to compute the sum T of all k targets and can verify whether $T' = T$. If so, return the output of π, and otherwise a uniformly random bit.

Note that apart from messages of π, τ transmits k pointers, each of which costs $\log w$ bits, in each of the d rounds, so the additional communication is $kd\log w$ in total. To see its success probability, observe that T' is a uniformly random guess independent of (A, B). Thus, it correctly hits T with probability $1/(kw^d)$ and τ succeeds with probability $1/2+\epsilon$, the same as π, conditioned on this event. Otherwise, τ succeeds with probability $1/2$ as a uniformly random bit is output. The overall success probability then follows as claimed. ∎

Claim 4.12. *For $k, d, w \geqslant 1$ and $\epsilon \in [0, 1/2]$, it holds that*

$$\mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w}, \frac{1}{2}-\epsilon+\Theta(\frac{1}{\sqrt{k}})}(\boldsymbol{Hint\text{-}ATPC}^{\oplus k}_{d,w}) \leqslant \mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w}, 1-\epsilon}(\boldsymbol{Hint\text{-}ATPC}^{\# k}_{d,w}).$$

Proof. Fix a d-round protocol π for solving $\mathbf{Hint\text{-}ATPC}^{\# k}_{d,w}$ with probability $1-\epsilon$ over $\mathcal{D}^k_{d,w}$ that has optimal communication. In the following, we construct a protocol τ for solving $\mathbf{Hint\text{-}ATPC}^{\oplus k}_{d,w}$ with the claimed properties. On input $((A, T), (B, T))$, τ simulates π on input $((A, T), (B, T))$, and outputs the parity of $\lfloor k/2 \rfloor + 1$ if π outputs 1 and outputs the parity of $\lfloor k/2 \rfloor$ otherwise. So it remains to calculate the success probability of τ.

Assume for now that π were perfectly correct (i.e., $\epsilon = 0$). Note that the answer to an instance drawn from $\mathcal{D}_{d,w}$ is simply a uniformly random bit. Therefore, τ succeeds with probability

$$\begin{aligned}
&\sum_{j=0}^{\lfloor k/2 \rfloor} \mathbf{1}[\lfloor k/2 \rfloor - j \text{ is even}] \cdot \frac{\binom{k}{j}}{2^k} + \sum_{j=\lfloor k/2 \rfloor+1}^{k} \mathbf{1}[j - \lfloor k/2 \rfloor - 1 \text{ is even}] \cdot \frac{\binom{k}{j}}{2^k} \\
&= \frac{1}{2} \cdot \left[\sum_{j=0}^{\lfloor k/2 \rfloor} \frac{\binom{k}{j}}{2^k} + \sum_{j=0}^{\lfloor k/2 \rfloor} (-1)^{\lfloor k/2 \rfloor - j} \cdot \frac{\binom{k}{j}}{2^k} \right] \\
&\quad + \frac{1}{2} \cdot \left[\sum_{j=\lfloor k/2 \rfloor+1}^{k} \frac{\binom{k}{j}}{2^k} + \sum_{j=\lfloor k/2 \rfloor+1}^{k} (-1)^{j - \lfloor k/2 \rfloor - 1} \cdot \frac{\binom{k}{j}}{2^k} \right] \\
&= \frac{1}{2} \cdot \left[1 + \sum_{j=0}^{\lfloor k/2 \rfloor} (-1)^{\lfloor k/2 \rfloor - j} \cdot \frac{\binom{k}{j}}{2^k} + \sum_{j=\lfloor k/2 \rfloor+1}^{k} (-1)^{j - \lfloor k/2 \rfloor - 1} \cdot \frac{\binom{k}{j}}{2^k} \right]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2} \cdot \left[1 + \frac{\binom{k-1}{\lfloor k/2 \rfloor}}{2^k} + \frac{\binom{k-1}{\lfloor k/2 \rfloor}}{2^k} \right] \\
&= \frac{1}{2} + \frac{\binom{k-1}{\lfloor k/2 \rfloor}}{2^k} \\
&= \frac{1}{2} + \Theta\left(\frac{1}{\sqrt{k}} \right),
\end{aligned}
$$

by Stirling's approximation. For π with error probability ϵ, by a union bound, the success probability of τ is reduced by at most ϵ. The claim hence follows. ∎

We remark that the reductions in proving Claims 4.11 and 4.12 actually have little to do with the base problem $\mathbf{ATPC}_{d,w}$ itself. In fact, almost identical reductions will also be used in Section 5 for proving the full version of our main result.

Now, we are ready to prove Lemma 4.3 using the following XOR lemma of [Yu22][9].

Lemma 4.13 ([Yu22]). *For $k, r \geqslant 1$, Boolean function f, and input distance μ, it holds that*

$$\mathbf{D}^{(r)}_{\mu^k, \frac{1}{2}+\frac{1}{2^k}}(f^{\oplus k}) \geqslant k \cdot \left(\frac{1}{r^{O(r)}} \cdot \mathbf{D}^{(r)}_{\mu, \frac{2}{3}}(f) - 1 \right).$$

Proof of Lemma 4.3. Fix a d-round protocol for solving $\mathbf{Hint\text{-}ATPC}^{\#k}_{d,w}$ with probability $1 - 1/\mathrm{poly}(k)$ over $\mathcal{D}^k_{d,w}$ that has optimal communication C. Applying Claim 4.12 and Claim 4.11 in sequence, we get a d-round protocol for solving $\mathbf{ATPC}^{\oplus k}_{d,w}$ with probability $1/2 + \Theta(1/(k^{3/2}w^d))$ over $\mathcal{D}^k_{d,w}$ that has communication $C + kd\log w$.

On the other hand, Lemma 4.2, together with Lemma 4.13, implies that

$$\mathbf{D}^{(d)}_{\mathcal{D}^k_{d,w}, \frac{1}{2}+\frac{1}{2^k}}(\mathbf{ATPC}^{\oplus k}_{d,w}) = \Omega\left(\frac{kw}{d^{O(d)}} \right),$$

under the assumption $d = o(\log w / \log\log w)$. Since $1/2^k \ll 1/(k^{3/2}w^d)$ under the assumption $k = \omega(d \log w)$, we have that $C = \Omega(kw/d^{O(d)})$. The lemma follows by observing that error reduction from $1/3$ down to $1/\mathrm{poly}(k)$ requires an $O(\log k)$-fold increase in communication. ∎

5 A Lower Bound in Optimal Number of Passes

In this section, we extend Theorem 1 to more passes as shown in Theorem 2, fully proving Result 1. Also, Result 2, formalized in Theorem 3, will be a direct consequence of Lemmas 5.1 and 5.2.

Theorem 2 (Formal version of Result 1). *For $p = o(\frac{\log n}{\log\log n})$, any p-pass dynamic streaming algorithm for solving $\boldsymbol{MST}_n$ with probability $2/3$ requires $\Omega(\frac{n^{1+\frac{1}{2p-1}}}{p^5 \log^3 n})$ space.*

[9] Technically, the main result (Theorem 1) of [Yu22] is an XOR lemma for randomized communication complexity, while a distributional version is required in our proof. Nevertheless, [Yu22] proves the main result via another one (Theorem 2) with asymmetirc communication, which directly works in the distributional model. The distributional version we need is a natural byproduct of the simple argument from Theorem 2 to Theorem 1; see Section 4 in the full version of [Yu22] for more details.

Theorem 3 (Formal version of Result 2)**.** *There exists $\epsilon_0 > 0$ such that for $n, r \geqslant 1$, $k \in [1, n]$, $\epsilon \in (0, \epsilon_0)$, Boolean function f, and input distribution μ, it holds that*

$$\mathbf{D}^{(r),k}_{\mu^n,\frac{1}{2}+\min(\epsilon_1,\epsilon_2)}(f^{\oplus n}) = \Omega\left(\frac{n}{k}\cdot\left(\frac{\epsilon}{r}\cdot\mathbf{D}^{(r)}_{\mu,\frac{1}{2}+\epsilon}(f) - O(r)\right)\right),$$

where $\epsilon_1 = (r\epsilon)^{\Omega(k/r)}$ and $\epsilon_2 = \epsilon^{\Omega(\epsilon k/r)}$.

As mentioned in Section 4, we take a two-step approach by amplifying first communication and then error probability. The first step uses the following XOR-direct-sum result[10], tight up to a factor of r, for bounded-round communication complexity, which is implicitly implied by [JPY12].

Lemma 5.1 (Bounded-round XOR direct sum)**.** *For $k, r \geqslant 1$, $\epsilon \in [0, 1]$, $\delta \in (0, \epsilon)$, Boolean function f, and input distribution μ, it holds that*

$$\mathbf{D}^{(r)}_{\mu^k,\epsilon}(f^{\oplus k}) = \Omega\left(k\cdot\left(\frac{\delta}{r}\cdot\mathbf{D}^{(r)}_{\mu,\epsilon-\delta}(f) - O(r)\right)\right).$$

For the second step, we prove an XOR-lemma-type result for the multi-party model.

Lemma 5.2 (Multi-party XOR lemma)**.** *There exists $\epsilon_0 > 0$ such that for $k, r \geqslant 1$, $\epsilon \in (0, \epsilon_0)$, Boolean function f, and input distribution μ, it holds that*

$$\mathbf{D}^{(r),k}_{\mu^k,\frac{1}{2}+\min(\epsilon_1,\epsilon_2)}(f^{\oplus k}) \geqslant \mathbf{D}^{(r)}_{\mu,\frac{1}{2}+\epsilon}(f),$$

where $\epsilon_1 = (r\epsilon)^{\Omega(k/r)}$ and $\epsilon_2 = \epsilon^{\Omega(\epsilon k/r)}$.

We remark that Lemma 5.2 is a weaker XOR lemma than the one of [Yu22] in the sense that the decrease in advantage is worse and it only applies to the multi-party model. It nevertheless meets our needs as we will eventually work in the streaming model. On the positive side, Lemma 5.2 no longer suffers a factor of $r^{O(r)}$ in communication, which is exactly the only barrier towards $o(\log n/\log\log n)$ passes as identified in Section 4.

Formal proofs of Lemmas 5.1 and 5.2 are deferred to Appendix B. We now show that they indeed imply Theorem 2. This is done via multi-party variants of the problems in Section 4.

Definition 5.3. *For $k_1, k_2, d, w \geqslant 1$, $\boldsymbol{ATPC}^{\oplus(k_1,k_2)}_{d,w}$ denotes the k_1k_2-fold XOR version of $\boldsymbol{ATPC}_{d,w}$ in the $2k_2$-party model, where each pair of Alice i and Bob i is given as input k_1 instances of $\boldsymbol{ATPC}_{d,w}$, where $i \in [k_2]$. Similarly, $\boldsymbol{ATPC}^{\#(k_1,k_2)}_{d,w}$ denotes the k_1k_2-fold majority version of $\boldsymbol{ATPC}_{d,w}$ in the $2k_2$-party model.*

When the hint, i.e., the total sum of targets of all k_1k_2 instances, is given to each of the $2k_2$ parties as part of its input, the resulting problems are denoted by $\boldsymbol{Hint\text{-}ATPC}^{\oplus(k_1,k_2)}_{d,w}$ and $\boldsymbol{Hint\text{-}ATPC}^{\#(k_1,k_2)}_{d,w}$.

Multi-party analogues of Claims 4.4, 4.11 and 4.12 are given below. The proofs are almost identical and omitted here, as the same reductions still apply.

Claim 5.4. *For $k_1, k_2, p, w, S \geqslant 1$ and $\epsilon \in [0, 1]$, if there exists a p-pass, S-space dynamic streaming algorithm for solving $\boldsymbol{MST}_{k_1k_2+w^{2p-1}+1}$ with probability ϵ, then there also exists a $(2p-1)$-round, $(2p-1)k_2S$-communication protocol for solving $\boldsymbol{Hint\text{-}ATPC}^{\#(k_1,k_2)}_{2p-1,w}$ with probability ϵ over $\mathcal{D}^{k_1k_2}_{2p-1,w}$.*

[10] A similar direct-sum result also holds for f^k. It is however subsumed by the direct-product result of [JPY12].

Claim 5.5. *For $k_1, k_2, d, w \geqslant 1$, and $\epsilon \in [0, 1/2]$, it holds that*

$$\mathbf{D}^{(d),k_2}_{\mathcal{D}^{k_1k_2}_{d,w}, \frac{1}{2}+\frac{\epsilon}{k_1k_2w^d}}(\boldsymbol{ATPC}^{\oplus(k_1,k_2)}_{d,w}) \leqslant \mathbf{D}^{(d),k_2}_{\mathcal{D}^{k_1k_2}_{d,w}, \frac{1}{2}+\epsilon}(\boldsymbol{Hint\text{-}ATPC}^{\oplus(k_1,k_2)}_{d,w}) + k_1k_2d\log w.$$

Claim 5.6. *For $k_1, k_2, d, w \geqslant 1$ and $\epsilon \in [0, 1/2]$, it holds that*

$$\mathbf{D}^{(d),k_2}_{\mathcal{D}^{k_1k_2}_{d,w}, \frac{1}{2}-\epsilon+\Theta(\frac{1}{\sqrt{k_1k_2}})}(\boldsymbol{Hint\text{-}ATPC}^{\oplus(k_1,k_2)}_{d,w}) \leqslant \mathbf{D}^{(d),k_2}_{\mathcal{D}^{k_1k_2}_{d,w}, 1-\epsilon}(\boldsymbol{Hint\text{-}ATPC}^{\#(k_1,k_2)}_{d,w}).$$

We are now ready to prove Theorem 2.

Proof of Theorem 2. Fix a p-pass dynamic streaming algorithm for solving $\mathbf{MST}_n$ with probability 2/3 that has space S. Let $k = (n-1)/2$, $k_2 = cd\log n$ and $k_1 = k/k_2$ for some sufficiently large constant $c > 0$. Also let $d = 2p - 1$, $w = (n-k-1)^{1/d}$, and $C = k_2dS$. Applying the reduction of Claim 5.4, we get a d-round protocol for solving $\mathbf{Hint\text{-}ATPC}^{\#(k_1,k_2)}_{d,w}$ with probability 2/3 over $\mathcal{D}^k_{d,w}$ that has communication C. The success probability can be boosted to $1 - 1/\text{poly}(n)$ by $O(\log n)$ parallel repetitions.

On the other hand, Lemma 4.2, together with Theorem 3 for some sufficiently small constant $\epsilon > 0$, implies that

$$\mathbf{D}^{(d),k_2}_{\mathcal{D}^k_{d,w}, \frac{1}{2}+\frac{1}{\text{poly}(n)}}(\mathbf{ATPC}^{\oplus(k_1,k_2)}_{d,w}) = \Omega\left(\frac{k_1w}{d^2}\right).$$

Applying Claims 5.5 and 5.6 in sequence, we further get

$$\mathbf{D}^{(d),k_2}_{\mathcal{D}^k_{d,w}, 1-\frac{1}{\text{poly}(n)}}(\mathbf{Hint\text{-}ATPC}^{\#(k_1,k_2)}_{d,w}) = \Omega\left(\frac{k_1w}{d^2}\right).$$

Combining the above arguments, we finally have

$$C\log n = \Omega\left(\frac{k_1w}{d^2}\right).$$

The theorem follows by rearranging the terms. ∎

We remark that the above proof uses the ϵ_2 case in Lemma 5.2 with $\epsilon = \Theta(1)$. It is also possible to prove Theorem 2 using the ϵ_1 case with $\epsilon = \Theta(1/r)$. However, this results in slightly worse dependence on p for the derived space lower bound on streaming algorithms. Both cases of Lemma 5.2 are provided just in case the result may be of independent interest to some readers.

Acknowledgement

Sepehr Assadi is supported in part by an Alfred P. Sloan Fellowship, a University of Waterloo startup grant, an NSF CAREER grant CCF-2047061, and a gift from Google Research. Gillat Kol is supported by a National Science Foundation CAREER award CCF-1750443 and by a BSF grant No. 2018325.

References

[ACG$^+$15] Kook Jin Ahn, Graham Cormode, Sudipto Guha, Andrew McGregor, and Anthony Wirth. Correlation clustering in data streams. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2237–2246. JMLR.org, 2015. 1

[AGM12a] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Analyzing graph structure via linear measurements. In *Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '12, pages 459–467. SIAM, 2012. 1, 2, 8, 10

[AGM12b] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Graph sketches: sparsification, spanners, and subgraphs. In *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2012, Scottsdale, AZ, USA, May 20-24, 2012*, pages 5–14, 2012. 1, 2

[AGM13] Kook Jin Ahn, Sudipto Guha, and Andrew McGregor. Spectral sparsification in dynamic graph streams. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques - 16th International Workshop, APPROX 2013, and 17th International Workshop, RANDOM 2013, Berkeley, CA, USA, August 21-23, 2013. Proceedings*, pages 1–10, 2013. 1, 2

[AHLW16] Yuqing Ai, Wei Hu, Yi Li, and David P. Woodruff. New characterizations in turnstile streams with applications. In *31st Conference on Computational Complexity, CCC 2016, May 29 to June 1, 2016, Tokyo, Japan*, pages 20:1–20:22, 2016. 1

[AKL16] Sepehr Assadi, Sanjeev Khanna, and Yang Li. Tight bounds for single-pass streaming complexity of the set cover problem. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 698–711, 2016. 2

[AKL17] Sepehr Assadi, Sanjeev Khanna, and Yang Li. On estimating maximum matching size in graph streams. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1723–1742, 2017. 2

[AKLY16] Sepehr Assadi, Sanjeev Khanna, Yang Li, and Grigory Yaroslavtsev. Maximum matchings in dynamic graph streams and the simultaneous communication model. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 1345–1364, 2016. 1

[AMS96] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. In *STOC*, pages 20–29. ACM, 1996. 1

[AN21] Sepehr Assadi and Vishvajeet N. Graph streaming lower bounds for parameter estimation and property testing via a streaming XOR lemma. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 612–625. ACM, 2021. 11

[BBCR13] Boaz Barak, Mark Braverman, Xi Chen, and Anup Rao. How to compress interactive communication. *SIAM Journal on Computing*, 42(3):1327–1363, 2013. 2, 32

[BHNT15] Sayan Bhattacharya, Monika Henzinger, Danupon Nanongkai, and Charalampos E. Tsourakakis. Space- and time-efficient algorithm for maintaining dense subgraphs on one-pass dynamic streams. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 173–182, 2015. 1

[BRWY13] Mark Braverman, Anup Rao, Omri Weinstein, and Amir Yehudayoff. Direct product via round-preserving compression. In Fedor V. Fomin, Rusins Freivalds, Marta Z. Kwiatkowska, and David Peleg, editors, *Automata, Languages, and Programming - 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part I*, volume 7965 of *Lecture Notes in Computer Science*, pages 232–243. Springer, 2013. 5, 11

[BS15] Marc Bury and Chris Schwiegelshohn. Sublinear estimation of weighted matchings in dynamic data streams. In *Algorithms - ESA 2015 - 23rd Annual European Symposium, September 14-16, 2015, Proceedings*, pages 263–274, 2015. 1

[CCE+16] Rajesh Chitnis, Graham Cormode, Hossein Esfandiari, MohammadTaghi Hajiaghayi, Andrew McGregor, Morteza Monemizadeh, and Sofya Vorotnikova. Kernelization via sampling with applications to finding matchings and related problems in dynamic graph streams. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, January 10-12, 2016*, pages 1326–1344, 2016. 1

[CCF02] Moses Charikar, Kevin C. Chen, and Martin Farach-Colton. Finding frequent items in data streams. In Peter Widmayer, Francisco Triguero Ruiz, Rafael Morales Bueno, Matthew Hennessy, Stephan J. Eidenbenz, and Ricardo Conejo, editors, *Automata, Languages and Programming, 29th International Colloquium, ICALP 2002, Malaga, Spain, July 8-13, 2002, Proceedings*, volume 2380 of *Lecture Notes in Computer Science*, pages 693–703. Springer, 2002. 1

[CCHM15] Rajesh Hemant Chitnis, Graham Cormode, Mohammad Taghi Hajiaghayi, and Morteza Monemizadeh. Parameterized streaming: Maximal matching and vertex cover. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 1234–1251, 2015. 1

[CKL22] Yu Chen, Sanjeev Khanna, and Huan Li. On weighted graph sparsification by linear sketching. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 474–485. IEEE, 2022. 1

[CT06] Thomas M. Cover and Joy A. Thomas. *Elements of information theory (2. ed.)*. Wiley, 2006. 30

[DK20] Jacques Dark and Christian Konrad. Optimal lower bounds for matching and vertex cover in dynamic graph streams. In Shubhangi Saraf, editor, *35th Computational Complexity Conference, CCC 2020, July 28-31, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 169 of *LIPIcs*, pages 30:1–30:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. 1

[FIS05] Gereon Frahling, Piotr Indyk, and Christian Sohler. Sampling in dynamic data streams and applications. In Joseph S. B. Mitchell and Günter Rote, editors, *Proceedings of the 21st ACM Symposium on Computational Geometry, Pisa, Italy, June 6-8, 2005*, pages 142–149. ACM, 2005. 1

[FKM$^+$05] Joan Feigenbaum, Sampath Kannan, Andrew McGregor, Siddharth Suri, and Jian Zhang. On graph problems in a semi-streaming model. *Theor. Comput. Sci.*, 348(2-3):207–216, 2005. 1

[FKN21] Arnold Filtser, Michael Kapralov, and Navid Nouri. Graph spanners by sketching in dynamic streams and the simultaneous communication model. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 1894–1913. SIAM, 2021. 1

[GMT15] Sudipto Guha, Andrew McGregor, and David Tench. Vertex and hyperedge connectivity in dynamic graph streams. In *Proceedings of the 34th ACM Symposium on Principles of Database Systems, PODS 2015, Melbourne, Victoria, Australia, May 31 - June 4, 2015*, pages 241–247, 2015. 1

[HP16] Zengfeng Huang and Pan Peng. Dynamic graph stream algorithms in o(n) space. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 18:1–18:16, 2016. 1

[JPY12] Rahul Jain, Attila Pereszlényi, and Penghui Yao. A direct product theorem for the two-party bounded-round public-coin communication complexity. In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, October 20-23, 2012*, pages 167–176, 2012. 5, 11, 24, 32

[JST11] Hossein Jowhari, Mert Sağlam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In *Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 49–58. ACM, 2011. 2

[KLM$^+$14] Michael Kapralov, Yin Tat Lee, Cameron Musco, Christopher Musco, and Aaron Sidford. Single pass spectral sparsification in dynamic streams. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 561–570, 2014. 1, 2

[KNP$^+$17] Michael Kapralov, Jelani Nelson, Jakub Pachocki, Zhengyu Wang, David P. Woodruff, and Mobin Yahyazadeh. Optimal lower bounds for universal relation, and for samplers and finding duplicates in streams. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 475–486. IEEE Computer Society, 2017. 2, 5

[KRW95] Mauricio Karchmer, Ran Raz, and Avi Wigderson. Super-logarithmic depth lower bounds via the direct sum in communication complexity. *Computational Complexity*, 5(3/4):191–204, 1995. 5

[LNW14] Yi Li, Huy L. Nguyen, and David P. Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 174–183, 2014. 1

[MNSW98] Peter Bro Miltersen, Noam Nisan, Shmuel Safra, and Avi Wigderson. On data structures and asymmetric communication complexity. *J. Comput. Syst. Sci.*, 57(1):37–49, 1998. 5

[MTVV15] Andrew McGregor, David Tench, Sofya Vorotnikova, and Hoa T. Vu. Densest subgraph in dynamic graph streams. In *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part II*, pages 472–482, 2015. 1

[NY19] Jelani Nelson and Huacheng Yu. Optimal lower bounds for distributed and streaming spanning forest computation. In Timothy M. Chan, editor, *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1844–1860. SIAM, 2019. 2, 4, 5

[Tsy09] Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer series in statistics. Springer, 2009. 32

[Yu21] Huacheng Yu. Tight distributed sketching lower bound for connectivity. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 1856–1873. SIAM, 2021. 2, 4, 8

[Yu22] Huacheng Yu. Strong XOR lemma for communication with bounded rounds. In *Proceedings of the 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1186–1192, 2022. 3, 9, 10, 14, 21, 23, 24

Appendix

A Basic Tools From Information Theory

We introduce some definitions from information theory that are needed in this paper. For a random variable A, we use $\text{supp}(\mathsf{A})$ to denote the support of A and $\text{dist}(\mathsf{A})$ to denote its distribution. When it is clear from context, we may abuse the notation and use A directly instead of $\text{dist}(\mathsf{A})$, e.g., write $A \sim \mathsf{A}$ to mean $A \sim \text{dist}(\mathsf{A})$, i.e., A is sampled from the distribution of the random variable A.

We denote the *Shannon entropy* of a random variable A by $\mathbb{H}(\mathsf{A})$, which is defined as:

$$\mathbb{H}(\mathsf{A}) = \sum_{A \in \text{supp}(\mathsf{A})} \Pr\left(\mathsf{A} = A\right) \cdot \log \frac{1}{\Pr\left(\mathsf{A} = A\right)}.$$

The *conditional entropy* of A conditioned on B is denoted by $\mathbb{H}(\mathsf{A} \mid \mathsf{B})$ and defined as:

$$\mathbb{H}(\mathsf{A} \mid \mathsf{B}) = \mathop{\mathbb{E}}_{B \sim \mathsf{B}} \left[\mathbb{H}(\mathsf{A} \mid \mathsf{B} = B)\right],$$

where $\mathbb{H}(\mathsf{A} \mid \mathsf{B} = B)$ is defined in a standard way by using the distribution of A conditioned on the event $\mathsf{B} = B$ in the previous equation. We denote the *mutual information* between two random variables A and B is by $\mathbb{I}(\mathsf{A} \,; \mathsf{B})$, which is defined as:

$$\mathbb{I}(\mathsf{A} \,; \mathsf{B}) = \mathbb{H}(\mathsf{A}) - \mathbb{H}(\mathsf{A} \mid \mathsf{B}) = \mathbb{H}(\mathsf{B}) - \mathbb{H}(\mathsf{B} \mid \mathsf{A}).$$

The *conditional mutual information* $\mathbb{I}(\mathsf{A} \,; \mathsf{B} \mid \mathsf{C})$ is defined to be $\mathbb{H}(\mathsf{A} \mid \mathsf{C}) - \mathbb{H}(\mathsf{A} \mid \mathsf{B}, \mathsf{C})$ and hence by linearity of expectation:

$$\mathbb{I}(\mathsf{A} \,; \mathsf{B} \mid \mathsf{C}) = \mathop{\mathbb{E}}_{C \sim \mathsf{C}} \left[\mathbb{I}(\mathsf{A} \,; \mathsf{B} \mid \mathsf{C} = C)\right].$$

We refer the interested readers to the excellent textbook by Cover and Thomas [CT06] for an introduction to the field of information theory.

A.1 Useful Properties of Entropy and Mutual Information

We use the following basic properties of entropy and mutual information throughout.

Fact A.1 (cf. [CT06])**.** *Let* A, B, C, *and* D *be four (possibly correlated) random variables.*

1. $0 \leqslant \mathbb{H}(\mathsf{A}) \leqslant \log|\text{supp}(\mathsf{A})|$. *The right equality holds iff* $\text{dist}(\mathsf{A})$ *is uniform.*

2. $\mathbb{I}(\mathsf{A} \,; \mathsf{B} \mid \mathsf{C}) \geqslant 0$. *The equality holds iff* A *and* B *are* independent *conditioned on* C.

3. Conditioning on a random variable reduces entropy*:* $\mathbb{H}(\mathsf{A} \mid \mathsf{B}, \mathsf{C}) \leqslant \mathbb{H}(\mathsf{A} \mid \mathsf{B})$. *The equality holds iff* $\mathsf{A} \perp \mathsf{C} \mid \mathsf{B}$.

4. Subadditivity of entropy*:* $\mathbb{H}(\mathsf{A}, \mathsf{B} \mid \mathsf{C}) \leqslant \mathbb{H}(\mathsf{A} \mid C) + \mathbb{H}(\mathsf{B} \mid \mathsf{C})$.

5. Chain rule for entropy*:* $\mathbb{H}(\mathsf{A}, \mathsf{B} \mid \mathsf{C}) = \mathbb{H}(\mathsf{A} \mid \mathsf{C}) + \mathbb{H}(\mathsf{B} \mid \mathsf{C}, \mathsf{A})$.

6. Chain rule for mutual information*:* $\mathbb{I}(\mathsf{A}, \mathsf{B} \,; \mathsf{C} \mid \mathsf{D}) = \mathbb{I}(\mathsf{A} \,; \mathsf{C} \mid \mathsf{D}) + \mathbb{I}(\mathsf{B} \,; \mathsf{C} \mid \mathsf{A}, \mathsf{D})$.

7. Data processing inequality*: for a function* $f(\mathsf{A}, \mathsf{C})$, $\mathbb{I}(f(\mathsf{A}, \mathsf{C}) \,; \mathsf{B} \mid \mathsf{C}) \leqslant \mathbb{I}(\mathsf{A} \,; \mathsf{B} \mid \mathsf{C})$.

A.2 Measures of Distance Between Distributions

We also use the following standard measures of distance (or divergence) between distributions.

KL-divergence. For two distributions μ and ν, the *Kullback-Leibler divergence* between μ and ν is denoted by $\mathbb{D}(\mu \mid\mid \nu)$ and defined as:

$$\mathbb{D}(\mu \mid\mid \nu) = \mathop{\mathbb{E}}_{a \sim \mu}\left[\log \frac{\mu(a)}{\nu(a)}\right].$$

The *conditional KL-divergence* $\mathbb{D}(\mu(\mathsf{A} \mid \mathsf{B}) \mid\mid \nu(\mathsf{A} \mid \mathsf{B}))$ between two conditional distributions $\mu(\mathsf{A} \mid \mathsf{B})$ and $\nu(\mathsf{A} \mid \mathsf{B})$ is defined to be:

$$\mathbb{D}(\mu(\mathsf{A} \mid \mathsf{B}) \mid\mid \nu(\mathsf{A} \mid \mathsf{B})) = \mathop{\mathbb{E}}_{b \sim \mu(\mathsf{B})}\left[\mathbb{D}(\mu(\mathsf{A} \mid \mathsf{B} = b) \mid\mid \nu(\mathsf{A} \mid \mathsf{B} = b))\right].$$

We use the following basic properties of KL-divergence.

Fact A.2. *Suppose μ is a distribution and $\mathcal{E}$ is an event, then,*

$$\mathbb{D}(\mu \mid \mathcal{E} \mid\mid \mu) \leqslant \log\left(\frac{1}{\mu(\mathcal{E})}\right).$$

The following states the relation between mutual information and KL-divergence.

Fact A.3. *For random variables* $\mathsf{A}, \mathsf{B}, \mathsf{C}$*,*

$$\mathbb{I}(\mathsf{A} ; \mathsf{B} \mid \mathsf{C}) = \mathop{\mathbb{E}}_{(b,c) \sim (\mathsf{B},\mathsf{C})}\left[\mathbb{D}(\text{dist}(\mathsf{A} \mid \mathsf{B} = b, \mathsf{C} = c) \mid\mid \text{dist}(\mathsf{A} \mid \mathsf{C} = c))\right].$$

We also use the following chain rule of KL-divergence.

Fact A.4. *Suppose μ and ν are two distributions for* A, B*, then,*

$$\mathbb{D}(\mu(\mathsf{A}, \mathsf{B}) \mid\mid \nu(\mathsf{A}, \mathsf{B})) = \mathbb{D}(\mu(\mathsf{A}) \mid\mid \nu(\mathsf{A})) + \mathbb{D}(\mu(\mathsf{B} \mid \mathsf{A}) \mid\mid \nu(\mathsf{B} \mid \mathsf{A})).$$

Total variation distance. We denote the total variation distance between two distributions μ and ν on the same support Ω by $\|\mu - \nu\|_{\text{tvd}}$, defined as:

$$\|\mu - \nu\|_{\text{tvd}} = \max_{\Omega' \subseteq \Omega}\left(\mu(\Omega') - \nu(\Omega')\right) = \frac{1}{2} \cdot \sum_{x \in \Omega} |\mu(x) - \nu(x)|.$$

We use the following basic properties of total variation distance.

Fact A.5. *Suppose μ and ν are two distributions for a non-negative random variable* A*, then,*

$$\left|\mathop{\mathbb{E}}_{\mu}[\mathsf{A}] - \mathop{\mathbb{E}}_{\nu}[\mathsf{A}]\right| \leqslant \|\mu - \nu\|_{\text{tvd}} \cdot \max_{a \in \text{supp}(\mathsf{A})} a.$$

The total variation distance between two distributions can be bounded in terms of their KL-divergence by Pinsker's inequality.

Fact A.6 (Pinsker's inequality)**.** *For distributions μ and ν,*

$$\|\mu - \nu\|_{\text{tvd}} \leqslant \sqrt{\frac{1}{2} \cdot \mathbb{D}(\mu \mid\mid \nu)}.$$

An alternative bound for Pinsker's inequality is used for distributions with large KL-divergence.

Fact A.7 ([Tsy09], Equation 2.25)**.** *For distributions μ and ν,*

$$\|\mu - \nu\|_{\mathrm{tvd}} \leqslant 1 - \frac{1}{2} \cdot \exp(-\mathbb{D}(\mu \mid\mid \nu)).$$

We also use the following chain rule of total variation distance.

Fact A.8. *Suppose μ and ν are two distributions for A, B, then,*

$$\|\mu(\mathsf{A},\mathsf{B}) - \nu(\mathsf{A},\mathsf{B})\|_{\mathrm{tvd}} \leqslant \|\mu(\mathsf{A}) - \nu(\mathsf{A})\|_{\mathrm{tvd}} + \mathop{\mathbb{E}}_{A \sim \mu(\mathsf{A})} \left[\|\mu(\mathsf{B} \mid \mathsf{A} = A) - \nu(\mathsf{B} \mid \mathsf{A} = A)\|_{\mathrm{tvd}}\right].$$

B Missing Proofs in Section 5

We denote the *bias* of a Boolean random variable A by $\mathrm{bias}(\mathsf{A})$, which is defined as

$$\mathrm{bias}(\mathsf{A}) = |\Pr(\mathsf{A} = 0) - \Pr(\mathsf{A} = 1)|.$$

We use the following basic property regarding the biases of independent Boolean random variables.

Fact B.1. *For independent Boolean random variables A, B,*

$$\mathrm{bias}(\mathsf{A} \oplus \mathsf{B}) = \mathrm{bias}(\mathsf{A}) \cdot \mathrm{bias}(\mathsf{B}).$$

B.1 Proof of Lemma 5.1

We use (simplified versions of) Theorem 5.1 in [BBCR13][11] and Lemma 3.4 in [JPY12].

Lemma B.2 ([BBCR13], Theorem 5.1)**.** *For $k, r \geqslant 1$, $\epsilon \in [0, 1]$, Boolean function f, and input distribution μ, there exists an r-round protocol π for solving f with probability ϵ over μ that has information cost $\mathbf{IC}_\mu(\pi) \leqslant \mathbf{D}^{(r)}_{\mu^k,\epsilon}(f^{\oplus k})/k + 2$.*

Lemma B.3 ([JPY12], Lemma 3.4)**.** *For $r \geqslant 1$, $\epsilon \in (0, 1)$, and input distribution μ, any r-round protocol π can be ϵ-simulated (in r rounds) with communication $O(r/\epsilon \cdot \mathbf{IC}_\mu(\pi) + r^2/\epsilon)$.*

Proof of Lemma 5.1. By Lemma B.2, there exists an r-round protocol π for solving f with probability ϵ over μ that has information cost $\mathbf{IC}_\mu(\pi) \leqslant \mathbf{D}^{(r)}_{\mu^k,\epsilon}(f^{\oplus k})/k + 2$. Furthermore, π can be δ-simulated with communication $O(r/(\delta k) \cdot \mathbf{D}^{(r)}_{\mu^k,\epsilon}(f^{\oplus k}) + r^2/\delta)$ by Lemma B.3. The same bound holds for $\mathbf{D}^{(r)}_{\mu,\epsilon-\delta}(f)$. Rearranging the terms concludes the proof. ∎

B.2 Proof of Lemma 5.2

This section constitutes a proof of Lemma 5.2. Fix an r-round, $2k$-party protocol π for solving $f^{\oplus k}$ over μ^k that has communication C. Suppose for the sake of contradiction that $C < \mathbf{D}^{(r)}_{\mu,1/2+\epsilon}(f)$. In the following, we show that π succeeds with probability less than $1/2 + \min(\epsilon_1, \epsilon_2)$ over μ^k, hence proving the lemma. Without loss of generality, assume π is deterministic.

We first introduce some random variables with respect to π when input is drawn from μ^k.

[11] Technically, Theorem 5.1 of [BBCR13] is for unbounded-round protocols. However, the bounded-round version also holds since the simulation protocol in their proof is round-preserving.

- X_i: The input to Alice i for $i \in [k]$;
- Y_i: The input to Bob i for $i \in [k]$;
- $\mathsf{M}_i^{(j)}$: The $\lceil j/2 \rceil$-th message posted by Alice i if j is odd, and the $\lceil j/2 \rceil$-th message posted by Bob i if j is even, for $i \in [k]$ and $j \in [r+1]$;
- $\mathsf{B}_i^{(j)}$: The blackboard after $\mathsf{M}_i^{(j)}$ is posted for $i \in [k]$ and $j \in [r+1]$.

For convenience, we slightly abuse the notation and write $\mathsf{B}_k^{(0)} = \perp$ and $\mathsf{B}_0^{(j)} = \mathsf{B}_k^{(j-1)}$ for $j \in [r+1]$. Let $\mathsf{B} = \mathsf{B}_k^{(r+1)}$.[12] For blackboard B, $\mathrm{bias}(B)$ is defined to be $\mathrm{bias}(f^{\oplus k}(\mathsf{X}, \mathsf{Y}) \mid \mathsf{B} = B)$, and for $i \in [k]$, define $\mathrm{bias}_i(B)$ as $\mathrm{bias}(f(\mathsf{X}_i, \mathsf{Y}_i) \mid \mathsf{B} = B)$. A couple of events are considered as well.

- $\mathcal{E}_1(i, B)$: The event $\mathrm{bias}_i(B) \geqslant \sqrt{\epsilon}$ for $i \in [k]$ and blackboard B;
- $\mathcal{E}_1(S, B)$: The event $\bigwedge_{i \in S} \mathcal{E}_1(i, B)$ for $S \subseteq [k]$ and blackboard B;
- $\mathcal{E}_1(B)$: The event that there exists $S \subseteq [k]$ of size $(1 - 1/(10r)) \cdot k$ such that $\mathcal{E}_1(S, B)$ holds, for blackboard B;
- $\mathcal{E}_2(i, B)$: The event $\mathrm{bias}_i(B) \geqslant \epsilon^{1/(2r)}$ for $i \in [k]$ and blackboard B;
- $\mathcal{E}_2(S, B)$: The event $\bigwedge_{i \in S} \mathcal{E}_2(i, B)$ for $S \subseteq [k]$ and blackboard B;
- $\mathcal{E}_2(B)$: The event that there exists $S \subseteq [k]$ of size $(1 - \epsilon) \cdot k$ such that $\mathcal{E}_2(S, B)$ holds, for blackboard B.

We will use the following basic properties of the protocol π. Specifically, Claim B.4 proves a rectangle property and Claim B.5 bounds the success probability of π in terms of biases.

Claim B.4. *For $i \in [k]$, $j \in [r+1]$, and* fixed *blackboard $\mathsf{B}_i^{(j)}$, it holds that*

$$\mathrm{dist}(\mathsf{X}, \mathsf{Y} \mid \mathsf{B}_i^{(j)}) = \bigtimes_{i' \in [k]} \mathrm{dist}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \mid \mathsf{B}_i^{(j)}).$$

Proof. The claim can be equivalently stated as that for $i, i' \in [k]$ and $j \in [r+1]$, it holds that

$$\mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} ; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}_i^{(j)}) = 0.$$

To this end, fix $i, i' \in [k]$ and $j \in [r+1]$. Suppose $i' = i$. Observe that

$$\begin{aligned}
\mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} ; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}_i^{(j)}) &\leqslant \mathbb{I}(\mathsf{M}_i^{(j)}, \mathsf{X}_i, \mathsf{Y}_i ; \mathsf{X}_{-i}, \mathsf{Y}_{-i} \mid \mathsf{B}_{i-1}^{(j)}) && (\text{as } \mathsf{B}_i^{(j)} = (\mathsf{B}_{i-1}^{(j)}, \mathsf{M}_i^{(j)})) \\
&= \mathbb{I}(\mathsf{X}_i, \mathsf{Y}_i ; \mathsf{X}_{-i}, \mathsf{Y}_{-i} \mid \mathsf{B}_{i-1}^{(j)}). && (\text{as } \mathsf{M}_i^{(j)} \text{ is a function of } \mathsf{B}_{i-1}^{(j)}, \mathsf{X}_i, \mathsf{Y}_i)
\end{aligned}$$

For $i' \neq i$, also observe that

$$\mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} ; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}_i^{(j)}) \leqslant \mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} ; \mathsf{M}_i^{(j)}, \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}_{i-1}^{(j)}) \qquad (\text{as } \mathsf{B}_i^{(j)} = (\mathsf{B}_{i-1}^{(j)}, \mathsf{M}_i^{(j)}))$$

[12] Recall that for consistency with the standard 2-party model, the multi-party model is defined so that the last party, who returns an output, does not post a message to the blackboard. Thus, there is an "$(r+1)$-th round" for an r-round protocol. To minimize confusion, we avoid any reference to the "j-th round" throughout.

$$= \mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \,; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}^{(j)}_{i-1}).$$

(as $\mathsf{M}^{(j)}_i$ is a function of $\mathsf{B}^{(j)}_{i-1}, \mathsf{X}_{-i'}, \mathsf{Y}_{-i'}$)

In both cases, we get

$$\mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \,; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}^{(j)}_{i}) \leqslant \mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \,; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}^{(j)}_{i-1}).$$

Consequently, an induction in increasing order of (j, i) implies that

$$\mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \,; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'} \mid \mathsf{B}^{(j)}_{i}) \leqslant \mathbb{I}(\mathsf{X}_{i'}, \mathsf{Y}_{i'} \,; \mathsf{X}_{-i'}, \mathsf{Y}_{-i'}).$$

The claim follows as $\mathsf{X}_{i'}, \mathsf{Y}_{i'} \perp \mathsf{X}_{-i'}, \mathsf{Y}_{-i'}$. ▮

Claim B.5. *The protocol π succeeds with probability at most*

$$\frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \mathrm{bias}_i(\mathsf{B})\right].$$

Proof. Observe that given blackboard B, since the protocol π is deterministic, it succeeds with probability at most

$$\max_{b \in \{0,1\}} \Pr(f^{\oplus k}(\mathsf{X}, \mathsf{Y}) = b \mid \mathsf{B} = B) = \frac{1}{2} + \frac{1}{2} \cdot \mathrm{bias}(B).$$

Summing over B, we get that the success probability of π is upper bounded by

$$\begin{aligned} \frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}[\mathrm{bias}(\mathsf{B})] &= \frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\mathrm{bias}(f^{\oplus k}(\mathsf{X}, \mathsf{Y}) \mid \mathsf{B})\right] \\ &= \frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\mathrm{bias}(\bigoplus_{i \in [k]} f(\mathsf{X}_i, \mathsf{Y}_i) \mid \mathsf{B})\right] \\ &= \frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \mathrm{bias}(f(\mathsf{X}_i, \mathsf{Y}_i) \mid \mathsf{B})\right] && \text{(by Fact B.1 and Claim B.4)} \\ &= \frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \mathrm{bias}_i(\mathsf{B})\right], \end{aligned}$$

as claimed. ▮

Claim B.6. *For $S \subseteq [k]$, it holds that*

$$\Pr_{\mathsf{B}}(\mathcal{E}_1(S, \mathsf{B})) < (4\sqrt{\epsilon})^{\frac{|S|}{r+2}},$$

and

$$\Pr_{\mathsf{B}}(\mathcal{E}_2(\mathsf{B})) < (6\epsilon^{1-\frac{1}{2r}})^{\frac{k}{r+2}}.$$

The proof of Lemma 5.2 is done with the help of the above technical claim. Assume its correctness for now. We show that it indeed implies Lemma 5.2. Let $\mathcal{B}_1$ be the subset of blackboards B such that $\mathcal{E}_1(B)$ holds. By Claim B.5, the success probability of π is at most

$$\begin{aligned}
&\frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \text{bias}_i(\mathsf{B})\right] \\
&\quad= \frac{1}{2} + \frac{1}{2} \cdot \left[\sum_{B \in \mathcal{B}_1} \Pr(\mathsf{B} = B) \cdot \prod_{i \in [k]} \text{bias}_i(B) + \sum_{B \notin \mathcal{B}_1} \Pr(\mathsf{B} = B) \cdot \prod_{i \in [k]} \text{bias}_i(B)\right] \\
&\quad\leqslant \frac{1}{2} + \frac{1}{2} \cdot \left[\Pr_{\mathsf{B}}(\mathcal{E}_1(\mathsf{B})) + \sum_{B \notin \mathcal{B}_1} \Pr(\mathsf{B} = B) \cdot \prod_{i \in [k]} \text{bias}_i(B)\right] && (\text{as } \text{bias}_i(B) \in [0,1]) \\
&\quad\leqslant \frac{1}{2} + \frac{1}{2} \cdot \left[\Pr_{\mathsf{B}}(\mathcal{E}_1(\mathsf{B})) + (\sqrt{\epsilon})^{\frac{k}{10r}}\right], && (\text{as } \Pr_{\mathsf{B}}(\overline{\mathcal{E}_1(\mathsf{B})}) \leqslant 1)
\end{aligned}$$

where we use the observation that for $B \notin \mathcal{B}_1$, $\text{bias}_i(B) < \sqrt{\epsilon}$ for more than $k/(10r)$ indices $i \in [k]$. By a union bound, we further upper bound the success probability of π by

$$\begin{aligned}
\frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \text{bias}_i(\mathsf{B})\right] &\leqslant \frac{1}{2} + \frac{1}{2} \cdot \left[\sum_{\substack{S \subseteq [k]: \\ |S| = (1 - \frac{1}{10r}) \cdot k}} \Pr_{\mathsf{B}}(\mathcal{E}_1(S, \mathsf{B})) + \epsilon^{\frac{k}{20r}}\right] \\
&< \frac{1}{2} + \frac{1}{2} \cdot \left[\binom{k}{(1 - \frac{1}{10r}) \cdot k} \cdot (4\sqrt{\epsilon})^{(1 - \frac{1}{10r}) \cdot k \cdot \frac{1}{r+2}} + \epsilon^{\frac{k}{20r}}\right] && (\text{by Claim B.6}) \\
&= \frac{1}{2} + \frac{1}{2} \cdot \left[\binom{k}{\frac{k}{10r}} \cdot (16\epsilon)^{\frac{10r-1}{20r} \cdot \frac{k}{r+2}} + \epsilon^{\frac{k}{20r}}\right] \\
&\leqslant \frac{1}{2} + \frac{1}{2} \cdot \left[\left(\frac{ek}{k/(10r)}\right)^{\frac{k}{10r}} \cdot (16\epsilon)^{\frac{10r-1}{20r} \cdot \frac{k}{r+2}} + \epsilon^{\frac{k}{20r}}\right] \\
&\leqslant \frac{1}{2} + \epsilon_1,
\end{aligned}$$

for sufficiently small $\epsilon < \epsilon_0$.

Meanwhile, let $\mathcal{B}_2$ be the subset of blackboards B such that $\mathcal{E}_2(B)$ holds. Similarly by Claim B.5, we can also upper bound the success probability of π by

$$\frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \text{bias}_i(\mathsf{B})\right] \leqslant \frac{1}{2} + \frac{1}{2} \cdot \left[\Pr_{\mathsf{B}}(\mathcal{E}_2(\mathsf{B})) + (\epsilon^{\frac{1}{2r}})^{\epsilon k}\right],$$

since for $B \notin \mathcal{B}_2$, $\text{bias}_i(B) < \epsilon^{1/(2r)}$ for more than ϵk indices $i \in [k]$. By Claim B.6, the success probability of π is at most

$$\frac{1}{2} + \frac{1}{2} \cdot \mathop{\mathbb{E}}_{\mathsf{B}}\left[\prod_{i \in [k]} \text{bias}_i(\mathsf{B})\right] < \frac{1}{2} + \frac{1}{2} \cdot \left[(6\epsilon^{1 - \frac{1}{2r}})^{\frac{k}{r+2}} + \epsilon^{\frac{\epsilon k}{2r}}\right] \leqslant \frac{1}{2} + \epsilon_2,$$

for sufficiently small $\epsilon < \epsilon_0$. This concludes the proof of Lemma 5.2. It now remains to show the correctness of Claim B.6. Suppose not, we will construct an r-round, 2-party protocol τ for solving f with probability $1/2+\epsilon$ over μ that has communication C, contradicting the assumption $C < \mathbf{D}^{(r)}_{\mu,1/2+\epsilon}(f)$. On input (X,Y), τ will simulate π as shown in Algorithm 4, where all random variables are with respect to π. The parameters $T \subseteq [k]$ and event $\mathcal{E}$ are to be determined and are supposed to be such that $\mathrm{bias}_{\mathsf{I}}(\mathsf{B})$ is large with high probability, conditioned on $\mathcal{E}$, for $\mathsf{I} \sim T$.

```
Algorithm 4. The protocol $\tau$ for solving $f$ on input $(X,Y)$.
Parameters: $T \subseteq [k]$ and event $\mathcal{E}$.

  1. Alice and Bob publicly sample $\mathsf{I}$ uniformly at random from $T$.
  2. Alice and Bob publicly sample $\mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}$ conditioned on $\mathsf{I}, \mathcal{E}$.
  3. Alice sets $\mathsf{X}_{\mathsf{I}} = X$ and Bob sets $\mathsf{Y}_{\mathsf{I}} = Y$.
  4. For $j \in [r+1]$, if $j$ is odd,
     (a) Alice simulates $\pi$ on $\mathsf{X}_{\leqslant \mathsf{I}}$ and computes $\mathsf{M}^{(j)}_{\leqslant \mathsf{I}}$, given $\mathsf{B}^{(j-1)}_{k}$.
     (b) Alice privately samples $\mathsf{M}^{(j)}_{>\mathsf{I}}$ conditioned on $\mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}$.
     (c) Alice sends $\mathsf{M}^{(j)}$ to Bob if $j \leqslant r$, and otherwise outputs
             $\arg\max_{b \in \{0,1\}} \Pr_{(x,y) \sim \mu^k | \mathsf{B}} (f(x_i, y_i) = b);$
     if $j$ is even,
     (a) Bob simulates $\pi$ on $\mathsf{Y}_{\leqslant \mathsf{I}}$ and computes $\mathsf{M}^{(j)}_{\leqslant \mathsf{I}}$, given $\mathsf{B}^{(j-1)}_{k}$.
     (b) Bob privately samples $\mathsf{M}^{(j)}_{>\mathsf{I}}$ conditioned on $\mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}$.
     (c) Bob sends $\mathsf{M}^{(j)}$ to Alice if $j \leqslant r$, and otherwise outputs
             $\arg\max_{b \in \{0,1\}} \Pr_{(x,y) \sim \mu^k | \mathsf{B}} (f(x_i, y_i) = b).$
```

It can be verified that the current transcript always reflects the up-to-date blackboard and thus τ is indeed an r-round protocol for solving f that has communication C. Regarding the success probability of τ, the following two technical claims show that all the random variables as sampled in τ almost perfectly follow their distribution in π conditioned on $\mathcal{E}$.

Claim B.7. *For $T \subseteq [k]$ and event $\mathcal{E}$, it holds that*

$$\mathop{\mathbb{E}}_{\mathsf{X}_{<\mathsf{I}},\mathsf{Y}_{<\mathsf{I}},\mathsf{I}|\mathcal{E}} \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \parallel \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}})) \leqslant \frac{1}{|T|} \cdot \log \frac{1}{\Pr(\mathcal{E})}.$$

Proof. Observe that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathsf{X}_{<\mathsf{I}},\mathsf{Y}_{<\mathsf{I}},\mathsf{I}|\mathcal{E}} \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \parallel \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}})) \\
&\quad = \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \parallel \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})) \qquad (\text{as } \mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \perp \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}) \\
&\quad = \frac{1}{|T|} \cdot \sum_{i \in T} \mathbb{D}(\mathrm{dist}(\mathsf{X}_{i}, \mathsf{Y}_{i} \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathsf{I} = i, \mathcal{E}) \parallel \mathrm{dist}(\mathsf{X}_{i}, \mathsf{Y}_{i} \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathsf{I} = i)) \\
&\qquad\qquad (\text{as } \mathsf{I} \text{ is uniform over } T)
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{|T|} \cdot \sum_{i \in T} \mathbb{D}(\mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i})) \\
&\hspace{20em} (\text{as } \mathsf{X}_{\leqslant i}, \mathsf{Y}_{\leqslant i} \perp \mathsf{I} = i \mid \mathcal{E} \text{ and } \mathsf{X}_{\leqslant i}, \mathsf{Y}_{\leqslant i} \perp \mathsf{I} = i) \\
&\leqslant \frac{1}{|T|} \cdot \sum_{i \in [k]} \mathbb{D}(\mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{X}_{<i}, \mathsf{Y}_{<i})) \qquad (\text{as } T \subseteq [k]) \\
&= \frac{1}{|T|} \cdot \mathbb{D}(\mathrm{dist}(\mathsf{X}, \mathsf{Y} \mid \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}, \mathsf{Y})) \qquad (\text{by chain rule of KL-divergence (Fact A.4)}) \\
&\leqslant \frac{1}{|T|} \cdot \log \frac{1}{\Pr(\mathcal{E})}, \qquad (\text{by Fact A.2})
\end{aligned}
$$

as claim. ∎

Claim B.8. *For $j \in [r+1]$, $T \subseteq [k]$, and event $\mathcal{E}$, it holds that*

$$
\mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \mathbb{D}(\mathrm{dist}(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})) \leqslant \frac{1}{|T|} \cdot \log \frac{1}{\Pr(\mathcal{E})}.
$$

Proof. Observe that

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \mathbb{D}(\mathrm{dist}(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})) \\
&\quad = \mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \log \frac{\Pr(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I}, \mathcal{E})}{\Pr(\mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})} \qquad (\text{as } \mathsf{B}^{(j)}_{k} = (\mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{M}^{(j)}_{>\mathsf{I}})) \\
&\quad = \mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \log \frac{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})}{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})} \qquad (\text{as } \tfrac{\Pr(A|B)}{\Pr(A)} = \tfrac{\Pr(B|A)}{\Pr(B)}) \\
&\quad = \mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \log \frac{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})}{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})} \\
&\qquad\quad - \mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \log \frac{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})}{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})} \qquad (\text{as } \log \tfrac{a}{b} = \log \tfrac{a}{c} - \log \tfrac{b}{c}) \\
&\quad = \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})) \\
&\qquad\quad - \mathop{\mathbb{E}}_{\mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant \mathsf{I}}, \mathsf{Y}_{\leqslant \mathsf{I}}, \mathsf{I} \mid \mathcal{E}} \log \frac{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E})}{\Pr(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})} \\
&\hspace{20em} (\text{as } \mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \perp \mathsf{M}^{(j)}_{>\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I} \text{ by Claim B.4}) \\
&\quad = \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})) \\
&\qquad\quad - \mathbb{D}(\mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_{\mathsf{I}}, \mathsf{Y}_{\mathsf{I}} \mid \mathsf{B}^{(j)}_{\mathsf{I}}, \mathsf{X}_{<\mathsf{I}}, \mathsf{Y}_{<\mathsf{I}}, \mathsf{I})) \\
&\quad \leqslant \frac{1}{|T|} \cdot \sum_{i \in T} \mathbb{D}(\mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathsf{I} = i, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathsf{I} = i)) \\
&\hspace{20em} (\text{as } \mathsf{I} \text{ is uniform over } T) \\
&\quad = \frac{1}{|T|} \cdot \sum_{i \in T} \mathbb{D}(\mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<i}, \mathsf{Y}_{<i}, \mathcal{E}) \mid\mid \mathrm{dist}(\mathsf{X}_i, \mathsf{Y}_i \mid \mathsf{B}^{(j)}_{k}, \mathsf{X}_{<i}, \mathsf{Y}_{<i})) \\
&\hspace{16em} (\text{as } \mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant i}, \mathsf{Y}_{\leqslant i} \perp \mathsf{I} = i \mid \mathcal{E} \text{ and } \mathsf{B}^{(j)}_{k}, \mathsf{X}_{\leqslant i}, \mathsf{Y}_{\leqslant i} \perp \mathsf{I} = i)
\end{aligned}
$$

$$\begin{aligned}
&\leqslant \frac{1}{|T|}\cdot\sum_{i\in[k]}\mathbb{D}(\mathrm{dist}(\mathsf{X}_i,\mathsf{Y}_i\mid \mathsf{B}_k^{(j)},\mathsf{X}_{<i},\mathsf{Y}_{<i},\mathcal{E})\mid\mid \mathrm{dist}(\mathsf{X}_i,\mathsf{Y}_i\mid \mathsf{B}_k^{(j)},\mathsf{X}_{<i},\mathsf{Y}_{<i})) && \text{(as } T\subseteq[k])\\
&= \frac{1}{|T|}\cdot\mathbb{D}(\mathrm{dist}(\mathsf{X},\mathsf{Y}\mid \mathsf{B}_k^{(j)},\mathcal{E})\mid\mid \mathrm{dist}(\mathsf{X},\mathsf{Y}\mid \mathsf{B}_k^{(j)}))\\
& && \text{(by chain rule of KL-divergence (Fact A.4))}\\
&\leqslant \frac{1}{|T|}\cdot\mathop{\mathbb{E}}_{\mathsf{B}_k^{(j)}\mid\mathcal{E}}\log\frac{1}{\Pr(\mathcal{E}\mid \mathsf{B}_k^{(j)})} && \text{(by Fact A.2)}\\
&\leqslant \frac{1}{|T|}\cdot\log\mathop{\mathbb{E}}_{\mathsf{B}_k^{(j)}\mid\mathcal{E}}\frac{1}{\Pr(\mathcal{E}\mid \mathsf{B}_k^{(j)})} && \text{(by concavity of } \log(\cdot))\\
&= \frac{1}{|T|}\cdot\log\sum_{\mathsf{B}_k^{(j)}}\frac{\Pr(\mathsf{B}_k^{(j)}\mid\mathcal{E})}{\Pr(\mathcal{E}\mid \mathsf{B}_k^{(j)})}\\
&= \frac{1}{|T|}\cdot\log\sum_{\mathsf{B}_k^{(j)}}\frac{\Pr(\mathsf{B}_k^{(j)})}{\Pr(\mathcal{E})} && \text{(as } \tfrac{\Pr(A|B)}{\Pr(B|A)}=\tfrac{\Pr(A)}{\Pr(B)})\\
&= \frac{1}{|T|}\cdot\log\frac{1}{\Pr(\mathcal{E})}.
\end{aligned}$$

This concludes the proof. ∎

We emphasize that Alice is able to compute $\mathsf{M}_{\leqslant\mathsf{I}}^{(j)}$ exactly for odd $j\in[r+1]$ as it is fully determined by $\mathsf{B}_k^{(j-1)}$, which is provided by the current transcript, and $\mathsf{X}_{\leqslant\mathsf{I}}$. Similarly, Bob is able to compute $\mathsf{M}_{\leqslant\mathsf{I}}^{(j)}$ exactly for even $j\in[r+1]$ since he has full knowledge of $\mathsf{B}_k^{(j)}$ and $\mathsf{Y}_{\leqslant\mathsf{I}}$. We are now ready to prove Claim B.6.

Proof of Claim B.6. Fix the input distribution μ. Let ν_π be the distribution of $\mathsf{B},\mathsf{X}_{\leqslant\mathsf{I}},\mathsf{Y}_{\leqslant\mathsf{I}},\mathsf{I}$ where I is drawn from T uniformly at random and $\mathsf{B},\mathsf{X}_{\leqslant\mathsf{I}},\mathsf{Y}_{\leqslant\mathsf{I}}$ follow their distribution in π conditioned on $\mathcal{E}$. Also let ν_τ be the distribution of $\mathsf{B},\mathsf{X}_{\leqslant\mathsf{I}},\mathsf{Y}_{\leqslant\mathsf{I}},\mathsf{I}$ as sampled in τ. By chain rule of KL-divergence (Fact A.4), together with Claims B.7 and B.8, we get that

$$\mathbb{D}(\nu_\pi\mid\mid\nu_\tau)\leqslant\frac{r+2}{|T|}\cdot\log\frac{1}{\Pr(\mathcal{E})}.$$

Using an alternative bound for Pinsker's inequality (Fact A.7), we further have

$$\|\nu_\pi-\nu_\tau\|_{\mathrm{tvd}}\leqslant 1-\frac{1}{2}\cdot\Pr(\mathcal{E})^{\frac{r+2}{|T|}}.$$

Fix a threshold value $t\in[0,1]$. Observe that the success probability of τ is

$$\begin{aligned}
\frac{1}{2}+\frac{1}{2}\cdot\mathop{\mathbb{E}}_{\mathsf{B},\mathsf{I}\sim\nu_\tau}[\mathrm{bias}_{\mathsf{I}}(\mathsf{B})] &\geqslant \frac{1}{2}+\frac{t}{2}\cdot\Pr_{\mathsf{B},\mathsf{I}\sim\nu_\tau}(\mathrm{bias}_{\mathsf{I}}(\mathsf{B})\geqslant t) && \text{(as } \mathrm{bias}_{\mathsf{I}}(\mathsf{B})\in[0,1])\\
&\geqslant \frac{1}{2}+\frac{t}{2}\cdot\left[\Pr_{\mathsf{B},\mathsf{I}\sim\nu_\pi}(\mathrm{bias}_{\mathsf{I}}(\mathsf{B})\geqslant t)-\|\nu_\pi-\nu_\tau\|_{\mathrm{tvd}}\right] && \text{(by Fact A.5)}\\
&\geqslant \frac{1}{2}+\frac{t}{2}\cdot\left[\Pr_{\mathsf{B},\mathsf{I}\sim\nu_\pi}(\mathrm{bias}_{\mathsf{I}}(\mathsf{B})\geqslant t)+\frac{1}{2}\cdot\Pr(\mathcal{E})^{\frac{r+2}{|T|}}-1\right].
\end{aligned}$$

Recall that τ has communication C and thus by assumption, it can only succeed with probability less than $1/2 + \epsilon$. Rearranging the terms above, we get

$$\Pr(\mathcal{E}) < \left[2 + \frac{4\epsilon}{t} - 2 \cdot \Pr_{\mathsf{B},\mathsf{I}\sim\nu_\pi}(\mathrm{bias}_{\mathsf{I}}(\mathsf{B}) \geqslant t)\right]^{\frac{|T|}{r+2}}.$$

For $S \subseteq [k]$, setting $T = S$, $\mathcal{E} = \mathcal{E}_1(S, \mathsf{B})$, and $t = \sqrt{\epsilon}$ implies

$$\begin{aligned}\Pr_{\mathsf{B}}(\mathcal{E}_1(S,\mathsf{B})) &< \left[2 + 4\sqrt{\epsilon} - 2 \cdot \Pr_{\substack{\mathsf{B}|\mathcal{E}_1(S,\mathsf{B}),\\ \mathsf{I}\sim S}}\left(\mathrm{bias}_{\mathsf{I}}(\mathsf{B}) \geqslant \sqrt{\epsilon}\right)\right]^{\frac{|S|}{r+2}}\\ &= \left(2 + 4\sqrt{\epsilon} - 2\cdot 1\right)^{\frac{|S|}{r+2}}\\ &= (4\sqrt{\epsilon})^{\frac{|S|}{r+2}},\end{aligned}$$

since conditioned on $\mathcal{E}_1(S, \mathsf{B})$, it always holds that $\mathrm{bias}_{\mathsf{I}}(\mathsf{B}) \geqslant \sqrt{\epsilon}$ for $\mathsf{I} \in S$. Meanwhile, setting $T = [k]$, $\mathcal{E} = \mathcal{E}_2(\mathsf{B})$, and $t = \epsilon^{1/(2r)}$ implies

$$\begin{aligned}\Pr_{\mathsf{B}}(\mathcal{E}_2(\mathsf{B})) &< \left[2 + 4\epsilon^{1-\frac{1}{2r}} - 2 \cdot \Pr_{\substack{\mathsf{B}|\mathcal{E}_2(\mathsf{B}),\\ \mathsf{I}\sim[k]}}\left(\mathrm{bias}_{\mathsf{I}}(\mathsf{B}) \geqslant \epsilon^{\frac{1}{2r}}\right)\right]^{\frac{k}{r+2}}\\ &\leqslant \left(2 + 4\epsilon^{1-\frac{1}{2r}} - 2\cdot(1-\epsilon)\right)^{\frac{k}{r+2}}\\ &\leqslant (6\epsilon^{1-\frac{1}{2r}})^{\frac{k}{r+2}}, \qquad \text{(as } \epsilon \leqslant \epsilon^{1-\frac{1}{2r}}\text{)}\end{aligned}$$

where in the second step, we use the fact that conditioned on $\mathcal{E}_2(\mathsf{B})$, at least a $1 - \epsilon$ fraction of indices $\mathsf{I} \in [k]$ satisfy $\mathrm{bias}_{\mathsf{I}}(\mathsf{B}) \geqslant \epsilon^{1/(2r)}$. This concludes the proof. ∎

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il