SIAM J. Comput.
Vol. 45, No. 1, pp. 174–196

LOWER BOUNDS FOR NUMBER-IN-HAND MULTIPARTY COMMUNICATION COMPLEXITY, MADE EASY[*]

JEFF M. PHILLIPS[†], ELAD VERBIN[‡], and QIN ZHANG[§]

Abstract. In this paper we prove lower bounds on randomized multiparty communication complexity, mainly in the *message-passing model*, where messages are sent player-to-player. Some of our results apply to the *blackboard model*, where each message is written on a blackboard for all players to see. We introduce a new technique for proving such bounds, called *symmetrization*, which is natural, intuitive, and often easy to use. For example, for the problem where each of k players gets a bit-vector of length n, and the goal is to compute the coordinatewise XOR of these vectors, we prove a tight lower bound of $\Omega(nk)$ in the blackboard model. For the same problem with AND instead of XOR, we prove a lower bound of roughly $\Omega(nk)$ in the message-passing model (assuming $k \leq n/3200$) and $\Omega(n \log k)$ in the blackboard model. We also prove lower bounds for bitwise majority, for a graph-connectivity problem, and for other problems; the technique seems applicable to a wide range of other problems as well. All of our lower bounds allow randomized communication protocols with two-sided error. We also use the symmetrization technique to prove several direct-sum-like results for multiparty communication.

Key words. multiparty communication complexity, number-in-hand, symmetrization

AMS subject classification. 68P01

DOI. 10.1137/15M1007525

1. Introduction.

In this work we consider multiparty communication complexity in the *number-in-hand* model. In this model, there are k players $\{p_1, \ldots, p_k\}$, each with his own n-bit input $x_i \in \{0,1\}^n$. The players wish to collaborate in order to compute a joint function of their inputs, $f(x_1, \ldots, x_k)$. To do so, they are allowed to communicate, until one of them figures out the value of $f(x_1, \ldots, x_k)$ and returns it. All players are assumed to have unlimited computational power, so all we care about is the amount of communication used. There are three variants to this model, according to the mode of communication:

1. the *blackboard model*, where any message sent by a player is written on a blackboard visible to all players;
2. the *message-passing model*, where a player p_i sending a message specifies another player p_j that will receive this message;
3. the *coordinator model*, where there is an additional $(k+1)$th player called the *coordinator*, who receives no input. Players can only communicate with the *coordinator*, and not with each other directly.

[*] Received by the editors February 9, 2015; accepted for publication (in revised form) December 14, 2015; published electronically February 25, 2016. A preliminary version of the paper was presented at the ACM-SIAM Symposium on Discrete Algorithms, January 2012.

http://www.siam.org/journals/sicomp/45-1/M100752.html

[†] School of Computing, University of Utah, Salt Lake City, UT 84112 (jeffp@cs.utah.edu). This author's work was supported by a subaward to the University of Utah under NSF award 0937060 to the Computing Research Association and NSF CCF-1350888, IIS-1251019, and ACI-1443046.

[‡] Institute for Theoretical Computer Science, Tsinghua University, Beijing 100084 (elad.verbin@gmail.com). This author's work was supported by the Danish National Research Foundation and the National Science Foundation of China (under grant 61061130540) for the Sino-Danish Center for the Theory of Interactive Computation, within which part of this work was performed.

[§] Corresponding author. Indiana University Bloomington, Bloomington, IN 47401 (qzhangcs@indiana.edu). This author's work was supported in part by NSF CCF-1525024, and IU's Office of the Vice Provost for Research through the Faculty Research Support Program.

We will work in all of these, but will mostly concentrate on the message-passing model and the coordinator model. Note that the coordinator model is almost equivalent to the message-passing model, up to a $\log k$ multiplicative factor, since instead of player i sending message x to player j, player i can transmit message (j, x) to the coordinator, and the coordinator forwards it to player j.

Lower bounds in the three models above are useful for proving lower bounds on the space usage of streaming algorithms, and for other models as well, as we explain in section 1.2. Most previous lower bounds have been proved in the blackboard model, but lower bounds in the message-passing model and the coordinator model can potentially give higher bounds for all the applications.

Note that another, entirely different, model for multiparty communication is the *number-on-forehead* model, where each player can see the inputs of all other players but *not* his own input. This model has important applications for circuit complexity (see, e.g., [30]). We do not discuss this model in this paper.

We allow all protocols to be randomized, with public coins, i.e., all players have unlimited access to a common infinite string of independent random bits. We allow the protocol to return the wrong answer with probability ε (which should usually be thought of as a small constant); here, the probability is taken over the sample space of public randomness. Note that the public coin model might seem overly powerful, but in this paper we are mainly interested in proving lower bounds rather than upper bounds, so giving the model such strength only makes our results stronger.

For more on communication complexity, see the book of Kushilevitz and Nisan [30] and the references therein. We give some more definitions in the preliminaries in section 2.

1.1. Warm-up. We begin by sketching two lower bounds obtained using our symmetrization technique, both of them for the coordinatewise k-XOR problem. These lower bounds can be proved without using symmetrization, but their proofs that use symmetrization are particularly appealing.

First consider the following problem: Each player gets a bitvector $x_i \in \{0,1\}^n$ and the goal is to compute the coordinatewise XOR of these vectors. We operate in the *blackboard model*, where messages are posted on a blackboard for all to see.

THEOREM 1.1. *The coordinatewise k-XOR problem requires communication $\Omega(nk)$ in the blackboard model.*

To see this, first let us specify the *hard distribution*: we prove the lower bound when the input is drawn from this distribution, and by the easy direction of Yao's minimax lemma (see, e.g., [30]), it follows that this lower bound applies for the problem as a whole. The hard distribution we choose is just the distribution where the inputs are independently drawn from the uniform distribution.

To prove the lower bound, consider a protocol P for this k-player problem, which works on this distribution, communicates $C(P)$ bits in expectation, and suppose for now that it never makes any errors (it will be easy to remove this assumption). We build from P a new protocol P' for a 2-player problem. In the 2-player problem, suppose that Alice gets input x and Bob gets input y, where $x, y \in \{0,1\}^n$ are independent random bitvectors. Then P' works as follows: Alice and Bob randomly choose two distinct indices $i, j \in \{1, \ldots, k\}$ using the public randomness, and they simulate the protocol P, where Alice plays player i and lets $x_i = x$, Bob plays player j and lets $x_j = y$, and they both play all of the rest of the players; the inputs of the rest of the players are chosen from shared randomness. Alice and Bob begin simulating the running of P. Every time player i should speak, Alice sends to Bob the message that player i was supposed to write on the board, and vice versa. When any other player p_r ($r \neq i, j$) should speak, Alice and Bob both know the input of player p_r, so they know what he should be writing on the board, thus no communication is actually needed (this is the key point of the symmetrization technique). A key observation is that the inputs of the k players are uniform and independent and thus entirely symmetrical,[1] and since the indices i and j were chosen uniformly at random, the expected communication performed by the protocol P' is $\mathbf{E}[C(P')] = 2C(P)/k$. Furthermore, once the players have finished simulating P', Alice knows the bitwise XOR of all the vectors $x_1, \ldots, x_k$, from which she can easily reconstruct the vector x_j (since she already knows all the other vectors x_r for $r \neq j$). It follows that using $2C(P)/k$ expected communication, Alice has managed to reconstruct Bob's entire input; from an easy information-theoretic argument, it follows that $2C(P)/k \geq n$, so $C(P) \geq \Omega(nk)$, proving the theorem.

Extending the above argument to cover the case where the protocol P is allowed to return the wrong answer with probability ε is also easy, simply by showing that if Alice managed to learn Bob's entire bitvector with probability $1 - \varepsilon$, then $(1 - \varepsilon) \cdot n$ bits must have been communicated: this also follows easily from information-theoretic arguments.

Note the crucial fact that the hard distribution we chose is symmetric: If the distribution of the inputs to the k players was not symmetric, then the protocol P could try to deduce the indices i and j from some statistical properties of the observed inputs, and act according to that. Then the best upper bound we could get on the communication complexity of P' would be $C(P') \leq C(P)$, which is much too weak.

We have just described the version of the symmetrization method for the blackboard model. A similar (slightly more complicated) line of thinking leads to a lower bound of $\Omega(n \log k)$ on the complexity of the coordinatewise AND problem in the blackboard model; see section 3.1.

Let us now sketch the symmetrization method as it is used in the *coordinator model*, where players can only send and receive messages to and from the coordinator. We prove a lower bound on the same problem as above, the coordinatewise k-XOR problem.

Theorem 1.2. *The coordinatewise k-XOR problem requires communication $\Omega(nk)$ in the coordinator model.*

Note that this theorem actually follows from Theorem 1.1, since the blackboard model is stronger than the coordinator model. However, the proof we sketch here is useful as a warmup exercise, since it shows an easy example of the symmetrization technique as applied to the coordinator model. In the paper we prove multiple lower bounds using this technique, most of which do not follow from corresponding bounds in the blackboard model.

To prove this theorem, we use the same hard distribution as above: all inputs are uniform and independent. Let P be a protocol in the coordinator model, which computes the coordinatewise XOR, uses communication $C(P)$, and for now assume that P never makes any errors. As before, we build a new protocol P' for a 2-player problem. In the 2-player problem, suppose that Alice gets input x and Bob gets input y, where $x, y \in \{0, 1\}^n$ are independent random bitvectors. Then P' works as follows: Alice and Bob choose a single index $i \in \{1, \ldots, k\}$ from public randomness. Alice and

[1] Here and throughout the paper, *symmetric* means that the inputs are drawn from a distribution where renaming the players does not change the distribution. Namely, a distribution D over X^n is called *symmetric* if exchanging any two coordinates in D keeps the distribution the same.

Bob simulate the protocol P, where Alice simulates player i and lets $x_i = x$, and Bob plays *all the rest of the players*, including the coordinator, and chooses their inputs uniformly, conditioned on their XOR being equal to y.

To simulate the protocol P, whenever player i needs to send a message to the coordinator, then Alice sends a message to Bob, and whenever the coordinator needs to send a message to player i, Bob sends a message to Alice. Whenever any player $j \neq i$ needs to speak to the coordinator, no communication is needed, since both are played by Bob. Note again that the distribution of the inputs of the k players is independent uniform (for this it is crucial to remember that x and y were uniform and independent in the first place). Once again, from reasons of symmetry, since the index i was chosen uniformly and the inputs are symmetric, the expected communication performed by the protocol P' is $\mathbf{E}[C(P')] \leq 2C(P)/k$. Furthermore, at the end of the running of P', Alice knows the value of $x \oplus y$ so she can reconstruct the value of y. As before, this implies the theorem. The assumption that we never make errors can once again be easily removed.

1.1.1. Discussion. We see that the crux of the symmetrization technique in the coordinator model is to consider the k-player problem that we wish to lower-bound, to find a symmetric distribution which is hard for it, to give Alice the input of one player (chosen at random) and Bob the input of all other players, and to prove a lower bound for this two-player problem. If the lower bound for the two player problem is L, the lower bound for the k-player problem will be kL. For the blackboard model, the proofs have the same outline, except in the 2-player problem Alice gets the input of one randomly chosen player, Bob gets the input of another, and they both get the inputs of all the rest of the players. There is one important thing to note here: This argument only works when the hard distribution is symmetric.

1.2. Motivation, previous work, and related models. Communication complexity is a widely studied topic. In multiplayer number-in-hand communication complexity, the most studied mode of communication is the blackboard model. The message-passing model was already considered in [19]. (This model can also be called the *private-message model*, but note that this name was used in [22, 20] for a different model.) The coordinator model can be thought of as a server-site setting,[2] where there is one server and k sites. Each site has gathered n bits of information and the server wants to evaluate a function on the collection of these $k \cdot n$ bits. Each site can only communicate with the server, and a server can communicate with any site. This server-site model has been widely studied in the databases and distributed computing communities. Work includes computing top-k [9, 34, 37] and heavy hitters [45, 24].

Another closely related model is the *distributed monitoring model*, in which we also have one server and k sites. The only difference is that now the computation is dynamic. That is, each site receives a stream of elements over time and the server would like to maintain continuously at all times some function f of all the elements in the k sites. Thus the server-site model can be seen as a one-shot version of the distributed streaming setting. It follows that any communication complexity lower bound in the message-passing model or the coordinator model also hold in the distributed monitoring model. A lot of work on distributed monitoring has been done recently in the theory community and the database community, including maintaining random samplings [16], frequency moments [13, 15], heavy hitters [5, 29, 32, 43, 26],

[2] This terminology is similar as the standard "client-server," and is used extensively in the literature.

quantiles [14, 43], entropy [4], and various sketches [17, 14].

We will come back to the latter two models in section 6. It is interesting to note that despite the large number of upper bounds (i.e., algorithms, communication protocols) in the above models, very few lower bounds have been proved in any of those models, likely because there were few known techniques to prove such results.

A further application of the message-passing model could be for Secure Multiparty Computation: in this model, there are several players who do not trust each other, but want to compute a joint function of their inputs, with each of them learning nothing about the inputs of the others players except what can be learned from the value of the joint function. Obviously, any lower bound in the message-passing model immediately implies a lower bound on the amount of communication required for Secure Multiparty Computation. For more on this model, see, e.g., [21].

One final application is for the streaming model [3]. In this model, there is a long stream of data that can only be scanned from left to right. The goal is to compute some function of the stream, and minimize the space usage. It is easy to see that if we partition the stream into k parts and give each part to a different player, then a lower bound of L on the communication complexity of the problem in the coordinator model implies a lower bound of L/k on the space usage. When t passes over the model are allowed, a lower bound of L in the coordinator model translates to a lower bound of L/tk in the streaming model. In this t-pass variant, direct sum results exist [23], and symmetrization provides another avenue for proving lower bounds.

1.3. Our results and paper outline. Our main technical result in this paper are lower bounds of $\Omega(nk)$ randomized communication for the bitwise k-party AND, OR, and MAJ (majority) functions in the coordinator model. These sidestep clever upper bound techniques (e.g., Slepian–Wolf coding) and can be found in section 3. In the same section we prove some lower bounds for AND and OR in the blackboard model as well. Back to the coordinator model, we show that the connectivity problem (given k players with subgraphs on a common set of nodes, determine if it is connected) requires $\Omega(nk/\log^2 k)$ communication. This is in section 4.

The coordinatewise lower bounds imply lower bounds for the well-studied problems of distinct elements, ε-approximate heavy hitters, and ε-kernels in the server-site model (or the other related models). We show any randomized algorithm requires at least $\Omega(nk)$, $\Omega(n/\varepsilon)$, and $\Omega(k/\varepsilon^{(d-1)/2})$ communication, respectively. The latter is shown to be tight. This is in section 6.

We give some direct-sum-like results in section 5.

1.4. Subsequent work. A series of work has been done after the conference version of this paper. Woodruff and Zhang [39, 41] combined the symmetrization technique and a new technique called *composition* to show strong lower bounds for approximately computing a number of statistical problems, including distinct elements and frequency moments, in the coordinator model. The same authors also used symmetrization to prove tight lower bounds for the exact computation of a number of graph and statistical problems [40], and shaved a $\log k$ factor for the connectivity problem studied in this paper. The other $\log k$ factor in the connectivity lower bound can be further shaved using a relaxation of symmetrization proposed in [41]. As will be discussed in section 7.2, due to a limitation of the symmetrization technique, it cannot be used to prove a tight lower bound for the k-player disjointness problem. This was listed as an open problem in the conference version of this paper, and was later settled by Braverman et al. [8], using a different technique based on information complexity. Recently Huang et al. [25] applied symmetrization together with information complexity to prove tight lower bounds for approximate maximum matchings; Li et al. [31] used the symmetrization technique to prove lower bounds for numerical linear algebra problems. In another recent work, Chattopadhyay, Radhakrishnan, and Rudra [12] further extended the symmetrization technique to general communication topology compared with the coordinator model which essentially has a star communication topology.

2. Preliminaries.

In this section we review some basic concepts and definitions. We denote $[n] = \{1, \ldots, n\}$. All logarithms are base-2 unless noted otherwise.

Communication complexity. Consider two players Alice and Bob, given bitvectors A and B, respectively. Communication complexity (see, for example, the book [30]) bounds the communication between Alice and Bob that is needed to compute some function $f(A, B)$. The *communication complexity* of a particular protocol is the maximal number of bits that are communicated, taken in the worst case over all pairs of inputs. The *communication complexity* of the problem f is the best communication complexity of P, taken over all protocols P that correctly compute f.

Certain functions (such as $f = \mathsf{EQ}$ which determines if A equals B) can be computed with less communication if randomness is permitted. Let $R^{\varepsilon}(f)$ denote the communication complexity when the protocol is allowed to make a mistake with probability ε. The error is taken over the randomness used by the protocol.

Sometimes we are interested in the case where the input of Alice and Bob is drawn from some distribution μ over pairs of inputs. We want to allow an error ε, this time taken over the distribution of the input. The worst-case communication complexity in this case is denoted by $D^{\varepsilon}_{\mu}(f)$. Yao [42] showed that $R^{\varepsilon}(f) = \max_{\mu} D^{\varepsilon}_{\mu}(f)$. Thus in order to prove a lower bound for randomized protocols, it suffices to find a hard distribution and prove a distributional lower bound for it. This is called the Yao minimax principle.

In this paper we use an uncommon notion of *expected distributional communication complexity*. In this case we consider the distributional setting as in the last paragraph, but this time consider the expected cost of the protocol, rather than the worst-case cost; again, the expectation is taken over the distribution of input. We denote this $\mathrm{ED}^{\varepsilon}_{\mu}(f)$.

2.1. Two-party lower bounds.

We state a couple of simple two-party lower bounds that will be useful in our reductions.

2-*BITS*. Let ζ_{ρ} be a distribution over bitvectors of length n, where each bit is 1 with probability ρ and 0 with probability $1 - \rho$. In this problem Alice gets a vector drawn from ζ_{ρ}, Bob gets a subset S of $[n]$ of cardinality $|S| = \alpha n$ for constant $\alpha \in (0, 1)$, and Bob wishes to learn the bits of Alice indexed by S.

The proof of the following lemma is in Appendix A.

Lemma 2.1. $\mathrm{ED}^{1/3}_{\zeta_{\rho}}(2\text{-}\mathit{BITS}) = \Omega(n\rho \log(1/\rho))$.

2-*DISJ*. In this problem Alice and Bob each have an n-bit vector. If we view vectors as sets, then each of them has a subset of $[n]$ corresponding to the 1 bits. Let x be the set of Alice and y be the set of Bob. It is promised that $|x \cap y| = 1$ or 0. The goal is to return 1 if $x \cap y \neq \emptyset$, and 0 otherwise.

We define the input distribution μ as follows. Let $l = (n+1)/4$. With probability $1/t$, x and y are random subsets of $[n]$ such that $|x| = |y| = l$ and $|x \cap y| = 1$. And with probability $1 - 1/t$, x and y are random subsets of $[n]$ such that $|x| = |y| = l$ and $x \cap y = \emptyset$. Razborov [38] (see also [27]) proved that for $t = 4$, $D^{1/100t}_{\mu}(2\text{-}\mathsf{DISJ}) = \Omega(n)$.

In the following theorem we extend this result to general t and also to the expected communication complexity. In section 3.1 we need only $t = 4$, and in section 4 we will need general t.

The proof for the following lemma is in Appendix B.

LEMMA 2.2. *When μ has $|x \cap y| = 1$ with probability $1/t$ then* $\mathrm{ED}_{\mu}^{1/100t}(2\text{-}\mathsf{DISJ}) = \Omega(n)$.

3. Bitwise problems.

3.1. Multiparty AND/OR.

We now consider multiparty AND/OR (below we use k-AND and k-OR for short). In the k-AND problem, each player i ($1 \leq i \leq k$) has an n-bit vector I_i and we want to establish the bitwise AND of I_i, that is, $f_j(I_1, \ldots, I_k) = \bigwedge_i I_{i,j}$ for $j = \{1, \ldots, n\}$. k-OR is similarly defined with OR. Observe that the two problems are isomorphic by $f_j(I_1, \ldots, I_k) = \neg g_j(\bar{I}_1, \ldots, \bar{I}_k)$ for $j = \{1, \ldots, n\}$, where $\bar{I}_i$ is obtained by flipping all bits of I_i and $g_j(\bar{I}_1, \ldots, \bar{I}_k) = \bigvee_i \bar{I}_{i,j}$. Therefore we need only consider one of them. Here we discuss k-OR.

3.1.1. Idea for the k-OR lower bound in the coordinator model.

We now discuss the hard distribution and sketch how to apply the symmetrization technique for the k-OR problem in the coordinator model. The formal proof can be found in the next subsection.

We in fact start by describing two candidate hard distributions that *do not* work. The reasons they do not work are interesting in themselves. Throughout this subsection, assume for simplicity that $k \geq 100 \log n$; note that this assumption is not required for our main result, but is used here to illustrate the main idea.

The most natural candidate for a hard distribution is to make each entry equal to 1 with probability $1/k$. This has the effect of having each bit in the output vector be roughly balanced, which seems suitable for being a hard case. This is indeed the hard distribution for the blackboard model, but for the coordinator model (or the message-passing model) it is an easy distribution: Each player can send his entire input to the coordinator, and the coordinator can figure out the answer. The entropy of each player's input is only $\Theta((n \log k)/k)$, so the total communication would be $\Theta(n \log k)$ in expectation using e.g., Shannon's coding theorem;[3] this is much smaller than the lower bound we wish to prove. Clearly, we must choose a distribution where each player's input has entropy $\Omega(n)$. This is the first indication that the k-player problem is significantly different than the 2-player problem, where the above distribution is indeed the hard distribution.

The next candidate hard distribution is to randomly partition the n coordinates into two equal-sized sets: The *important set*, where each entry is equal to 1 with probability $1/k$, and the *balancing set*, where all entries are equal to 1. Now the entropy of each player's input is $\Theta(n)$, and the distribution seems like a good candidate, but there is a surprising upper bound for this distribution: the coordinator asks $100 \log n$ players to send him their entire input, and from this can easily figure out which coordinates are in the balancing set and which are in the important set. Henceforth, the coordinator knows this information, and need only learn the players' values in the important set, which again have low entropy. We would want the players to send these values, but the players themselves do not know which coordinates

[3] To show the upper bounds in this subsection we use some notions from information theory without giving complete background for them. The reader can refer to, e.g., [18], or alternatively can skip them entirely, as they are inconsequential for the remainder of the paper and for understanding the symmetrization technique.

are in the important set, and the coordinator would need to send nk bits to tell all of them this information. However, they do not need to know this in order to get all the information across: using a protocol known as *Slepian–Wolf coding* (see, e.g., [18]) the players can transmit to the coordinator all of their values in the important coordinates, with only $n \log k$ total communication (and a small probability of error). The idea is roughly as follows: each player p_i chooses $100n \log k/k$ sets $S_{i,j} \subseteq [n]$ independently and uniformly at random from public randomness. For each j, the player XORs the bits of his input in the coordinates of $S_{i,j}$, and sends the value of this XOR to the coordinator. The coordinator already knows the balancing set, so he only has $\Theta(n \log k/k)$ bits of uncertainty about player i's input, meaning he can reconstruct the player's input with high probability (say by exhaustive search). The upper bound follows.

To get an actual hard distribution, we modify the hard distribution from the last paragraph. We randomly partition the n coordinates into two equal-sized sets: The *important set*, where each entry is equal to 1 with probability $1/n$, and the *noise set*, where each entry is equal to 1 with probability $1/2$.[4] Clearly, each player's input has entropy $\Theta(n)$. Furthermore, the coordinator can again cheaply figure out which coordinates are in the important set and which are in the noise set, but the players do not know this information, and nothing like Slepian–Wolf coding exists to help them transmit the information to the coordinator. The distribution that we use in our formal proof is a little different than this, for technical reasons, but this distribution is hard as well.

We now sketch how to apply the symmetrization technique to prove that this distribution is indeed hard. To apply the symmetrization technique, we imagine giving Alice the input of one of the players, and to Bob the input of all of the others. Bob plays the coordinator, so Bob needs to compute the output; the goal is to prove a lower bound of $\Omega(n)$ on the communication complexity between Alice and Bob. What can Bob deduce about the answer? He can immediately take the OR of all the vectors that he receives, getting a vector where with good probability all of the noise bits are equal to 1. (Recall that we assumed that the number of players is $k \geq 100 \log n$.) Among the important bits, roughly one of Alice's bits is equal to 1, and the goal is to discover which bit it is. Alice cannot know which coordinates are important, and in essence we are trying to solve a problem very similar to set disjointness. A lower bound of $\Omega(n)$ is easy to get convinced of, since it is similar to the known lower bounds for set disjointness. Proving it is not entirely trivial, and requires making some small modifications to Razborov's classical lower bound on set disjointness [38].

We see that the lower bound for this distribution has to (implicitly) rule out a Slepian–Wolf type upper bound. This provides some evidence that any lower bound for the k-OR problem would have to be nontrivial.

3.1.2. The proof. We prove the lower bound on k-OR by performing a reduction from the promise version of the two-party *set disjointness* problem (2-DISJ) which we lower-bounded in Lemma 2.2. Given an (x, y) for 2-DISJ drawn from the hard distribution μ, we construct an input for k-OR. Note that our mapping is not necessarily one-to-one, that is why we need a lower bound on the *expected* distributional communication complexity of 2-DISJ.

[4] We could have chosen each entry in the important set to be equal to 1 with probability $1/k$ as well, but choosing a value of $1/n$ makes the proofs easier. The important thing is to choose each of the noise bits to be equal to 1 with probability $1/2$.

Reduction. We start with Alice's input set x and Bob's input set y from the distribution μ, with $t = 4$. That is, both x and y have $l = n/4 \ 1$ bits chosen at random under the condition that they intersect at one point with probability $1/4$, otherwise they intersect at no point. The set y will play the role of the *important set* from the intuitive argument above. For the 2-DISJ problem, we construct k players' input sets $I_1, \ldots, I_k$ as follows. Let $z = [n] - y$. Let $S_2^l, \ldots, S_k^l$ be random subsets of size l from z, and let $S_2^{l-1}, \ldots, S_k^{l-1}$ be random subsets of size $l-1$ from z. Let $T_2, \ldots, T_k$ be random elements from y. We have the following:

$$\begin{cases} I_1 = x, \\ I_j \ (j = 2, \ldots, k) = \begin{cases} S_j^l & \text{w.p. } 1 - 1/4, \\ S_j^{l-1} \cup T_j & \text{w.p. } 1/4. \end{cases} \end{cases}$$

Let μ' be this input distribution for k-OR. If I_j $(2 \le j \le k)$ contains an element T_j, then we call this element a special element.

This reduction can be interpreted as follows: Alice simulates a random player I_1, and Bob, playing as the coordinator, simulates all the other $k-1$ players $I_2, \ldots, I_k$. Bob also keeps a set V containing all the special elements that ever appear in some I_j $(j = 2, \ldots, k)$. It is easy to observe the following fact.

LEMMA 3.1. *All I_j $(j = 1, \ldots, k)$ are chosen from the same distribution.*

Proof. Since by definition I_j $(j = 2, \ldots, k)$ are chosen from the same distribution, we need only show that $I_1 = x$ under μ is chosen from the same distribution as any I_j is under μ'. Given y, note that x is a random set of size l in $z = [n] - y$ with probability $1/4$; and with the remaining probability x is the union of a random set of size $l-1$ in z along with a single random element from y. This is precisely the distribution of each I_j for $j \ge 2$. □

This lemma actually implies that Alice can simulate an arbitrary player instead of a random player. The following lemma shows the properties of our reduction.

LEMMA 3.2. *If there exists a protocol $\mathcal{P}'$ for k-OR on input distribution μ' with communication complexity C and error bound ε, then there exists a protocol $\mathcal{P}$ for the 2-DISJ on input distribution μ with expected communication complexity $O(C/k)$ and error bound $\varepsilon + 4k/n$.*

Proof. Let us again view I_j $(j = 1, \ldots, k)$ as n-bit vectors. We show how to construct a protocol $\mathcal{P}$ for 2-DISJ from a protocol $\mathcal{P}'$ for k-OR with the desired communication cost and error bound. $\mathcal{P}$ is constructed as follows: Alice and Bob first run $\mathcal{P}'$ on $I_1, \ldots, I_k$. Let $W \subseteq [n]$ be the set of indices where the results are 1. Bob checks whether there exists some $w \in W \cap y$ such that $w \notin V$. If yes, then $\mathcal{P}$ returns "yes," otherwise $\mathcal{P}$ returns "no."

We start by analyzing the communication cost of $\mathcal{P}$. Since player I_1 is chosen randomly from the k players, and from Lemma 3.1 that all players' inputs are chosen from a same distribution, the expected amount of communication between I_1 (simulated by Alice) and the other $k-1$ players (simulated by Bob) is at most a $2/k$ fraction of the total communication cost of $\mathcal{P}$. Therefore, the expected communication cost of $\mathcal{P}$ is at most $O(C/k)$.

For the error bound, we have the following claim: With probability at least $(1 - 4k/n)$, there exists a $w \in W \cap y$ such that $w \notin V$ if and only if $x \cap y \notin \emptyset$. First, if $x \cap y = \emptyset$, then $I_1 = x$ cannot contain any element $w \in V \subseteq y$, thus the resulting bits in $W \cap y$ cannot contain any special element that is not in V. On the other hand, we have $\mathbf{Pr}[((W \cap y) \subseteq V) \wedge (x \cap y \neq \emptyset)] \le 4k/n$. This is because $((W \cap y) \subseteq V)$ and $(x \cap y \neq \emptyset)$ hold simultaneously if and only if there exist some T_j $(1 \le j \le k)$ such that $T_j \in x \cap y$. According to our random choices of T_j ($j = 1, \dots, k$), this holds with probability at most $k/l \leq 4k/n$. Therefore, if $\mathcal{P}'$ is incorrect with probability at most ε, then $\mathcal{P}$ is incorrect with probability at most $\varepsilon + 4k/n$. □

Combining Lemmas 2.2 and 3.2, we have the following theorem.

THEOREM 3.3. *$D_{\mu'}^{1/800}(k\text{-}\mathsf{OR}) = \Omega(nk)$, for $n \geq 3200k$ in the coordinator model.*

Proof. If there exists a protocol $\mathcal{P}'$ that computes k-OR on input distribution μ' with communication complexity $o(nk)$ and error bound 1/800, then by Lemma 3.2 there exists a protocol $\mathcal{P}$ that computes 2-DISJ on input distribution μ with expected communication complexity $o(n)$ and error bound $1/800 + 4k/n \leq 1/400$, contradicting Lemma 2.2 (when $t = 4$). □

We discuss the applications of this lower bound to the distinct elements problem in section 6.

3.2. Multiparty AND/OR with a blackboard. Denote the k-OR problem in the blackboard model by k-OR-board. The general idea to prove a lower bound for k-OR-board is to perform a reduction from a 2-party bitwise OR problem (2-OR for short) with public randomness. The 2-OR problem is the following. Alice and Bob each have an n-bit vector drawn from the following distribution: Each bit is 1 with probability $1/k$ and 0 with probability $1 - 1/k$. They also use public random bits to generate another $(k-2)$ n-bit vectors such that each bit of these vectors is 1 with probability $1/k$ and 0 with probability $1 - 1/k$. That is, the bits are drawn from the same distribution as their private inputs. Let ν be this input distribution. Alice and Bob want to compute bitwise OR of all these k n-bit vectors. For this problem we have the following theorem.

THEOREM 3.4. *$\mathrm{ED}_{\nu}^{1/3}(2\text{-}\mathsf{OR}) = \Omega(n/k \cdot \log k)$.*

Proof. Without loss of generality, let us assume that Bob outputs the final result of 2-OR. It is easy to see that if we take the bitwise OR of the $(k-2)$ n-bit vectors generated by public random bits and Bob's input vector, the resulting n-bit vector b will have at least a constant density of 0 bits with probability at least $1 - o(1)$, by a Chernoff bound. Since Bob can see the $k-2$ public vectors, to compute the final result, all that Bob has to know are bits of Alice's vector on those indices i ($1 \leq i \leq n$) where $b[i] = 0$. In other words, with probability at least $1 - o(1)$, Bob must learn a specific set of Alice's bit vector up to error 1/3, and these represent at least a constant fraction of all of Alice's bits. Plugging in Lemma 2.1 with $\rho = 1/k$, we know that the expected communication complexity is at least $(1 - o(1)) \cdot \Omega(n/k \cdot \log k) = \Omega(n/k \cdot \log k)$. □

We reduce this problem to k-OR-board as follows: Alice and Bob each simulate a random player, and they use public random bits to simulate the remaining $k-2$ players: The $(k-2)$ n-bit vectors generated by their public random bits are used as inputs for the remaining $k-2$ random players. Observe that the input of all the k players are drawn from the same distribution, thus the k players are symmetric. Consequently, the expected amount of communication between the two random players simulated by Alice and Bob and other players (including the communication between the two random players) is at most $O(1/k)$ faction of the total communication cost of protocol for k-OR-board. We have the following theorem.

THEOREM 3.5. *$D_{\nu}^{1/3}(k\text{-}\mathsf{OR\text{-}board}) = \Omega(n \cdot \log k)$.*

Proof. It is easy to see that if we have a protocol for the k-OR-board on input distribution ν with communication complexity $o(n \cdot \log k)$ and error bound 1/3, then we have a protocol for 2-OR on input distribution ν with expected communication complexity $o(n/k \cdot \log k)$ and error bound 1/3, contradicting Theorem 3.4. □

It is easy to show a tight deterministic upper bound of $O(n \log k)$ for this problem in the blackboard model: each player speaks in turn and writes the coordinates where he has 1 and no player that spoke before him had 1.

3.3. Majority. In the k-MAJ problem, we have k players, each having a bit vector of length n, and they want to compute bitwise majority, i.e., to determine for each coordinate whether the majority of entries in this coordinate are 1 or 0. We prove a lower bound of $\Omega(nk)$ for this problem in the coordinator model by a reduction to 2-BITS via symmetrization.

For the reduction, we consider $k = 2t + 1$ players and describe the input distribution τ as follows. For each coordinate we assign it either t or $(t + 1)$, each with probability $1/2$, independently over all coordinates. This indicates whether the k players contain t or $(t + 1)$ 1 bits among them in this coordinate, and hence whether that index has a majority of 0 or 1, respectively. Then we place either t or $(t + 1)$ 1 bits randomly among the k players inputs in this coordinate. It follows that under τ: (i) each player's bits are drawn from the same distribution, (ii) each bit of each player is 1 with probability $1/2$ and 0 with probability $1/2$, and (iii) each index has probability $1/2$ of having a majority of 1s.

THEOREM 3.6. *$D_{\tau}^{1/6}(k\text{-}MAJ) = \Omega(nk)$ in the coordinator model.*

Proof. Now we use symmetry to reduce to the two-player problem 2-BITS. Alice will simulate a random player under τ and Bob will simulate the other $k - 1$ players. Notice that by (ii) any subset of n' indices of Alice's are from $\zeta_{1/2}$. We will show that Alice and Bob need to solve 2-BITS on $\Omega(n)$ bits to solve k-MAJ. And, by (i), since all players have the same distribution and Alice is a random player, her expected cost in communicating to Bob is at most $O(C/k)$ if k-MAJ can be solved in C communication.

Now consider the aggregate number of 1 bits Bob has for each index; he has $(t - 1)$ 1 bits with probability $1/4$, t 1 bits with probability $1/2$, and $(t + 1)$ 1 bits with probability $1/4$. Thus for at most $(3/4)n$ indices (with probability at least $1 - \exp(-2(n/4)^2/n) = 1 - \exp(-n/8) \geq 1 - 1/7$) that have either $(t - 1)$ or $(t + 1)$ 1 bits Bob knows that these either will or will not have a majority of 1 bits, respectively. But for the other at least $n/4 = \Omega(n)$ remaining indices for which Bob has exactly t 1 bits, whether or not these indices have a majority of 1 bits depends on Alice's bit. And conditioned on this situation, each of Alice's relevant bits are 1 with probability $1/2$ and 0 with probability $1/2$, hence distributed by $\zeta_{1/2}$. Thus conditioned on at least $n/4$ undecided indices, this is precisely the 2-BITS problem between Alice and Bob of size $n/4$.

Thus a protocol for k-MAJ in $o(nk)$ communication and error bound $1/6$ would yield a protocol for 2-BITS in expected $o(n)$ communication and error bound $1/6 + 1/7 < 1/3$, by running the protocol simulated between Alice and Bob. This contradicts Lemma 2.1, and proves that k-MAJ requires $\Omega(kn)$ communication when allowing error on at most $1/6$ fraction of inputs. □

Extensions. This lower bound can easily be extended beyond just majority (threshold $1/2$) to any constant threshold $\phi(0 < \phi < 1)$, by assigning to each coordinate either $\lfloor k\phi \rfloor$ or $(\lfloor k\phi \rfloor + 1)$ 1 bits with probability $1/2$ each. Let τ_ϕ denote this distribution. Then the analysis just uses ζ_ϕ in place of $\zeta_{1/2}$, which also yields an $\Omega(n)$ lower bound for 2-BITS. We call this extended k-MAJ problem (k, ϕ)-MAJ.

COROLLARY 3.7. *$D_{\tau_\phi}^{1/6}((k, \phi)\text{-}MAJ) = \Omega(nk)$ for any constant $\phi(0 < \phi < 1)$ in the coordinator model.*

We discuss the applications of this lower bound to the heavy-hitter problem in section 6.

4. Graph connectivity. In the k-CONN problem, we have k players, each having a set of edges in an n-vertex graph. The goal is to decide whether the graph consisting the union of all of these edges is connected. In this section we prove an $\Omega(nk/\log^2 k)$ lower bound for k-CONN in the coordinator model by performing a symmetry-based reduction from 2-DISJ.

4.1. Proof idea and the hard distribution. Let us start by discussing the hard distribution. Assume for simplicity $k \geq 100 \log n$. We describe a hard distribution which is not quite the same as the one in the proof (due to technical reasons), but is conceptually clearer. In this hard distribution, we consider a graph G, which consists of two disjoint cliques of size $n/2$. We also consider one edge between these two cliques, called the *connector*; the connector is not part of G. Each player gets as input $n/10$ edges randomly and uniformly chosen from the graph G; furthermore, with probability $1/2$ we choose exactly one random edge in one random player's input, and replace it by the connector. It is easy to see that if one of the players gets the connector, then with high probability the resulting set of edges span a connected graph; otherwise the graph is not connected.

To get convinced that the lower bound holds, notice that the coordinator can easily reconstruct the graph G. However, in order to find out if one of the players has received the connector, the coordinator needs to speak with each of the players to find this out. The situation is roughly analogous to the situation in the k-OR problem, since the players themselves did not get enough information to know G, and no Slepian–Wolf type protocol is possible since the edges received by each player are random-looking. The actual distribution that we use is somewhat more structured than this, in order to allow an easier reduction to the 2-player disjointness problem.

4.2. The proof. We first recall 2-DISJ. Similarly to before, in 2-DISJ, Alice and Bob have inputs x ($|x| = \ell$) and y ($|y| = \ell$) chosen uniformly at random from $[n]$ ($n = 4\ell - 1$), with the promise that with probability $1/10k$, $|x \cap y| = 1$ and with probability $1 - 1/10k$, $|x \cap y| = 0$. Let φ be this input distribution for 2-DISJ. Now given an input (x, y) for 2-DISJ, we construct an input for k-CONN.

Let $K_{2n} = (V, E)$ be the complete graph with $2n$ vertices. Given Alice's input x and Bob's input y, we construct k players' input $I_1, \ldots, I_k$ such that $|I_j| = \ell$ and $I_j \subseteq E$ for all $1 \leq j \leq k$. We first pick a random permutation σ of $[2n]$. Alice constructs $I_1 = \{(\sigma(2i-1), \sigma(2i)) \mid i \in x\}$.

Bob constructs $I_2, \ldots, I_k$. It is convenient to use σ and y to divide V into two subsets L and R. For each i ($1 \leq i \leq n$), if $i \in y$, then with probability $1/2$, we add $\sigma(2i-1)$ to L and $\sigma(2i)$ to R; and with the rest of the probability, we add $\sigma(2i-1)$ to R and $\sigma(2i)$ to L. Otherwise if $i \notin y$, then with probability $1/2$, we add both $\sigma(2i-1)$ and $\sigma(2i)$ to L; and with the rest of the probability, we add both $\sigma(2i-1)$ and $\sigma(2i)$ to R. Let $K_L = (L, E_L)$ and $K_R = (R, E_R)$ be the two complete graphs on sets of vertices L and R, respectively. Now using E_R and E_L, Bob can construct each I_j. With probability $1 - 1/10k$, I_j is a random subset of disjoint edges (i.e., a matching) from $E_L \cup E_R$ of size ℓ; and with probability $1/10k$, I_j is a random subset of disjoint edges from $E_L \cup E_R$ of size $\ell - 1$ and one random edge from $E \setminus (E_L \cup E_R)$.

Let φ' be the input distribution for k-CONN defined as above for each I_j ($1 \leq j \leq k$). We define the following two events.

ξ_1: Both edge-induced subgraphs $\cup_{j=2}^k I_j \bigcap E_L$ and $\cup_{j=2}^k I_j \bigcap E_R$ are connected, and span L and R, respectively.

ξ_2: $\cup_{j=2}^k I_j \bigcap E$ is *not* connected.

It is easy to observe the following two facts by our construction.

LEMMA 4.1. *All I_j $(j = 1, \dots, k)$ are chosen from the same distribution.*

Proof. Since σ is a random permutation of $[2n]$, according to the distribution φ', Alice's input can also be seen as follows: With probability $1 - 1/10k$, it is a random matching of size ℓ from $E_L \cup E_R$; and with probability $1/10k$, it is a matching consisting of $\ell - 1$ random edges from $E_L \cup E_R$ and one random edge from $E \backslash (E_L \cup E_R)$, which is $(\sigma(2z-1), \sigma(2z))$ where $z = x \cap y$. ◻

LEMMA 4.2. *ξ_1 happens with probability at least $1 - 1/2n$ when $k \geq 68 \ln n + 1$.*

Proof. (This is a proof sketch; full proof in Appendix C.) First, note that by our construction, both $|L|$ and $|R|$ are $\Omega(n)$ with high probability. To locally simplify notation, we consider a graph (V, E) of n nodes where edges are drawn in $(k-1) \geq 68 \ln n$ rounds, and each round $n/4$ disjoint edges are added to the graph. If (V, E) is connected with probability at least $(1 - 1/4n)$, then by union bound over $\cup_{j=2}^{k} I_j \bigcap E_L$ and $\cup_{j=2}^{k} I_j \bigcap E_R$, ξ_1 is true with probability at least $(1 - 1/2n)$. The proof follows four steps.

(S1) Using the first $28 \ln n$ rounds, we can show that all vertices have degree at least $8 \ln n$ with probability at least $1 - 1/12n$.

(S2) Conditioned on (S1), any subset $S \subset V$ of $h < n/10$ vertices is connected to at least $\min\{h \ln n, n/10\}$ distinct vertices in $V \setminus S$, with probability at least $1 - 1/n^2$.

(S3) Iterate (S2) $\ln n$ times to show that there must be a single connected component S_G of size at least $n/10$, with probability at least $1 - 1/12n$.

(S4) Conditioned on (S3), using the last $40 \ln n$ rounds we can show that all vertices are connected to S_G with probability at least $1 - 1/12n$. ◻

The following lemma shows the properties of our reduction.

LEMMA 4.3. *Assume $k \geq 100 \log n$. If there exists a protocol $\mathcal{P}'$ for k-CONN on input distribution φ' with communication complexity C and error bound ε, then there exists a protocol $\mathcal{P}$ for the 2-DISJ on input distribution φ with expected communication complexity $O(C/k \cdot \log k)$ and error bound $(29 \ln k \cdot \varepsilon + 1/2000k)$.*

Proof. In $\mathcal{P}$, Alice and Bob first construct $\{I_1, \dots, I_k\}$ according to our reduction, and then run the protocol $\mathcal{P}'$ on it. By Lemma 4.2 we have that ξ_1 holds with probability at least $1 - 1/2n$. And by our construction, conditioned that ξ_1 holds, ξ_2 holds with probability at least $(1 - 1/10k)^{k-1} \geq 1 - 1/10$. Thus the input generated by a random reduction encodes the 2-DISJ problem with probability at least $(1 - 1/2n - 1/10) > 8/9$. We call such an input a *good* input. We repeat the random reduction $c \ln k$ times (for some large enough constant c; e.g., $c \geq 29$) and run $\mathcal{P}'$ on each of the resulting inputs for k-CONN. The probability that at least 2/3-fraction of inputs are good is at least $1 - 1/2000k$ (by picking c large enough; a Chernoff bound shows this probability is at most $2\exp(-2(2/9)^2(c \ln k))$ which is less than $1/(2000k)$ with $c \geq 29$ and $k \geq 100$). Thus we can output the majority of the $c \ln k$ runs of $\mathcal{P}'$, and obtain a protocol $\mathcal{P}$ for 2-DISJ with expected communication complexity $O(C/k \cdot \log k)$ and error bound $\varepsilon \cdot 29 \ln k + 1/2000k$; the $\varepsilon \cdot 29 \ln k$ term comes from a union bound over the ε error on each of $29 \ln k$ runs of k-CONN. ◻

Combining Lemmas 2.2 and 4.3, we have the following theorem.

THEOREM 4.4. *$D_{\varphi'}^{1/(2000 \cdot (29 \ln k) \cdot k)}(k\text{-CONN}) = \Omega(nk/\log k)$, for $k \geq 100 \log n$ in the coordinator model.*

Proof. If there exists a protocol $\mathcal{P}'$ that computes k-CONN with communication complexity $o(nk/\log k)$ and error $1/(2000 \cdot (29 \ln k) \cdot k)$, then by Lemma 3.2 there exists a protocol $\mathcal{P}$ that computes 2-DISJ with expected communication complexity $o(n)$ and error at most $(1/(2000 \cdot (29 \ln k) \cdot k) \cdot (29 \ln k)) + 1/2000k = 1/1000k$, contradicting Lemma 2.2 (when $t = 10k$). □

Finally, we have the following immediate consequence:

$$\begin{aligned} R^{1/3}(k\text{-}\mathsf{CONN}) &\geq \Omega(R^{1/(2000\cdot(29\ln k)\cdot k)}(k\text{-}\mathsf{CONN})/\log k) \\ &\geq \Omega(D_{\varphi'}^{1/(2000\cdot(29\ln k)\cdot k)}(k\text{-}\mathsf{CONN})/\log k) \\ &\geq \Omega(nk/\log^2 k). \end{aligned}$$

5. Some direct-sum-like results. Let $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ be an arbitrary function. Let μ be a probability distribution over $\mathcal{X} \times \mathcal{Y}$. Consider a setting where we have $k+1$ players: Carol and $P_1, P_2, \ldots, P_k$. Carol receives an input from $x \in \mathcal{X}$ and each P_i receives an input $y_i \in \mathcal{Y}$. Let $R^\varepsilon(f^k)$ denote the randomized communication complexity of computing f on Carol's input and each of the k other players' inputs, respectively; i.e., computing $f(x, y_1), f(x, y_2), \ldots, f(x, y_k)$. Our direct-sum-like theorem in the message-passing model states the following theorem.

THEOREM 5.1. *In the message-passing model, for any function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ and any distribution μ on $\mathcal{X} \times \mathcal{Y}$, we have $R^\varepsilon(f^k) \geq \Omega(k \cdot \mathrm{ED}_\mu^\varepsilon(f))$.*

Note that this is not a direct-sum theorem in the strictest sense, since it relates randomized communication complexity to expected distributional complexity. However, it should probably be good enough for most applications. The proof of this theorem is dramatically simpler than most direct-sum proofs known in the literature (e.g., [7]). This is not entirely surprising, as it is weaker than those theorems: it deals with the case where the inputs are spread out over many players, while in the classical direct-sum setting the inputs are only spread out over two players (this would be analogous to allowing the players P_i to communicate with each other for free, and only charging them for speaking to Carol). However, perhaps more surprisingly, *optimal* direct-sum results are *not known* for most models, and are considered to be central open questions, while the result above is essentially optimal. Optimal direct-sum results in 2-player models would have dramatic consequences in complexity theory (see, e.g., [28] as well as [36]), so it seems interesting to check whether direct-sum results in multiparty communication could suffice for achieving those complexity-theoretic implications.

Proof. Given the distribution μ on $\mathcal{X} \times \mathcal{Y}$, we construct a distribution ν on $\mathcal{X} \times \mathcal{Y}^k$. Let ρ_x be the marginal distribution on $\mathcal{Y}$ induced by μ condition on $X = x$. We first pick $(x, y_1) \in \mathcal{X} \times \mathcal{Y}$ according to μ, and then pick $y_2, \ldots, y_k$ independently from $\mathcal{Y}$ according to ρ_x . We show that $D_\nu^\varepsilon(f^k) \geq \Omega(k \cdot \mathrm{ED}_\mu^\varepsilon(f))$. The theorem follows by Yao's min-max principle.

Suppose that Alice and Bob get inputs (u, w) from $\mathcal{X} \times \mathcal{Y}$ according to μ. We can use a protocol for f^k to compute $f(u, w)$ as follows: Bob simulates a random player in $\{P_1, \ldots, P_k\}$. Without loss of generality, say it is P_1. Alice simulates Carol and the remaining $k-1$ players. The inputs for Carol and $P_1, \ldots, P_k$ are constructed as follows: $x = u$, $y_1 = w$, and $y_2, \ldots, y_k$ are picked from $\mathcal{Y}$ according to ρ_u (Alice knows u and μ so she can compute ρ_u). Let ν be the distribution of $(x, y_1, \ldots, y_k)$ in this construction. We now run the protocol for f^k on $x, y_1, \ldots, y_k$. The result also gives $f(u, w)$.

Since $y_1, \ldots, y_k$ are chosen from the same distribution and P_1 is picked uniformly at random from the k players other than Carol, we have that in expectation, the expected amount of communication between P_1 and $\{\text{Carol}, P_2, \ldots, P_k\}$, or equivalently, the communication between Alice and Bob according to our construction, is at most a $2/k$ fraction of the total communication of the $(k+1)$-player game. Thus $D^{\varepsilon}_{\nu}(f^k) \geq \Omega(k \cdot \mathrm{ED}^{\varepsilon}_{\mu}(f))$. $\square$

5.1. With combining functions. In this section we extend Theorem 5.1 to the complexity of the AND/OR of k copies of 0/1 function f. Similar to before, AND and OR are essentially the same so we only talk about OR here. Let $R^{\varepsilon}(f^k_{\mathrm{OR}})$ denote the randomized communication complexity of computing $f(x,y_1) \vee f(x,y_2) \vee \cdots \vee f(x,y_k)$.

THEOREM 5.2. *In the message-passing model, for any function* $f : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ *and every distribution* μ *on* $\mathcal{X} \times \mathcal{Y}$ *such that* $\mu(f^{-1}(1)) \leq 1/10k$*, we have* $R^{1/3}(f^k_{\mathrm{OR}}) \geq \Omega(k/\log^2(1/\varepsilon) \cdot \mathrm{ED}^{\varepsilon}_{\mu}(f))$.

Proof. The reduction is the same as that in the proof of Theorem 5.1. Note that if $\mu(f^{-1}(1)) \leq 1/10k$, then with probability $(1-1/10k)^{k-1} \geq 0.9$, we have $f^k_{\mathrm{OR}}(x, y_1, \dots, y_k) = f(u,w)$. Similar to the proof of Lemma 4.3, we can repeat the reduction for $c\log(1/\varepsilon)$ times for some large enough constant c, and then the majority of $f^k_{\mathrm{OR}}(x, y_1, \dots, y_k)$'s will be equal to $f(u,w)$ with probability $1-\varepsilon/2$. Therefore, if we have a protocol for f^k_{OR} with error probability $\varepsilon/(2c \cdot \log 1/\varepsilon)$ under ν and communication complexity C, then we have a protocol for f with error probability $(c\log 1/\varepsilon)\cdot(\varepsilon/(2c\log 1/\varepsilon)+\varepsilon/2) = \varepsilon$ under μ and expected communication complexity $O(\log(1/\varepsilon) \cdot C/k)$. Consequently, $R^{1/3}(f^k_{\mathrm{OR}}) \geq \Omega\left(R^{\varepsilon/(2c\log 1/\varepsilon)}(f^k_{\mathrm{OR}})/\log(1/\varepsilon)\right) \geq \Omega\left(D^{\varepsilon/(2c\log 1/\varepsilon)}_{\nu}(f^k_{\mathrm{OR}})/\log(1/\varepsilon)\right) \geq \Omega\left(k/\log^2(1/\varepsilon) \cdot \mathrm{ED}^{\varepsilon}_{\mu}(f)\right)$. $\square$

6. Applications.

We now consider applications where multiparty communication complexity lower bounds such as ours are needed. As mentioned in the introduction, our multiparty communication problems are strongly motivated by research on the server-site model and the more general distributed streaming model. We discuss three problems here: the heavy-hitters problem, which asks us to find the approximately most frequently occurring elements in a set which is distributed among many clients; the distinct elements problems, which lists the distinct elements from a fixed domain where the elements are scattered across distributed databases possibly with multiplicity; and the ε-kernel problem, which asks us to approximate the convex hull of a set which is distributed over many clients.

Distinct elements. Consider a domain of n possible elements and k distributed databases each of which contains a subset of these elements. The exact distinct elements problem is to list the set of all distinct elements from the union of all elements across all distributed databases. This is precisely the k-OR problem and follows from Theorem 3.3, since the existence of each element in each distributed data point can be signified by a bit, and the bitwise OR represents the set of distinct elements.

THEOREM 6.1. *For a set of k distributed databases, each containing a subset of n elements, it requires $\Omega(nk)$ communication total between the databases to list the set of distinct elements with probability at least* $2/3$.

ε-kernels. Given a set of n points $P \subset \mathbb{R}^d$, the width in direction u is denoted by

$$\mathsf{wid}(P,u) = \left(\max_{p \in P}\langle p, u\rangle\right) - \left(\min_{p \in P}\langle p, u\rangle\right),$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product operation. Then an ε-kernel [2, 1] K is a subset of P so that for any direction u we have

$$\mathsf{wid}(P,u) - \mathsf{wid}(K,u) \leq \varepsilon \cdot \mathsf{wid}(P,u).$$

An ε-kernel K approximates the convex hull of a point set P, such that if the convex hull of K is expanded in any direction by an ε-factor it contains P. As such, this coreset has proven useful in many applications in computational geometry such as approximating the diameter and smallest enclosing annulus of point sets [2, 1]. It has been shown that ε-kernels may require $\Omega(1/\varepsilon^{(d-1)/2})$ points (on a $(d-1)$-sphere in $\mathbb{R}^d$) and can always be constructed of size $O(1/\varepsilon^{(d-1)/2})$ in time $O(n+1/\varepsilon^{d-3/2})$ [44, 11].

We note a couple of other properties about ε-kernels. Compposibility: If $K_1,\dots,K_k$ are ε-kernels of $P_1,\dots,P_k$, respectively, then $K=\bigcup_{i=1}^k K_i$ is an ε-kernel of $P=\bigcup_{i=1}^k P_i$. Transitivity: If K_1 is an ε_1-kernel of P and K_2 is an ε_2-kernel of K_1, then K_2 is an $(\varepsilon_1+\varepsilon_2)$-kernel of P. Thus it is easy to see that each site i can simply send an $(\varepsilon/2)$-kernel K_i of its data of size $n_\varepsilon = O(1/\varepsilon^{(d-1)/2})$ to the server, and the server can then create and $(\varepsilon/2)$-kernel of $\bigcup_{i=1}^k K_i$ of size $O(1/\varepsilon^{(d-1)/2})$. This is asymptotically the optimal size for and ε-kernel of the full distributed data set. We next show that this procedure is also asymptotically optimal in regards to communication.

Theorem 6.2. *For a distributed set of k sites, it requires $\Omega(k/\varepsilon^{(d-1)/2})$ communication total between the sites and the server for the server to create an ε-kernel of the distributed data with probability at least $2/3$.*

Proof. We describe a construction which reduces k-OR to this problem, where each of k players has $n_\varepsilon = \Theta(1/\varepsilon^{(d-1)/2})$ bits of information. Theorem 3.3 shows that this requires $\Omega(n_\varepsilon k)$ communication.

We let each player have very similar data $P_i=\{p_{i,1},\dots,p_{i,n_\varepsilon}\}$, each player's data points lie in a unit ball $B=\{q\in\mathbb{R}^d \mid \|q\|\le 1\}$. For each player, their n_ε points are in a similar position. Each player's jth point $p_{i,j}$ is along the same direction u_j, and its magnitude is either $\|p_{i,j}\|=1$ or $\|p_{i,j}\|=1-2\varepsilon$. Furthermore, the set of directions $U=\{u_i\}$ are well-distributed such that for any player i, and any point $p_{i,j}$ that $P_i \setminus p_{i,j}$ is not an ε-kernel of P_i; that is, the only ε-kernel is the full set. The existences of such a set follows from the known lower bound construction for size of an ε-kernel.

We now claim that the k-OR problem where each player has n_ε bits can be solved by solving the distributed ε-kernel problem under this construction. Consider any instance of k-OR, and translate to the ε-kernel problem as follows. Let the jth point $p_{i,j}$ of the ith player have norm $\|p_{i,j}\|=1$ when jth bit of the player is 1, and have norm $\|p_{i,j}\|=1-2\varepsilon$ if the jth bit is 0. By construction, an ε-kernel of the full set must acknowledge (and contain) the jth point from some player that has such a point with norm 1, if one exists. Thus the full ε-kernel encodes the solution to the k-OR problem: it must have n_ε points and, independently, the jth point has norm 1 if the jth OR bit is 1, and has norm $1-2\varepsilon$ if the jth OR bit is 0. □

Heavy hitters. Given a multiset S that consists of n elements, a threshold parameter ϕ, and an error parameter ε, the *approximate heavy-hitters* problem asks for a set of elements which contains all elements that occur at least ϕk times in S and contains no elements that occur fewer than $\phi k(1-\varepsilon)$ times in S. On a static nondistributed data set this can easily be done with sorting. This problem has been famously studied in the streaming literature where the Misra–Gries [35] and Space-Saving [33] summaries can solve the problem in optimal space. In the distributed setting the best known algorithms use random sampling of the indices and require either $O((1/\varepsilon^2)n\log n)$ or $O(k+\sqrt{k}n/\varepsilon\cdot\log n)$ communication to guarantee a correct set with constant probability [24]. We will prove a lower bound of $\Omega(n/\varepsilon)$ here. After our work Woodruff and Zhang [39] showed a lower bound of $\Omega(\min\{n/\varepsilon^2, \sqrt{k}n/\varepsilon\})$ (translating to our setting; their setting is a bit different), which is tight up to a log factor.

We now present a specific formulation of the approximate heavy-hitters problem as (k, ϕ, ε)-HH as follows. Consider k players, each with a bit sequence (either 0 or 1) of length n where each coordinate represents an element. The goal is to answer YES for each index with at least ϕk elements, NO for each index with no more than $\phi k(1-\varepsilon)$ elements, and either YES or NO for any count in between.

The reduction is based on a distribution $\tau_{\phi,\varepsilon}$ where independently each index has either ϕk or $\phi k(1-\varepsilon)$ 1 bits, each with probability 1/2. In the reduction the players are grouped into sets of $k\varepsilon$ players each, and all grouped players for each index are either given a 1 bit or all players are given a 0 bit. These 1 bits are distributed randomly among the $1/\varepsilon$ groups. The proof then uses Corollary 3.7.

THEOREM 6.3. $D^{1/6}_{\tau_{\phi,\varepsilon}}((k, \phi, \varepsilon)\text{-}HH) = \Omega(n/\varepsilon)$.

Proof. To lowerbound the communication for (k, ϕ, ε)-HH we first show another problem is hard: $(1/\varepsilon, \phi)$-HH (assume $1/\varepsilon$ is an integer). Here there are only $1/\varepsilon$ players, and each player at each index has a count of 0 or $k\varepsilon$, and we again want to distinguish between total counts of at least $k\phi$ (YES) and at most $k\phi(1-\varepsilon)$ (NO). By distribution $\tau_{\phi,\varepsilon}$, each index has a total of either $k\phi$ or $k\phi(1-\varepsilon)$ exactly. And then we distribute these bits to players so each player has precisely either 0 or $k\varepsilon$ at each index. When k is odd, this is precisely the $(1/\varepsilon, \phi)$-MAJ problem, which by Corollary 3.7 takes $\Omega(n/\varepsilon)$ communication.

Now it is easy to see that $D^{1/6}_{\tau_{\phi,\varepsilon}}((1/\varepsilon, \phi)\text{-HH}) \leq D^{1/6}_{\tau_{\phi,\varepsilon}}((k, \phi, \varepsilon)\text{-HH})$, since the former on the same input allows $(1/\varepsilon)$ sets of $k\varepsilon$ players to talk to each other at no cost. □

7. Concluding remarks.

In this paper we have introduced the symmetrization technique, and have shown how to use it to prove lower bounds for k-player communication games. This technique seems widely applicable, and we expect future work to find further uses.

7.1. A brief comparison to the icost method.

In this section we make a brief comparison between our symmetrization method and the celebrated *icost* method [10, 6]. Readers who are familiar with the icost method may notice that the k-XOR, k-MAJ, and blackboard k-OR/AND problems discussed in this paper can also be handled by the icost method. However, for problems whose complexities are different in the blackboard model and the message-passing model, e.g., k-OR and k-CONN, the icost method cannot be used to obtain tight lower bounds in the message-passing/coordinator model, while the symmetrization method still applies.

If we view the input to k players as a matrix with players as rows each having an n-bit input, the icost method first "divides" the whole problem to n copies of primitive problems columnwise, and then analyzes a single primitive problem. While the symmetrization method first reduces the size of a problem in the row space, that is, it first reduces a k-player problem to a 2-player problem, and then analyzes the 2-player problem. We can certainly use the icost method again when analyzing the resulting 2-player problem, which gives us an elegant way to combine these two techniques.

We notice that after the conference version of this paper, Braverman et al. [8] and Huang et al. [25] independently developed two new (and different) definitions for icost in the coordinator model, and used them to prove some tight lower bounds in the coordinator model.

7.2. Limitations and future directions.

The symmetrization technique also has several limitations, which we wish to discuss here.

First, there are problems that might be impossible to lower bound using symmetrization. Consider, for example, the k-player disjointness problem, where each player gets a subset of $\{1, \ldots, n\}$, and the goal is to decide whether there is an element that appears in all of the sets. This problem looks to be easier than the coordinatewise AND problem. But in the conference version of this paper we conjectured that this problem has a communication lower bound of $\Omega(nk)$ in the coordinator model as well. However, it seems impossible to prove this lower bound using symmetrization for the following reason. Suppose we give Alice the input of a randomly chosen player, and give Bob the inputs of all the other players. Then either there are a small number of interesting bits where Bob has all 1s which he can query of Alice, or there are many such bits where Bob has all 1s, and then with high probability he can guess that Alice will also have a 1 in at least one of those locations by symmetry. Recently Braverman et al. [8] confirmed our conjecture that the k-player disjointness problem has a lower bound $\Omega(nk)$ in the coordinator model, using a very different method via information complexity.

The second limitation is that in order to use symmetrization, one needs to find a hard distribution for the k-player problem which is symmetric. This is usually impossible when the problem itself is not symmetric, i.e., when the players have different roles. For example, one could envision a problem where some of the players get as input elements of some group and the rest of the players get as input integers. However, note that for such problems, symmetrization can still be useful in a somewhat-generalized version. For example, suppose there are two sets of players: in set P, the players get group elements, and in set P', the players get integers. Assume each of the sets contains exactly $k/2$ players. To use symmetrization, we would try to find a hard distribution that is symmetric inside of P and also inside of P'; namely, a distribution where permuting the players inside P has no effect on the distribution, and similarly for permuting the players inside P'. Then, to use symmetrization we can have Alice simulate two random players, p_i and p_j, where p_i is from P and p_j is from P'; Bob will simulate all the rest of the players. Now symmetrization can be applied. If, alternatively, the set P contained just 3 players and P' contained $k - 3$ players, we can have Alice simulate one of the players in P', Bob can simulate the rest of the players in P', and either Alice or Bob can play the three players from P. As can be seen, with a suitable choice of distribution, it should still be possible to apply symmetrization to problems that exhibit some amount of symmetry.

The main topic for future work seems to be to find more setting and problems where symmetrization can prove useful. Recently, this technique has found applications in several other statistical, numerical linear algebra, and graph problems in the message-passing/coordinator model [39, 40, 25, 41, 31]. We believe it has the potential to be a widely useful tool.

Appendix A. Omitted proof for 2-BITS.

Lemma 2.1 (restated). $\mathrm{ED}_{\zeta_\rho}^{1/3}(2\text{-}\mathit{BITS}) = \Omega(n\rho\log(1/\rho))$.

Proof. Here we will make use of several simple tools from information theory. Given a random variable X drawn from a distribution μ, we can measure the amount of randomness in X by its entropy $H(X) = -\sum_x \mu(x)\log_2\mu(x)$. The conditional entropy $H(X \mid Y) = H(XY) - H(Y)$ describes the amount of entropy in X, given that Y exists. The mutual information $I(X;Y) = H(X) + H(Y) - H(XY)$ measures the randomness in both random variables X and Y.

Let $\mathcal{P}$ be any valid communication protocol. Let X be Alice's (random) input vector. Let Y be Bob's output as the Alice's vector he learns after the communication.

Let Π be the transcript of $\mathcal{P}$. Let $\varepsilon = 1/3$ be the error bound allowed by Bob. First, since after running $\mathcal{P}$, with probability at least $1-\varepsilon$, we have $Y = X$, thus Bob knows X; in this case we say the protocol transcript Π is *good* (otherwise Π is *bad*). Therefore,

$$\begin{aligned}
I(X;Y \mid \Pi) &= \mathbf{Pr}[\Pi \text{ is good}] \cdot I(X;Y \mid \Pi \text{ is good}) + \mathbf{Pr}[\Pi \text{ is bad}] \cdot I(X;Y \mid \Pi \text{ is bad}) \\
&\leq 0 + \varepsilon \cdot I(X;Y \mid \Pi \text{ is bad}) \\
&= \varepsilon(H(X \mid \Pi \text{ is bad}) - H(X \mid Y, \Pi \text{ is bad})) \\
&\leq \varepsilon H(X \mid \Pi \text{ is bad}) \leq \varepsilon H(X).
\end{aligned}$$

Consequently,

$$\begin{aligned}
\varepsilon H(X) &\geq I(X;Y \mid \Pi) \\
&= H(X \mid \Pi) + H(Y \mid \Pi) - H(XY \mid \Pi) \\
&= H(X\Pi) - H(\Pi) + H(Y\Pi) - H(\Pi) \\
&\quad -(H(XY\Pi) - H(\Pi)) \\
&= H(X\Pi) + H(Y\Pi) - H(XY\Pi) - H(\Pi) \\
&= I(X\Pi; Y\Pi) - H(\Pi) \\
&\geq (1-\varepsilon)H(X\Pi) - H(\Pi) \\
&\geq (1-\varepsilon)H(X) - H(\Pi).
\end{aligned}$$

Therefore, $\mathbf{E}[|\Pi|] \geq H(\Pi) \geq (1-2\varepsilon)H(X) \geq \Omega(nH(\rho)) \geq \Omega(n\rho\log(1/\rho))$. □

Appendix B. Omitted proofs for the biased 2-party set disjointness.

Lemma 2.2 (restated). *When μ has $|x \cap y| = 1$ with probability $1/t$, then* $\mathrm{ED}_{\mu}^{1/100t}(2\text{-}\mathsf{DISJ}) = \Omega(n)$.

The proof is based on [38]. Before giving the proof, we first introduce some notation and a key technical lemma. Define

$$A = \{(x,y) \ : \ (\mu(x,y) > 0) \wedge (x \cap y = \emptyset)\}$$

and

$$B = \{(x,y) \ : \ (\mu(x,y) > 0) \wedge (x \cap y \neq \emptyset)\}.$$

Thus $\mu(A) = 1 - 1/t$ and $\mu(B) = 1/t$. We need the following key lemma, which is an easy extension of the main lemma in Razbarov [38] by rescaling the measures on the YES and NO instances.

Lemma B.1 (see [38]). *Let A, B, μ be defined as above. Let $R = C \times D$ be any rectangle in the communication protocol. Then we have $\mu(B \cap R) \geq 1/40t \cdot \mu(A \cap R) - 2^{-0.01n}$.*

Proof for Lemma 2.2. Let $\mathcal{R} = \{R_1, \ldots, R_t\}$ be the minimal set of disjoint rectangles in which the protocol outputs "1," i.e, $x \cap y = \emptyset$. Imagine that we have a binary decision tree built on top of these rectangles. If we can show that there exists $\mathcal{O} \subseteq \mathcal{R}$ such that $\mu(\bigcup_{R_i \in \mathcal{O}} R_i) \geq 0.5 \cdot \mu(\bigcup_{R_i \in \mathcal{R}} R_i)$ and each of $R_i \in \mathcal{O}$ lies on a depth at least $0.005n$ in the binary decision tree, then we are done. Since $\mu(\bigcup_{R_i \in \mathcal{O}} R_i) \geq 0.5 \cdot \mu(\bigcup_{R_i \in \mathcal{R}} R_i) \geq 0.5 \cdot (\mu(A) - 1/100t) = \Omega(1)$ and querying inputs in each rectangle in $\mathcal{O}$ costs $\Omega(n)$ bits.

We prove this by contradiction. Suppose that there exists $\mathcal{O}' \subseteq \mathcal{R}$ such that $\mu(\bigcup_{R_i \in \mathcal{O}'} R_i) > 0.5 \cdot \mu(\bigcup_{R_i \in \mathcal{R}} R_i)$ and each of $R_i \in \mathcal{O}'$ lies on a depth less than $0.005n$ in the binary decision tree. We have the following two facts.

1. There are at most $2^{0.005n}$ disjoint rectangles that lie on depths less than $0.005n$, i.e., $|\mathcal{O}'| \leq 2^{0.005n}$.
2. $\mu\left(\bigcup_{R_i \in \mathcal{O}'}(R_i \cap A)\right) > 0.5 - 1/100t$.

Combining the two facts with Lemma B.1 we reach the following contradiction of our error bound:

$$\begin{aligned}\mu\left(\bigcup_{i=1}^{t}(R_i \cap B)\right) &\geq \mu\left(\bigcup_{R_i \in \mathcal{O}'}(R_i \cap B)\right)\\ &\geq \sum_{R_i \in \mathcal{O}'}(1/40t \cdot \mu(R_i \cap A) - 2^{-0.01n})\\ &>1/40t \cdot (0.5 - 1/100t) - 2^{0.005n} \cdot 2^{-0.01n}\\ &>1/100t\ . \qquad \square\end{aligned}$$

Appendix C. Omitted proofs graph connectivity. We provide here a full proof for the probability of the event ξ_1 that both subset of the graph L and R are connected.

Lemma 4.2 (restated). *ξ_1 happens with probability at least $1 - 1/2n$ when $k \geq 68 \ln n + 1$.*

Proof. First, note that by our construction both $|L|$ and $|R|$ are $\Omega(n)$ with high probability. To locally simplify notation, we consider a graph (V, E) of n nodes where edges are drawn in $(k-1) \geq 68 \ln n$ rounds, and each round $n/4$ disjoint edges are added to the graph. If (V, E) is connected with probability $(1 - 1/4n)$, then by union bound over $\cup_{j=2}^{k} I_j \bigcap E_L$ and $\cup_{j=2}^{k} I_j \bigcap E_R$, ξ_1 is true with probability $(1 - 1/2n)$. The proof follows four steps:

(S1): *All points have degree at least $8 \ln n$.* Since for each of k rounds each point's degree increases by 1 with probability $1/2$, then the expected degree of each point is $14 \log n$ after the first $28 \ln n$ rounds. A Chernoff–Hoeffding bound says that the probability that a point has degree less than $8 \ln n$ is at most $2\exp(-2(6 \ln n)^2/\ (14 \ln n)) \leq 2\exp(-5 \ln n) \leq 2/n^5 \leq 1/12n^2$. Then by the union bound, this holds for none of the n points with probability at least $1 - 1/12n$.

(S2): *Conditioned on (S1), any subset $S \subset V$ of $h < n/10$ points is connected to at least $\min\{h \ln n, n/10\}$ distinct points in $V \setminus S$.* At least $9n/10$ points are outside of S, so each point in S expects to be connected at least $(9/10)8 \ln n \geq 7 \ln n$ times to a point outside of S. Each of these edges occur in different rounds, so they are independent. Thus we can apply a Chernoff–Hoeffding bound to say the probability that the number of edges outside of S for any point is less than $3 \ln n$ is at most $2\exp(-2(4 \ln n)^2/(8 \ln n)) = 2\exp(-4 \ln n) = 2/n^4$. Thus the probability that no point in S has fewer than $\ln n$ edges outside S is (since $h < n/10$) at most $1/5n^3$.

If the $h \cdot 3 \ln n$ edges outside of S (for all h points) are drawn independently at random, then we need to bound the probability that these go to more than $n/10$ distinct points or $h \ln n$ distinct points. Since the edges are drawn to favor going to distinct points in each round, it is sufficient to analyze the case where all of the edges are independent, which can only increase the chance they collide. In either case $h \ln n < n/10$ or $h \ln n > n/10$ each time an edge is chosen (until $n/10$ vertices have been reached, in which case we can stop), $9/10$ of all possible vertices are outside the set of edges already connected to. So if we select the $3h \ln n$ edges one at a time, each event connects to distinct points with probability at least $9/10$, so we expect at least $(9/10)(3h\ln n) > 2h\ln n$ distinct points. Again by a Chernoff–Hoeffding bound, the probability that fewer than $h\ln n$ distinct points have been reached is at most $2\exp(-2(h\ln n)^2/(3hn)) \leq 2\exp(-(2/3)h\ln n) < 2\cdot\exp(-5\ln n) \leq 2/n^5 \leq 1/5n^2$ (for $h \geq 8$). Together the probability of these events not happening is at most $1/2n^2$.

(S3): *There is a single connected component S_G of size at least $n/10$.* Start with any single point, we know from step 1 its degree is at least $8\ln n$. Then we can consider the set S formed by these $h_1 = 8\ln n$ points, and apply step 2 to find another $h_1 \ln n = 8\ln^2 n = h_2$ points; add these points to S. The process iterates and at each round $h_i = 8\ln^i n$, by growing only from h_i the newly added points. So, by round $i = \ln n$ the set $S = S_G$ has grown to at least size $n/10$. Taking the union bound over these $\ln n$ rounds shows that this process fails with probability at most $1/12n$.

(S4): *All points in $V \setminus S_G$ are connected to S_G.* Each round point p is connected to S_G with probability at least $1/20$. So by coupon collector's bound, using the last $40\ln n$ rounds all points are connected after $2\ln n$ sets of 20 rounds with probability at least $1 - 1/12n$.

By union bound, the probability that steps (S1), (S3), and (S4) are successful is at least $1 - 1/4n$, proving our claim. □

Acknowledgments. We would like to thank Andrew McGregor for helpful discussions, and for suggesting studying the connectivity problem. We also thank David Woodruff for some insightful comments. We would like to thank the SODA and SICOMP anonymous reviewers for their insightful reviews.

REFERENCES

[1] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan, *Geometric approximations via coresets*, in Combinatorial and Computational Geometry, Math. Sci. Res. Inst. Publ., 52, Cambridge Univ. Press, Cambridge, 2005, pp. 1–30.

[2] P. K. Agarwal, S. Har-Peled, and K. R. Varadarajan, *Approximating extent measure of points*, J. ACM, 51 (2004), pp. 606–635.

[3] N. Alon, Y. Matias, and M. Szegedy, *The space complexity of approximating the frequency moments*, J. Comput. Syst. Sci., 58 (1999), pp. 137–147.

[4] C. Arackaparambil, J. Brody, and A. Chakrabarti, *Functional monitoring without monotonicity*, in Automata, Languages and Programming, Part I, Lecture Notes in Comput. Sci. 5555, Springer, Berlin, 2009, pp. 95–106.

[5] B. Babcock and C. Olston, *Distributed top-k monitoring*, in Proceedings of the ACM SIGMOD International Conference on Management of Data, 2003, pp. 28–39.

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

[7] B. Barak, M. Braverman, X. Chen, and A. Rao, *How to compress interactive communication*, in Proceedings of the ACM Symposium on Theory of Computing, ACM, New York, 2010, pp. 67–76.

[8] M. Braverman, F. Ellen, R. Oshman, T. Pitassi, and V. Vaikuntanathan, *A tight bound for set disjointness in the message-passing model*, in Proceedings of the IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2013, pp. 668–677.

[9] P. Cao and Z. Wang, *Efficient top-k query calculation in distributed networks*, in Proceedings of the ACM Symposium on Principles of Distributed Computing, ACM, New York, 2004, pp. 206–215.

[10] A. Chakrabarti, Y. Shi, A. Wirth, and A. C.-C. Yao, *Informational complexity and the direct sum problem for simultaneous message complexity*, in Proceedings of the IEEE Sym-

posium on Foundations of Computer Science, IEEE Computer Soc., Los Alamitos, CA, 2001, pp. 270–278.

[11] T. Chan, *Faster core-set constructions and data-stream algorithms in fixed dimensions*, Comput. Geom., 35 (2006), pp. 20–35.

[12] A. Chattopadhyay, J. Radhakrishnan, and A. Rudra, *Topology matters in communication*, in Proceedings of the IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2014, pp. 631–640.

[13] G. Cormode and M. Garofalakis, *Sketching streams through the net: Distributed approximate query tracking*, in Proceedings of the 31st International Conference on Very Large Data Bases, VLDB Endowment, 2005, pp. 13–24.

[14] G. Cormode, M. Garofalakis, S. Muthukrishnan, and R. Rastogi, *Holistic aggregates in a networked world: Distributed tracking of approximate quantiles*, in Proceedings of the ACM SIGMOD International Conference on Management of Data, ACM, New York, 2005, pp. 25–36.

[15] G. Cormode, S. Muthukrishnan, and K. Yi, *Algorithms for distributed functional monitoring*, in Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, 2008, pp. 1076–1085.

[16] G. Cormode, S. Muthukrishnan, K. Yi, and Q. Zhang, *Optimal sampling from distributed streams*, in Proceedings of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, ACM, New York, 2010, pp. 77–86.

[17] G. Cormode, S. Muthukrishnan, and W. Zhuang, *What's different: Distributed, continuous monitoring of duplicate-resilient aggregates on data streams*, in Proceedings of the 22nd IEEE International Conference on Data Engineering, IEEE, Washington, DC, 2006, article 57.

[18] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, New York, 1991.

[19] P. Duris and J. D. P. Rolim, *Lower bounds on the multiparty communication complexity*, J. Comput. System Sci., 56 (1998), pp. 90–95.

[20] A. Gál and P. Gopalan, *Lower bounds on streaming algorithms for approximating the length of the longest increasing subsequence*, in Proceedings of the IEEE Symposium on Foundations of Computer Science, 2007, pp. 294–304; SIAM J. Comput., 39 (2010), pp. 3463–3479.

[21] O. Goldreich, *Secure multi-party computation*, Working draft, 2002, available at http://www.wisdom.weizmann.ac.il/~oded/pp.html.

[22] S. Guha and Z. Huang, *Revisiting the direct sum theorem and space lower bounds in random order streams*, in Proceedings of the International Colloquium on Automata, Languages and Programming, 2009, Lecture Notes in Comput. Sci. 5555, Springer, Berlin, 2009, pp. 513–524.

[23] S. Guha and A. McGregor, *Tight lower bounds for multi-pass stream computation via pass elimination*, in Proceedings of the International Colloquium on Automata, Languages and Programming, 2008, Lecture Notes in Comput. Sci. 5125, Springer, Berlin, 2008, pp. 760–772.

[24] Z. Huang, K. Yi, Y. Liu, and G. Chen, *Optimal sampling algorithms for frequency estimation in distributed data*, in 2011 Proceedings IEEE INFOCOM, IEEE, Washington, DC, 2011, pp. 1997–2005.

[25] Z. Huang, B. Radunovic, M. Vojnovic, and Q. Zhang, *Communication Complexity of Approximate Maximum Matching in Distributed Graph Data*, Tech. report MSR-TR-2013-35, Microsoft Research, Redmond, WA, 2013.

[26] Z. Huang, K. Yi, and Q. Zhang, *Randomized algorithms for tracking distributed count, frequencies, and ranks*, in Proceedings of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, 2012, ACM New York, pp. 295–306.

[27] B. Kalyanasundaram and G. Schintger, *The probabilistic communication complexity of set intersection*, SIAM J. Discrete Math., 5 (1992), pp. 545–557.

[28] M. Karchmer, R. Raz, and A. Wigderson, *Super-logarithmic depth lower bounds via the direct sum in communication complexity*, Comput. Complexity, 5 (1995), pp. 191–204.

[29] R. Keralapura, G. Cormode, and J. Ramamirtham, *Communication-efficient distributed monitoring of thresholded counts*, in Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data, ACM New York, 2006, pp. 289–300.

[30] E. Kushilevitz and N. Nisan, *Communication Complexity*, Cambridge University Press, Cambridge, UK, 1997.

[31] Y. Li, X. Sun, C. Wang, and D. P. Woodruff, *On the communication complexity of linear algebraic problems in the message passing model*, Distributed Computing, Lecture Notes in Comput. Sci. 8784, Fabian Kuhn, ed., Springer, Heidelberg, 2014, pp. 499–513.

[32] A. Manjhi, V. Shkapenyuk, K. Dhamdhere, and C. Olston, *Finding (recently) frequent items in distributed data streams*, in Proceedings of the 21st International Conference on Data Engineering, 2005, IEEE Computer Society Washington, DC, 2005, pp. 767–778.

[33] A. Metwally, D. Agrawal, and A. El Abbadi, *An integrated efficient solution for computing frequent and top-*k *elements in data streams*, ACM Trans. Database Syst., 31 (2006), pp. 1095–1133.

[34] S. Michel, P. Triantafillou, and G. Weikum, *Klee: A framework for distributed top-k query algorithms*, in Proceedings of the 31st International Conference on Very Large Databases, 2005, pp. 637–648.

[35] J. Misra and D. Gries, *Finding repeated elements*, Sci. Comput. Programming, 2 (1982), pp. 143–152.

[36] M. Patrascu, *Towards polynomial lower bounds for dynamic problems*, in Proceedings of the 2010 ACM Symposium on Theory of Computing, 2010, ACM, New York, 2010, pp. 603–609.

[37] B. Patt-Shamir and A. Shafrir, *Approximate distributed top-k queries*, Distrib. Comput., 21 (2008), pp. 1–22.

[38] A. A. Razborov, *On the distributional complexity of disjointness*, Theoret. Comput. Sci., 106 (1992), pp. 385–390.

[39] D. P. Woodruff and Q. Zhang, *Tight bounds for distributed functional monitoring*, in Proceedings of the ACM Symposium on Theory of Computing, 2012, ACM, New York, 2012, pp. 941–960.

[40] D. P. Woodruff and Q. Zhang, *When distributed computation is communication expensive*, in Proceedings of the International Symposium on Distributed Computing, 2013, Lecture Notes in Comput. Sci. 8205, Springer, Heidelberg, 2013, pp. 16–30.

[41] D. P. Woodruff and Q. Zhang, *An optimal lower bound for distinct elements in the message passing model*, 2014, SIAM, Philadelphia, 2014, pp. 718–733.

[42] A. C.-C. Yao, *Probabilistic computations: Toward a unified measure of complexity (extended abstract)*, in Proceedings of the 18th Annual Symposium on Foundations of Computer Science, 1977, IEEE Computer Society, Long Beach, CA, pp. 222–227.

[43] K. Yi and Q. Zhang, *Optimal tracking of distributed heavy hitters and quantiles*, in Proceedings of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, 2009, pp. 167–174; Algorithmica 65 (2013), pp. 206–223.

[44] H. Yu, P. K. Agarwal, R. Poreddy, and K. R. Varadarajan, *Practical methods for shape fitting and kinetic data structures using coresets*, in Proceedings of the ACM Symposium on Computational Geometry, 2004, pp. 263–272; Algorithmica 52 (2008), pp. 378–402.

[45] Q. Zhao, M. Ogihara, H. Wang, and J. Xu, *Finding global icebergs over distributed data sets*, in Proceedings of the ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, 2006, ACM, New York, 2006, pp. 298–307.