Information Dissemination via Broadcasts in the Presence of Adversarial Noise

Klim Efremenko
Ben-Gurion University of the Negev, Beer-Sheva, Israel

Gillat Kol
Princeton University, NJ, USA

Dmitry Paramonov
Princeton University, NJ, USA

Ran Raz
Princeton University, NJ, USA

Raghuvansh R. Saxena
Tata Institute of Fundamental Research, Mumbai, India

Abstract

We initiate the study of *error correcting codes* over the multi-party *adversarial broadcast channel*. Specifically, we consider the classic *information dissemination problem* where n parties, each holding an input bit, wish to know each other's input. For this, they communicate in rounds, where, in each round, one designated party sends a bit to all other parties over a channel governed by an adversary that may corrupt a constant fraction of the received communication. We mention that the dissemination problem was studied in the *stochastic* noise model since the 80's.

While stochastic noise in multi-party channels has received quite a bit of attention, the case of adversarial noise has largely been avoided, as such channels cannot handle more than a $\frac{1}{n}$-fraction of errors. Indeed, this many errors allow an adversary to completely corrupt the incoming or outgoing communication for one of the parties and fail the protocol. Curiously, we show that by eliminating these "trivial" attacks, one can get a simple protocol resilient to a constant fraction of errors. Thus, a model that rules out such attacks is both necessary and sufficient to get a resilient protocol.

The main shortcoming of our dissemination protocol is its length: it requires $\Theta(n^2)$ communication rounds whereas n rounds suffice in the absence of noise. Our main result is a matching lower bound of $\Omega(n^2)$ on the length of any dissemination protocol in our model. Our proof first "gets rid" of the channel noise by converting it to a form of "input noise", showing that a noisy dissemination protocol implies a (noiseless) protocol for a version of the *direct sum gap-majority* problem. We conclude the proof with a tight lower bound for the latter problem, which may be of independent interest.

2012 ACM Subject Classification Theory of computation → Communication complexity

Keywords and phrases Radio Networks, Interactive Coding, Error Correcting Codes

Digital Object Identifier 10.4230/LIPIcs.CCC.2024.19

Funding *Klim Efremenko*: Supported by European Research Council Grant No. 949707.
Gillat Kol: Supported by the National Science Foundation CAREER award CCF-1750443.
Ran Raz: Supported by a Simons Investigator Award and by the National Science Foundation grant No. CCF-2007462.

1 Introduction

We initiate the study of *error correcting codes* over the multi-party *adversarial broadcast channel*, where n parties take turns broadcasting a bit to all other parties, but an adversary may corrupt a constant fraction of the received bits. Multi-party broadcast channels were

39th Computational Complexity Conference (CCC 2024).
Editor: Rahul Santhanam; Article No. 19; pp. 19:1–19:33
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

studied under various noise models in many recent works. However, almost all prior work assumed that the noise is stochastic, meaning that each sent message is corrupted with some small constant probability.

The reason the adversarial noise model has received considerably less attention is due to the fundamental limitation that any scheme, regardless of its rate, cannot withstand an adversarial noise rate exceeding $\frac{1}{n}$. To illustrate, with a budget of $\frac{1}{n}$-fraction of corruptions, the adversary can corrupt all messages broadcast by the participant who communicates the least, thereby obstructing the other parties from successfully computing a function that relies on this individual's input. Likewise, within the same budget, the adversary can disrupt all messages received by one of the participants, preventing them from producing a correct output.

The starting point of this paper is the observation that by excluding the two simple adversarial attacks mentioned earlier, we can circumvent the nonexistence of protocols capable of withstanding adversarial corruptions beyond a fraction of $\frac{1}{n}$. Specifically, we consider the adversarial channel where the adversary can corrupt any number of messages, provided that they do not corrupt more than a θ-fraction of the messages received by each party and a θ-fraction of the messages sent by each party[1], for some constant $\theta > 0$. We call such adversaries *θ-limited.*

Indeed, consider the following simple protocol for the *information dissemination* problem, where the input to each party is a bit and all parties wish to know all inputs: in the first half of the protocol, each party broadcasts their input bit the same number of times, and then each party computes the majority of the bits they received from each of the other parties. In the second half of the rounds, each party broadcasts an error correcting code of all the majority bits they computed. It is not hard to show that this protocol is resilient to θ-limited adversaries for some constant θ.[2]

Although our protocol exhibits good error resilience, a notable drawback is its rate, which is at most $\frac{1}{n}$. This is because, in the second half of the protocol, each of the n parties broadcasts the encoding of all n of their majority bits with an error correcting code, resulting in a length of at least n^2 bits. This situation prompts the following question:

> *Is there an information dissemination protocol robust to θ-limited adversaries with constant rate, or at least $\omega(\frac{1}{n})$ rate?*

1.1 Our Result

We answer this question in the negative, showing that the above simple protocol is essentially optimal.

▶ **Theorem 1** (Informal; see Theorem 6). *Every protocol for information dissemination that is resilient to a* 0.01*-limited adversary has length* $\Omega(n^2)$.

[1] More formally, if a party broadcasts in t rounds, then the adversary may corrupt up to θtn out of the tn messages received by the parties in those t rounds.

[2] The argument is that for every $i \in [n]$, in each half of the protocol at most 2θ-fraction of the messages received by party i are corrupted. Thus, party i correctly decodes at least $(1 - \mathcal{O}(\theta))$-fraction of the error correcting codes sent in the second half. This means that if party i outputs a wrong guess for player j's input, then the bit for party j must be incorrect in at least $(\frac{1}{2} - \mathcal{O}(\theta))$-fraction of the broadcast codes. This means that at least $\theta' = (\frac{1}{4} - \mathcal{O}(\theta))$-fraction of the transmissions of party j in the first half were corrupted. By choosing θ such that $\theta' > 2\theta$, we get that the output of all the parties is correct. See Section 4.

Technique

The proof of Theorem 1 consists of two steps. The first step *converts the noise in the channel to noise in the inputs.* Roughly speaking, we show that any protocol for information dissemination in our noisy model implies a protocol for solving a direct-sum gap-majority problem in the absence of noise[3]. Here, one copy of the gap-majority problem, denoted GapMaj_n, is the following: each party gets a bit with the promise that at least 0.9-fraction of the parties received the same bit. The parties' goal is to all output the majority bit. In the direct sum gap-majority problem, denoted here GapMaj_n^m, each party gets m bits, and we are promised that, for all j, the j-th bits of all the parties are an instance for GapMaj_n. Consider that one can view the majority bit for each copy j as the "true j-th bit" and view the j-th input bit of each player as a *noisy version* of this bit. In this sense, this indeed converts the noise in the channel to a noise in the inputs.

We then proceed to prove a lower bound on the communication cost of GapMaj_n^m over the noiseless channel. We show:

▶ **Theorem 2** (Informal; see Theorem 10)**.** *Every protocol for* GapMaj_n^m *with* $m = \Theta(n)$*, has length* $\Omega(n^2)$*.*

We note that our above definition of the direct sum problem is different from other definitions in the literature on one crucial point: *if the promise is violated, even for a single copy, any output is accepted.* In other words, as is usually the case, if all m copies satisfy the promise, the parties need to solve all copies. However, if the promise is violated for some of the copies, we don't require the protocol to solve the copies on which the promise does hold. Since our definition is easier to be satisfied by an algorithm, the lower bound in Theorem 2 is stronger. Because the relevant, known direct sum theorems only rule out algorithms that solve all the copies where the promise is satisfied, they are insufficient for our purpose (see more about this in Sections 1.2 and 2.2).

To prove our lower bound, we first note that Theorem 2 is essentially tight, as with $\mathcal{O}(n^2)$ communication, the parties can exchange their entire input. Moreover, as at least $\Omega(n)$ parties need to speak to solve the single-copy problem, Theorem 2 implies that the best protocol essentially solves each copy separately. Put differently, one can say that Theorem 2 is equivalent to the statement that trying to correlate the copies of gap-majority does not help the parties in solving the GapMaj_n^m problem. Interestingly, our proof establishes this in a pretty strong sense, showing that even trying to correlate three copies does not help the protocol, as it shows the result only assuming that the copies are *pairwise independent* at the end of the protocol.

In other words, we prove that the only way to make progress towards solving the GapMaj_n^m problem is to try to create a lot of correlations between pairs of copies. However, as this is only a constant factor away from giving a lot of information about individual copies (which is bounded by the overall communication), the number of these correlations can also be bounded by a constant times the overall communication, hence the lower bound. For more details, see Section 2.

[3] We mention that this step is inspired by the beautiful work of [32] that gives a lower bound under stochastic noise by lower bounding a different problem where the noise is in the inputs. However, our implementation is largely different, see Section 2.1.

1.2 Related Work

Dissemination over the stochastic broadcast channel

El Gamal [25] initiated the study of the noisy broadcast model as a simple abstraction for the effect of noise on highly distributed wireless systems. The noise in his model was stochastic – in each round the bit received by each party is flipped with some constant probability $\epsilon > 0$, independently. El Gamal asked whether there is a communication-efficient information dissemination protocol over this channel. The answer came from Gallager [24], who gave an elegant $\mathcal{O}(n \log \log n)$-round protocol, which was later proved to be optimal by the beautiful paper [32]. Variants of El Gamal's stochastic noisy broadcast channel were studied in many follow up works [24, 48, 42, 23, 45, 32, 10, 17, 14, 15]. We mention that our initial motivation for the study in this paper was the question of whether communication-efficient protocols like Gallager's were also possible in the presence of adversarial noise.

Interactive coding

In this work we consider the information dissemination problem, which is, perhaps, the most basic multi-party problem. It can be viewed as a generalization of the classical coding task to the multi-sender, multi-receiver setting (in traditional coding there is a single sender and a single receiver and the goal is to transfer a message from the sender to the receiver). It can be shown that a dissemination protocol with a certain resilience implies a protocol for any other problem, as the parties can first exchange their inputs and then compute the output themselves. Of course, this protocol is not always practical, as the input size may be much greater than the communication required to compute a solution to the problem.

The field of *interactive coding* aims to make this practical by converting (general) protocols designed to work over a noiseless channel to noise resilient protocols with a small overhead in the communication. The study of interactive codes was initiated by a seminal paper of Schulman [47] that considered two-party protocols and was the topic of many works since. Interactive codes for multi-party distributed channels were also studied, including codes for peer-to-peer networks [46, 31, 36, 43, 35, 2, 5, 1, 28, 9, 29, 30] and codes for various types of broadcast channels [24, 48, 42, 23, 45, 32, 10, 17, 11, 18, 20, 14, 44, 15, 16].

Peer-to-peer with adversarial noise

In this paper, we consider the broadcast channel under adversarial noise. The case of adversarial noise was previously considered in different peer-to-peer settings, where the parties are nodes in a graph and a node can send (potentially different) messages to its neighbors.

The work of [35] gives an interactive coding result in the synchronous, "fully utilized" model, where the communication is in rounds, and in each round each node sends a message to all other nodes. They show a scheme for converting any noiseless protocol to a protocol that is robust against $\Theta(\frac{1}{n})$-fraction of adversarial errors with multiplicative overhead of $\mathcal{O}((|E| \log n)/n)$ in the communication, where $|E|$ is the number of edges in the graph and n is the number of vertices. [36] consider the synchronous, non-fully-utilized model, and show that if the network graph contains the star topology, then a noiseless protocol can be converted to a protocol that is robust against $\Theta(\frac{1}{n})$-fraction of adversarial errors and has length linear in the length of the original protocol. [43] improve the communication balance of the [36] scheme. [9] consider the asynchronous setting and give an interactive coding scheme with error resilience $\Theta(\frac{1}{n})$ and multiplicative overhead of $\mathcal{O}(n \log^2 n)$ in the communication.

In all the above results, the noise tolerance of $\Theta(\frac{1}{n})$ is optimal, up to constants. For example, it is noted by [36] that, "by investing $\frac{1}{n}$-fraction of error an adversary can completely silence a party (the quietest party)". Recall that we get around this attack by forcing the adversary to not corrupt more than a constant fraction of the bits sent/received by any party.

A more challenging peer-to-peer channel where the adversarial noise can insert, delete, or alter communicated messages is considered in [30]. However, the obtained noise tolerance is $\Theta(\frac{1}{|E|\log|E|})$, which is even smaller than $\Theta(\frac{1}{n})$. Another line of work considers *oblivious* adversaries in peer-to-peer models [1, 29, 30]. Oblivious adversaries are not allowed to see the content of the communication channel when making their decision of what messages to corrupt. See more about that in Section 1.3.

Direct sum

The direct sum problem in communication complexity asks whether the communication required to solve k independent copies of a communication task is k times the communication required for solving a single copy. This problem has a rich history and was studied in several different settings (e.g., in the deterministic, non-deterministic, randomized, and distributional settings) and for different types of problems (relations, complete functions).

Currently, non-deterministic communication complexity is the best understood model in this regard, and an "almost perfect" direct sum theorem is known. The work of [22] and [40] showed that solving k copies of a relation R takes almost k times the amount of non-deterministic communication. More formally, $N(R^k) \geq k(N(R) - \log n - \mathcal{O}(1))$, where n is the number of bits required to describe an input for R, $N(R')$ denotes the non-deterministic communication complexity of R', and R^k is the problem of solving k instances of R simultaneously. Note that if R represents a partial (or promise) problem, then solving R^k means giving the correct output on all the copies where the promise is satisfied.

For deterministic communication complexity, denoted D, and *total* functions f, [22] show a weaker direct sum theorem $D(f^k) \geq k(\sqrt{D(f)/2} - \log n - \mathcal{O}(1))$.

The direct sum problem (and related problems like the direct product and XOR lemmas) were extensively studied in the randomized settings and are known to be related to other questions in complexity theory, like parallel repetition theorems and interactive compression schemes, [12, 39, 3, 34, 41, 6, 4, 8, 7, 37, 38, 26, 27, 49, to cite a few]. We mention that [26, 27] show that perfect direct sum does not hold for randomized communication complexity, however, weak direct sum theorems are known to hold, see, e.g., [4].

1.3 Additional Discussion and Future Directions

In this work we study the power and limitations of θ-limited adversaries in the broadcast model. We next discuss some of our modeling decisions and suggest other related questions.

Non-adaptive *vs.* adaptive protocols

In this work we follow the footsteps of El Gamal [25] and the followup works and assume that the order of communication in the protocol is *predetermined* and is independent of the players' inputs and the channel noise (and therefore also independent of the parties' received transcripts). Such protocols are called *non-adaptive*. Non-adaptive protocols are widely studied as they model certain common types of wireless networks, prevent *signaling*[4], and can trivially ensure that exactly one party is broadcasting in every round.

[4] Signaling is the situation in which information is inferred from whether a certain party has broadcast or not, rather than from the content of their communicated message.

Inspired by the radio network models in distributed computing [13], many recent works consider *adaptive* models, where a party decides whether to broadcast or not based on their input and their received transcript, see, e.g., [33, 10, 17, 11, 18, 19, 20, 21, 16]. As hinted above, such a model is prone to collision rounds (where more than one party broadcasts) and silent rounds (where no party broadcasts).

Note, however, that all the above mentioned adaptive models assume stochastic noise, and that it is *unclear how to adapt the definition of θ-limited adversaries to adaptive settings*. The main issue is that, for every i, our limited adversaries are only allowed to corrupt a θ-fraction of the total number of the messages received in rounds where party i broadcasts. But for adaptive models this number may not be fixed. Extending the notion of θ-limited adversaries to adaptive protocols is an intriguing question that may be motivated by the fact that, in various settings, adaptive protocols were shown to be much "stronger" than non-adaptive ones, see, e.g., [33, 17, 19, 21].

Interactive coding with limited adversaries

In this paper, we study the dissemination problem with θ-limited adversaries. As discussed in Section 1.2, such dissemination protocols imply a protocol for solving any other communication task with θ-limited adversaries, but the blow-up in communication may be substantial. In other words, interactive codes (with bad rate) are possible with θ-limited adversaries. It can likely be shown that, in some cases, such blow-up cannot be avoided[5]. An interesting goal is to find the "minimum additional restrictions" to be posed on the adversary that would allow for interactive coding with low overhead.

Randomness in adversarial models

Our simple dissemination protocol and our lower bound in Theorem 1, as well as most of the study of error correcting codes over adversarial channels in the literature, assume the *deterministic* setting. One can also consider *randomized* settings, where the parties share a random string. Note, however, that if this string is known to the adversary at the beginning of the protocol, then the protocol is essentially deterministic. On the other hand, if the random string is unknown to the adversary for the entire duration of the protocol, then the parties may use parts of it as one-time pads and ensure that the adversary is *oblivious* to the contents of the messages. Such adversaries are known to be weak (at least in the peer-to-peer setting) and every noiseless protocol can be simulated in the presence of such adversaries with only a constant overhead in the communication, see, e.g., [1, 29, 29].

An interesting direction for future work is to consider *"intermediate models"* where, for example, fresh randomness is sampled in every round, the adversary gets to see it immediately after it is sampled, but the adversary does not get to see randomness in future rounds when deciding on what corruptions to make. Can we design communication-efficient protocols in such models?

[5] Consider, for example, the pointer chasing protocol on a tree of depth n, where party i has an edge coming out from each of the vertices in level i in the tree and the parties wish to find the unique root-to-leaf path contained in the union of their edges. While the parties can exchange their huge inputs and get constant resilience, an attempt to simulate the noiseless chasing protocol directly will result in noise resilience $o(1)$. Indeed, the adversary that erases the communication to the second party in the first θ-fraction of the rounds, then erases the third party for the next θ-fraction of the rounds, *etc.*, is θ-limited, but prevents the parties from computing the correct output. We mention that [36] show similar limitations in the peer-to-peer setting.

The adversarial erasure channel

In this work we have allowed the adversary to *corrupt*, or flip, some of the received bits. Can better protocols be designed for the easier setting where the adversary is only allowed to erase some of the received bits?

Noise tolerance

What is the maximum noise tolerance of dissemination protocols in our model? That is, what is the largest fraction of errors that can be handled by dissemination protocols? What is the rate vs. tolerance tradeoff?

2 Proof Overview

In this section, we give a detailed overview of our proof for Theorem 1. For the rest of this section we set $\theta = 0.01$. Let GapMaj_n be the following n-party problem: each of the n parties gets an input bit with the promise that at least $(1-2\theta)$-fraction of the input bits agree. The goal is for all parties to output the majority bit. Let GapMaj_n^m be the problem where each of the n parties gets m input bits with the promise that the j-th bit of all the parties is an instance of GapMaj_n. In addition, it is promised that for every player i, there exists a set consisting of $(1-2\theta)$-fraction of the copies j, such that the bit of party i for copy j is the majority bit of copy j (that is, $(1-2\theta)$-fraction of the bits of each party are "correct").

Our proof consists of two main parts. We first show that any protocol for information dissemination in our noisy model implies a protocol for GapMaj_n^m with the same communication cost. Here and for the rest of the section we set $m = \Theta(n)$. As explained in Section 1.1, this means that we can convert the noise in the channel to a type of noise in the inputs. We then prove an $\Omega(n^2)$ lower bound on the communication cost of GapMaj_n^m.

2.1 Reducing Noiseless GapMaj_n^m to Noisy Dissemination

The reduction for simple protocols

We first explain why "simple" information dissemination protocols, structured like the simple dissemination protocol we described in Section 1 (also see Section 4), imply a protocol for GapMaj_n^n with a similar number of rounds. Later in the section, we show how to extend the reduction to general dissemination protocols. Consider a protocol Π where all parties broadcast the same number of times. Additionally, assume that the protocol Π consists of two phases: in the first phase, which is, say, the first half of the rounds, the parties take turns broadcasting their input bits. Then, in the second phase, consisting of the second half of the rounds, the messages broadcast by the parties are only functions of their received transcript and are independent of their private input (*i.e.*, players *"forget"* their inputs).

To best see the connection to GapMaj_n^n, consider such a two-phase protocol Π in the following weak noise model. In this model, the adversary is θ-limited, and, in addition, it is only allowed to corrupt the messages received in the first phase, while the messages broadcast in the second phase are always received correctly. Furthermore, the only type of corruptions allowed in the first phase are as follows: for every two parties i and j, party i receives only 0s from party j or receives only 1s for party j. Since party j only broadcasts their input bit, this means that either party i receives all their transmissions correctly (j's input is $b \in \{0,1\}$ and i received only b bits), or party i received all the transmissions flipped (party j's input is b and i received only $\bar{b}$ bits). Clearly, lower bounds in this weaker noise model imply similar lower bounds for our noise model.

Next, observe that in the first phase of Π, for every party i, at least $(1-2\theta)$-fraction of the parties received (many repetitions of) the correct input of i. This is because the adversary can only corrupt θ-fraction of the total outgoing messages of party i, since party i broadcasts in $\frac{1}{n}$ faction of the rounds, and since the first phase is half of the total communication. Similarly, since the adversary can only corrupt a θ-fraction of the total incoming messages of a party, every party i receives the correct input of at least $(1-2\theta)$-fraction of the parties.

Next, we claim that in the second, noiseless, phase, the parties are left with solving an instance of GapMaj_n^n in the absence of noise. In this instance, the input of party i for the j-th copy of GapMaj_n is the input that player i received from player j in the first phase. The promise in the definition of GapMaj_n^n is indeed satisfied: for every j, at least $(1-2\theta)$-fraction of the parties i have the majority bit as their input for copy j, and for every i, for at least $(1-2\theta)$-fraction of the j's, party i's input for copy j is the true majority bit of the j-th copy.

Removing the assumption of same number of broadcasts

So far we have considered "simple" protocols. We next show how to handle general protocols. First, we wish to remove the assumption that each party broadcasts in the same number of rounds. Note that this assumption is needed for the above argument. For example, if party i broadcasts in all the rounds of the second phase, then since the adversary is allowed to corrupt a θ-fraction of the *total* received communication for rounds where this party broadcasts, the adversary can corrupt all the messages received from this party in the first phase.

To rectify this situation, we "reveal" the input of all parties that broadcast in at least $\frac{3}{n}$-fraction of the rounds (this is 3 times the average communication). Note that since we are working in the non-adaptive model, the number of times that each party broadcasts is determined ahead of time. By Markov's inequality, at least $\frac{2n}{3}$ parties speak in less than $\frac{3}{n}$-fraction of the rounds and are not fixed by being revealed. Therefore, we end up with a dissemination protocol that only needs to disseminate the input bits of $m \geq \frac{2n}{3}$ parties to all n parties. Note that since our reduction converts the dissemination of the input of one of the parties to one copy of GapMaj_n, applying the reduction to disseminate the value of m parties results in an instance of GapMaj_n^m (instead of GapMaj_n^n).

Removing the rest of the assumptions

The other assumptions we made when considering simple protocols, were that the protocol had two phases of equal lengths. In the first phase, parties only broadcast their inputs, and then, in the second phase, they "forget" their inputs, meaning that the messages broadcast by a party are independent of their input.

To handle general protocols Π, we use the following clever observation used by [32] to analyze protocols under stochastic noise: a message sent by player i in round t of Π given that their received transcript for the $t-1$ first rounds of Π is π, can be deduced from the following three pieces of information:

1. the input bit of party i,
2. the message that party i would have sent had their input been a 0 (and their received transcript was π), and
3. the message that party i would have sent had their input been a 1 (and their received transcript was π).

This gives a way of converting any protocol Π to a two-phase protocol Π' of our desired structure: for rounds $t = 1, 2, \ldots$, if player i broadcasts in round t of Π, add two rounds to the first phase of Π' where player i broadcasts their input. Additionally, add two rounds to the second phase of Π' where in the first, party i sends the message they would have sent had their input been a 0, and in the second they send the message they would have sent had their input been a 1.

2.2 Communication Lower Bound for GapMaj_n^m

Our next goal is to prove a deterministic communication complexity lower bound for GapMaj_n^m, as is promised by Theorem 2. One straightforward approach would be to prove a lower bound for one copy of GapMaj_n and then use one of the known direct sum theorems[6]. While it is not hard to prove a deterministic, or even a non-deterministic, communication lower bound for GapMaj_n, and while a perfect direct sum theorem is known for non-deterministic communication complexity [22, 40], this still does not give us the required bound. The reason is that, as explained after Theorem 2, such direct sum theorems only rule out "strong" communication protocols that solve all the copies that satisfy the promise, whereas we also need to rule out protocols that only output correctly when the promise is satisfied for all copies.

We also mention that the randomized communication complexity of GapMaj_n^m is low, at least when constant error probability is allowed. Consider the following protocol: each party broadcasts $t \cdot \frac{m}{n}$ random bits from their input, for some $t \leq n$. So, the expected number of bits communicated per copy is t, implying a total communication of tm and, using the Chernoff bound, the success probability of the protocol is at least $1 - 2^{-\frac{t}{10}}$. By taking $t = \frac{n}{10}$, we get a protocol with $\frac{mn}{10}$ communication and success probability $1 - 2^{-\frac{n}{100}}$.

We prove a randomized lower bound, showing that the latter tradeoff is optimal up to constant factors in the exponent. Specifically, we show that the success probability cannot be as high as $1 - 2^{-100n}$. Note that this is stronger than the deterministic lower bound we need for the proof of Theorem 1 to go through.

Our hard distribution(s)

To show our randomized lower bound, we prove a distributional lower bound and use Yao's minimax theorem. Consider the following distribution $\mathcal{D}$ on inputs for GapMaj_n^m: for every i and j, party i gets the bit 0 for copy j with probability $1 - \theta'$, independently, where $\theta' = \theta^{200}$.[7] It may seem at first that our distribution is "easy", as the right answer is the all-zeros vector, except with exponentially small probability. However, while exponentially small, the error probability of this protocol is still too large (the error probability is the probability that the promise is satisfied, but the correct answer is not the all-zeros vector).

To show that, observe what happens when we fix the first copy (say) to be 1 for all the players, which is an event whose probability is exponentially small. As the copies are mutually independent, the distribution of the other copies is not affected by this conditioning. Thus, for each one of the remaining copies, they satisfy the promise except with an exponentially small probability. Using a union bound, we get that conditioned on this event, all the copies satisfy the promise except with an exponentially small probability. This means that conditioned on the event that the first copies is fixed to be all-ones, it is likely that the the input is counted in the error probability, implying that the error probability of this protocol

[6] Since most direct sum theorems are for the two-party setting, one would first need to adapt the theorem to the multi-party setting.

[7] With an exponentially small probability, an instance sampled from this distribution does not satisfy the promise in the definition of GapMaj_n^m. If this is the case, we'll accept any output by the protocol.

is *at least* exponentially small (greater than our allowed error probability of 2^{-100n}). Also, observe that, as we union bounded over all the copies, the exact same argument can be made even if the copies are only *pairwise independent* instead of mutually independent.

In fact, the same arguments can be used to show that GapMaj_n^m cannot be solved by a 0-communication protocol over a large set of other distributions. This set contains the following distributions:

1. Distributions where the probability that party i gets the bit 1 for copy j is between $\frac{\theta'}{2}$ and $2\theta'$, and, as in $\mathcal{D}$, all input bits are independent.
2. Distributions where for every i, the input bits of party i are pairwise-independent (while the inputs of different parties continue to be independent).

   GapMaj_n^m cannot be solved over such distributions with 0-communication protocols, because, as in the case of the distribution $\mathcal{D}$, the all-zeros vector is the correct solution except with exponentially small probability (the argument for this fact does not use independence, and therefore still holds). Meanwhile, our above argument for showing that the correct solution is not the all-zeros vector with probability greater than the allowed error probability only relied on pairwise independence, so it still holds.
3. Distributions where the bits for the same player may not be pairwise-independent, but the distribution of each pair of input bits of the same party are close in total variation distance to a distribution that is independent.

Lower bound over $\mathcal{D}$

The arguments above only showed that protocols with 0-communication will not solve GapMaj_n^m when the inputs are sampled from any distribution in the large set of distributions above. However, the lower bound we desire is for protocols with $o(n^2)$ communication. For this, our approach at a high level is to show that if the inputs of the parties are sampled from the distribution $\mathcal{D}$, then after $o(n^2)$ rounds of communication, the distribution of the inputs of the parties conditioned on the observed transcript (with high probability over the transcript), stays inside the set of distributions above. As distributions in the set are hard for 0-communication protocols, it follows that the original distribution is hard for $o(n^2)$ communication protocols.

Let $\mathcal{D}'$ be the distribution $\mathcal{D}$ conditioned on the observed transcript (for a typical transcript that we omit from the notation for the purposes of this sketch). Our goal is to show that $\mathcal{D}'$ is in the set of distributions defined by Items 1–3. This requires showing that, for any player, the marginal distribution for any pair of copies (and also for any single copy) is close to the corresponding marginal in $\mathcal{D}$ in total variation distance. As it turns out to be easier to handle, we actually measure the distance between these marginals in terms of the KL-divergence (*a.k.a*, relative entropy) and move to total variation distance using Pinsker's inequality later in the proof.

Our goal therefore, is to show that for every party and every bit or pair of bits held by this party, the marginal distribution for this bit or pair of bits in $\mathcal{D}$ and in $\mathcal{D}'$ are close in terms of KL-divergence. Unfortunately, it is easily seen that this goal is impossible: consider the protocol where player 1 sends the bit in the first copy (and nothing happens after that). This simple protocol already violates Item 1 as now the marginal of the first party's bit of the first copy in $\mathcal{D}'$ is a point mass. Getting around this impossibility is the next main part of the proof and we do it in two steps.

Revealing information and the use of pairwise independence

First, we show that the chain rule of KL-divergence and the fact that the protocol has $o(n^2)$ communication implies that the number of bits for which Item 1 fails is $o(n^2)$. For instance, the protocol mentioned above satisfies this, as sending any bit requires a bit of communication. With this bound, whenever a bit violates Item 1 (or a pair of bits violates Item 3), we "fix" the concerned bit(s) (*i.e.*, reveal it to the players for free). The knowledge of these bits changes the marginal distribution of the remaining bits, and in particular, may cause more of them to violate Item 1, causing us to fix even more bits. Nonetheless, as explained below, we are able to show using the chain rule for KL-divergence that this iterative procedure of fixing bits will terminate after $o(n^2)$ bits are fixed.

Indeed, as the protocol has $o(n^2)$ communication, we get that the KL-divergence between $\mathcal{D}$ and $\mathcal{D}'$ before any of the bits are fixed is $o(n^2)$. Because we only fix bits or pairs of bits whose marginal distributions have KL-divergence $\Omega(1)$, every time a bit or pair of bits is fixed, the chain rule for KL-divergence implies that the KL-divergence of the distributions $\mathcal{D}$ and $\mathcal{D}'$, when restricted to the unfixed bits, goes down by at least $\Omega(1)$. As the initial KL-divergence is $o(n^2)$, this implies that the total number of fixed bits is $o(n^2)$.

The second step is to ensure that even after $o(n^2)$ bits are fixed, we still have the property that 0-communication protocols cannot compute GapMaj_n^m. To this end, let us carefully examine the argument above. The crux of the argument above was that the same transcript (which is the empty transcript for 0-communication protocols) is generated by two sets of inputs for which the output of gap majority is different. As the output is determined by the transcript, this means that the output of the protocol must be incorrect for at least one of the two sets, giving us the lower bound. Specifically, we observed that a typical input from the distribution $\mathcal{D}$ satisfied the promise of GapMaj_n^m and resulted in the output being the all-zeros vector. Moreover, once we fix one of the copies to be one for all the parties, the remaining copies can still be fixed to satisfy the promise and have the majority value in the copies be zero.

We claim that, except with small probability, we can make this exact argument even after $o(n^2)$ of the bits have been fixed. This is because if the total number of bits fixed is $o(n^2)$, then most of the copies have only $o(n)$ fixed bits. Thus, regardless of the values these bits are fixed to, their number is small enough to not affect the output of GapMaj_n, which is determined by the other bits (with high probability). It remains to consider the copies that have $\Theta(n)$ fixed bits.

For these copies, it is possible that the values of the fixed bits prohibit the promise of gap majority from being satisfied, *e.g.*, when $\frac{n}{2}$ of the bits are fixed to 0 and $\frac{n}{2}$ of the bits are fixed to 1. However, as our distribution $\mathcal{D}$ is heavily biased towards 0 and likely to remain this way for a typical transcript, this is an unlikely event and can be ignored. What cannot be ignored on the other hand is the case where almost all, say 0.999-fraction, of the fixed bits are fixed to 0, and the remaining 0.001-fraction are fixed to 1. Because the number of fixed bits is large, when this happens, we can no longer fix the remaining bits in the copy to satisfy the promise and have the majority value be 1, affecting the second of the two properties we desire.

To tackle this, we recall the fact that the number of copies for which this can happen is small, as most of the copies will have $o(n)$ fixed bits. In our proof, we track these copies with a large number of fixed entries and fix all their remaining bits. This makes sure that any copy with even one unfixed bit, can be fixed to satisfy the promise and have the majority value be 1, while also making sure that the total number of fixed bits stays $o(n^2)$, as we only increase the number of fixed bits by a constant factor. Note that, as before, fixing any of the bits changes the marginal distribution of the remaining bits, and this fixing must therefore be done iteratively until there are no more bits that need to be fixed.

3 Model and Preliminaries

3.1 Concentration Inequalities

▶ **Lemma 3** (Multiplicative Chernoff bound). *Suppose $X_1, \cdots, X_n$ are independent random variables taking values in $[0,1]$. Let X denote their sum and let $\mu = \mathbb{E}[X]$ denote the sum's expected value. Then,*

$$\Pr(X \geq (1+\delta)\mu) \leq \mathrm{e}^{-\frac{\delta^2\mu}{2+\delta}}, \qquad \forall 0 \leq \delta,$$
$$\Pr(X \leq (1-\delta)\mu) \leq \mathrm{e}^{-\frac{\delta^2\mu}{2}}, \qquad \forall 0 \leq \delta \leq 1.$$

In particular, we have that:

$$\Pr(X \geq (1+\delta)\mu) \leq \mathrm{e}^{-\frac{\delta\mu}{3}\cdot\min(\delta,1)}, \qquad \forall 0 \leq \delta,$$
$$\Pr(|X-\mu| \geq \delta\mu) \leq 2\cdot\mathrm{e}^{-\frac{\delta^2\mu}{3}}, \qquad \forall 0 \leq \delta \leq 1.$$

3.2 Error Correcting Codes

We use the following standard result about the existence of error correcting codes.

▶ **Lemma 4.** *Let $\delta > 0$ and define $K_0 = \lceil 10/\delta^2 \rceil$. For all $n > 0$, there exists a function $\mathsf{ECC}_{n,\delta} : \{0,1\}^n \to \{0,1\}^{K_0 n}$ such that for all $s \neq t \in \{0,1\}^n$, we have*

$$\Delta(\mathsf{ECC}_{n,\delta}(s), \mathsf{ECC}_{n,\delta}(t)) > \left(\frac{1}{2} - \delta\right) \cdot K_0 n.$$

3.3 The Adversarial Broadcast Channel

Protocols

Our communication model is the multi-party adversarial broadcast channel. Throughout this paper, we use n to denote the number of parties. An n-party protocol in this model is defined by a tuple:

$$\Pi = \left(\left\{\mathcal{X}^{(i)}\right\}_{i\in[n]}, \mathcal{Y}, T, \sigma, \{M_j\}_{j\in[T]}, \mathsf{out}\right),$$

where

1. for all $i \in [n]$, $\mathcal{X}^{(i)}$ is set of inputs of party i. We also define $\mathcal{X} = \mathcal{X}^{(1)} \times \cdots \times \mathcal{X}^{(n)}$.
2. $\mathcal{Y}$ is set of outputs of a protocol.
3. $T = \|\Pi\| \in \mathbb{N}$ is the length (number of rounds) in the protocol.
4. $\sigma \in [n]^T$ is a vector indicating for all rounds $j \in [T]$, which is the (unique) party scheduled to speak in round j.
5. for all $j \in [T]$, $M_j : \mathcal{X}^{\sigma_j} \times \{0,1\}^{j-1} \times \left(\{0,1\}^*\right)^j \to \{0,1\}$ is the message function used by party σ_j that uses his input, the bits he received in the first $j-1$ rounds, and the randomness sampled in the first j rounds to output the bit he will broadcast in round j.
6. $\mathsf{out} : \{0,1\}^T \to \mathcal{Y}$ is the function that the parties use to compute the output of the protocol based on the bits they receive.

We will omit $\mathcal{X}^{(i)}$ and $\mathcal{Y}$ when they are clear from context.

Adversaries

Let Π be a protocol as above. An adversary for Π is defined by a tuple $\mathsf{Adv} = (\mathsf{Adv}_{i,j})_{i\in[n],j\in[T]}$, where $\mathsf{Adv}_{i,j} : \mathcal{X} \times \left(\{0,1\}^*\right)^j \to \{0,1\}$. Here, for all $i \in [n]$ and $j \in [T]$, the function $\mathsf{Adv}_{i,j}$ takes as input the inputs of all the parties and the randomness sampled in the first j rounds and outputs 1 if he wants to flip (corrupt) the bit party i receives in round j, and outputs 0 otherwise. Our formulation thus, captures adversaries that have knowledge of all the parties' inputs and the randomness they sampled so far, but are unaware of the randomness they will sample in the future.

Protocol execution

We next describe the execution of a protocol Π in the presence of adversary Adv: Each party $i \in [n]$ starts with an input $x^{(i)} \in \mathcal{X}^{(i)}$. Let $x = \left(x^{(1)}, \dots, x^{(n)}\right)$. The execution takes place in T rounds, maintaining the invariant that for all parties $i \in [n]$ and all rounds $j \in [T]$, party i has a transcript $\pi^{(i)}_{<j}$ before the execution of round j. In each round $j \in [T]$, the parties first sample a shared random string $r_j \in \{0,1\}^*$. The player σ_j then computes $\pi_j = M_j\left(x^{(\sigma_j)}, \pi^{(\sigma_j)}_{<j}, r_{\leq j}\right)$ and broadcasts it over the channel. All parties $i \in [n]$ then receive a (potentially corrupted) bit $\pi^{(i)}_j = \pi_j \oplus \mathsf{Adv}_{i,j}(x, r_{\leq j})$ and append it to $\pi^{(i)}_{<j}$ to get $\pi^{(i)}_{\leq j}$. At the end of the protocol, all parties $i \in [n]$ output $\mathsf{out}\left(\pi^{(i)}_{\leq T}\right)$. We say that an execution is *noiseless* if $\mathsf{Adv}_{i,j}$ always outputs 0, and we call this adversary the noiseless adversary.

Additional discussion of the model

We finish this section with a few remarks about the above definition: Note that when Π is executed in the presence of Adv, the output of any party $i \in [n]$ is determined by the parties' inputs $x = \left(x^{(1)}, \dots, x^{(n)}\right)$ and the sampled randomness $r_{\leq T}$. Owing to this, we denote it using the notation $\Pi_{\mathsf{Adv},i}(x, r_{\leq T})$. We will omit writing Adv, i when the adversary is noiseless as in this case, all players compute the same transcripts, and therefore, also the same output. We also omit $r_{\leq T}$ from our notation when talking about deterministic protocols. Also, as mentioned above, we define our adversaries to have complete knowledge of the inputs of the parties and the randomness they sampled so far but they are not aware of any future randomness the parties might have. Due to their knowledge of the randomness sampled so far (and the inputs), the adversaries can also compute all the bits sent and received over the channel so far, and we do not explicitly include this in our notation. Moreover, as our upper bound is deterministic, we will only need this assumption in our lower bound, and it only makes our result stronger.

Next, note that, as defined, the output function out is the same for all parties and depends only on the transcript of the protocol and not on the inputs of the parties. This is without loss of generality, as we can always extend the protocol so that one of the parties computes and sends the output over the channel (using an error correcting code) and all the parties then decode it from the transcript. Finally, note that our definition allows us to easily measure the number of bits corrupted by the adversary. As we are interested only in adversaries that do not corrupt too many bits sent or received by any given party, we define, for all $i \in [n]$, the set $\sigma^{-1}(i) = \{j \in [T] : \sigma(j) = i\}$ to contain rounds where party i broadcast and also define:

▶ **Definition 5.** *Let* Π *be a protocol and* Adv *be an adversary for* Π*. Let* $\theta > 0$*. We say that* Adv *is* θ*-limited, if for all* $x = \left(x^{(1)}, \ldots, x^{(n)}\right)$*, all* $r_{\leq T}$*, and all* $i \in [n]$*, we have:*

$$\sum_{j \in [T]} \mathsf{Adv}_{i,j}(x, r_{\leq j}) \leq \theta T,$$

$$\sum_{j \in \sigma^{-1}(i)} \sum_{i' \in [n]} \mathsf{Adv}_{i',j}(x, r_{\leq j}) \leq \theta n \cdot \left|\sigma^{-1}(i)\right|.$$

Computation over the model

Let Π be a protocol as above and $f : \mathcal{X} \to \mathcal{Y}$ be a (possibly partial) function. Let $\theta, p > 0$. We say that the protocol Π computes f with probability p resilient to θ-adversarial noise if for all inputs $x = \left(x^{(1)}, \ldots, x^{(n)}\right)$ in the domain of f and all θ-limited adversaries, we have:

$$\Pr(\forall i \in [n] : \Pi_{\mathsf{Adv},i}(x, r_{\leq T}) = f(x)) \geq p. \tag{1}$$

We omit writing "resilient to" when $\theta = 0$. As there is only one adversary that is 0-limited, we will also omit Adv from the subscript in this case.

The ID and GapMaj problems

We consider the n-party *Information Dissemination* function, denoted ID_n, that simply outputs its n-sized tuple of arguments. That is, $\mathsf{ID}_n : \{0,1\}^n \to \{0,1\}^n$, where $\mathsf{ID}_n(x_1, \ldots, x_n) = (x_1, \ldots, x_n)$.

We also define the partial function $\mathsf{GapMaj}_{\epsilon,n}^{\delta,m}$ that is parametrized by numbers $\epsilon, \delta > 0$ and an integer m and is such that $\mathcal{X}^{(i)} = \mathcal{Y} = \{0,1\}^m$ for all $i \in [n]$. $\mathsf{GapMaj}_{\epsilon,n}^{\delta,m}$ is defined only if there exists a $\hat{x} \in \{0,1\}^m$ such that

1. for all $i \in [n]$, the Hamming distance between $\hat{x}$ and x_i (the input vector for party i) is at most δm,
2. for all $j \in [m]$, we have $\hat{x}_j \neq x_j^{(i)}$ for at most ϵn values of $i \in [n]$,

and outputs the (unique, for small ϵ) vector $\hat{x}$. For notational convenience, we will interpret $\mathsf{GapMaj}_{\epsilon,n}^{\delta,m}$ as outputting a set of possible values, where the set is the singleton set $\{\hat{x}\}$ if the conditions above are satisfied, and is $\{0,1\}^m$ otherwise.

3.4 Our Result

We are now ready to state the formal version of Theorem 1.

▶ **Theorem 6.** *For all* $\theta > 0$*, there exists* $\kappa > 0$ *such that for all integers* n *large enough, any protocol* Π *that computes* ID_n *with probability* 1 *resilient to* θ*-adversarial noise has length* $\|\Pi\| > \kappa n^2$*.*

We will actually prove the following slightly stronger Theorem 7 that implies Theorem 6:

▶ **Theorem 7.** *For all* $\theta > 0$*, there exists* $\kappa > 0$ *such that for all integers* n *large enough, any protocol* Π *that computes* ID_n *with probability* $1 - \kappa^n$ *resilient to* θ*-adversarial noise has length* $\|\Pi\| > \kappa n^2$*.*

The proof of Theorem 7 has two main parts. The first part is a reduction showing that lower bounds for protocol computing Information Dissemination can be obtained from lower bounds from protocols computing Gap Majority. This is described in Section 5. The next part is a lower bound for protocols computing Gap Majority, which is written in Section 6.

4 Our Information Dissemination Protocol

In order to demonstrate that our result in Theorem 6 is tight, we provide a simple algorithm with length $\mathcal{O}(n^2)$, which is resilient to θ-adversarial noise, for all $\theta < 1/40$.

High level description

To summarize, the idea is to proceed in two phases. In the first phase, every player says their input bit $\mathcal{O}(n)$ times. Each other player then takes the majority value of what they heard. Thus, each player now has a guess for each player's input. Then, in the second phase, every player encodes the string of guesses they have using an error-correcting code, and broadcasts the result. Each player then takes all the error-correcitng codes they've received, decodes them, and sets their guess for each player's input to be the majority value among the guesses they've decoded.

The reason this works is that in order to corrupt a player's output, the adversary needs to either corrupt a lot of the error-correcting codes received by that player, or they need to corrupt a lot of players' guesses about some specific party's input. In either case, the adversary ends up corrupting too many rounds of communication, thus showing that a θ-limited adversary cannot possibly corrupt even a single player's output, exactly as desired.

The formal protocol

Our protocol is given in Algorithm 1. It uses an error correcting code $\mathsf{ECC}_{n,\delta}$ as promised by Lemma 4, where $\delta = 1/10$. We use K_0 as given in that lemma.

Algorithm 1 The algorithm computing ID_n.

```
**Input:** Each party $k \in [n]$ has an input $x_k \in \{0,1\}$.
**Output:** Each party $i \in [n]$ outputs a $\hat{x}_{i,1}, \ldots, \hat{x}_{i,n}$, such that $\hat{x}_{i,k} = x_k$ for all $k \in [n]$.
for $k \in [n]$ do
    Party $k$ broadcasts $x_k$ $K_0 n$ times.
    Each party $j \in [n]$ sets $y_{j,k}$ equal to the majority value they received in the previous $K_0 n$ broadcasts.
end for
for $j \in [n]$ do
    Party $j$ computes and broadcasts $\mathsf{ECC}_{n,\delta}(y_{j,1}, \ldots, y_{j,n})$. This takes $K_0 n$ broadcasts.
    Each party $i \in [n]$ sets $\hat{y}_{i,j,1}, \ldots, \hat{y}_{i,j,n}$ to the minimize the distance of $\mathsf{ECC}_{n,\delta}(\hat{y}_{i,j,1}, \ldots, \hat{y}_{i,j,n})$ to the messages they received in the previous $K_0 n$ broadcasts.
end for
Each party $i \in [n]$ sets $\hat{x}_{i,k}$ to be the majority value between $\hat{y}_{i,1,k}, \ldots, \hat{y}_{i,n,k}$ for all $k \in [n]$.
Each party $i \in [n]$ outputs $\hat{x}_{i,1}, \ldots, \hat{x}_{i,n}$.
```

▶ **Theorem 8.** *For all $\theta < 1/40$, the protocol in Algorithm 1 solves ID_n resilient to θ-adversarial noise.*

Proof. We begin by noting that this protocol is deterministic. As such, the behavior of the protocol is completely determined by the inputs to the players and the adversary Adv, and that, for simplicity, adversaries can be assumed to just be a function of the inputs $x_1, \ldots, x_n$.

Fix some adversary Adv. We wish to demonstrate that if there exists some input $x_1, \ldots, x_n$ such that executing the protocol in Algorithm 1 against Adv on inputs $x_1, \ldots, x_n$ results in a player outputting an incorrect output, then the adversary Adv is not θ-limited.

To see this, suppose that a player outputs some incorrect value. In particular, suppose that $\hat{x}_{i,k} \neq x_k$ for some $i \in [n]$ and $k \in [n]$. That means that the majority value among $\hat{y}_{i,1,k}, \ldots, \hat{y}_{i,n,k}$ was not x_k.

That implies that there must exist some set $S \subseteq [n]$ such that $|S| \geq n/4$, and that one of the two following conditions must hold true:

1. For all $j \in S$, $\hat{y}_{i,j,k} \neq y_{j,k}$, or
2. for all $j \in S$, $y_{j,k} \neq x_k$.

We claim that in order for either of these cases to occur, the adversary Adv must not be θ-limited. First suppose that for all $j \in S$, $\hat{y}_{i,j,k} \neq y_{j,k}$. That implies that for each $j \in S$, during iteration j of the loop at Line 5, at least $0.2K_0 n$ of the bits sent by player j are received incorrectly by player i, by the properties promised about $\mathsf{ECC}_{n,\delta}$ in Lemma 4. That implies that over all the iterations of the loop at Line 5, player i hears at least $\frac{0.2}{4}K_0 n^2 = \frac{1}{20}K_0 n^2$ messages incorrectly. That means that at least $1/40$ of all $2K_0 n^2$ messages sent during the protocol are misheard by player i. This, thus, shows that Adv is not θ-limited.

On the other hand, suppose that for all $j \in S$, $y_{j,k} \neq x_k$. That implies that for all $j \in S$, during iteration k of the loop at Line 1, at least $0.5K_0 n$ of the bits sent by player k are received incorrectly by player j. That implies that there are at least $\frac{0.5}{4}K_0 n^2 = 1/8 K_0 n^2$ corruptions in messages from player k to other players, of a total of $2K_0 n^2$ messages received in rounds during which this player broadcasts. This, thus, shows that Adv is not θ-limited. ◀

5 Reducing Gap Majority to Information Dissemination

This section has the first part of our proof, which is a reduction from Gap Majority to Information Dissemination. Specifically, our reduction shows that noise resilient protocols for Information Dissemination imply noiseless protocols for Gap Majority, as formalized next:

▶ **Theorem 9.** *Let parameters $0 < \theta < \frac{1}{3}$, $0 < p < 1$ and $n \in \mathbb{N}$ be given. If there exists a protocol Π computing ID_n with probability p resilient to θ-adversarial noise, there exists another protocol Π' computing $\mathsf{GapMaj}_{\theta,n}^{\theta,\theta n}$ with probability p such that $\|\Pi'\| \leq 2 \cdot \|\Pi\| + n$.*

Proof. Fix θ, p, n, and Π as in the theorem statement. Let $\Pi = \left(T, \sigma, \{M_j\}_{j \in [T]}, \mathsf{out}\right)$. Define the set $I' = \left\{i \in [n] \mid \left|\sigma^{-1}(i)\right| \leq \frac{T}{\theta n}\right\}$ to be the set of parties that do not broadcast too often. By Markov's inequality, note that $|I'| \geq (1-\theta)n > \theta n$. We let $m = \theta n$, I be the first m elements of I' and assume without loss of generality that $I = [m]$. We are now ready to define the protocol Π'. We note that throughout the description of Π' and its analysis, we will treat vectors in $\{0,1\}^m$ also as vectors in $\{0,1\}^n$ by padding with an appropriate number of zeros.

Algorithm 2 The algorithm Π' computing $\mathsf{GapMaj}^{\theta,m}_{\theta,n}$.

```
Input: Party $i \in [n]$ has an input $x'^{(i)} \in \{0,1\}^m$.
1: Each party $i \in [n]$ sets $\pi'^{(i)} \leftarrow \varepsilon$, the empty string.
2: for $j \in [T]$ do
3:     The parties together sample a random string $r_j \in \{0,1\}^*$.
4:     Party $\sigma_j$ sets $\tau_{j,b} \leftarrow M_j\big(b, \pi'^{(\sigma_j)}, r_{\leq j}\big)$ for all $b \in \{0,1\}$.    ▷ $\big|\pi'^{(\sigma_j)}\big| = j-1$.
5:     Party $\sigma_j$ broadcasts $\tau_{j,b}$ for all $b \in \{0,1\}$ to all other players.
6:     Each party $i \in [n]$ extends $\pi'^{(i)}$ by appending $\tau_{j,x'^{(i)}_{\sigma_j}}$.
7: end for
8: Each party $i \in [n]$ outputs the first $m$ bits of $\mathsf{out}\big(\pi'^{(i)}\big)$.
```

Observe that, as written, the output function of the protocol Π' depends on the inputs of the parties. However, as mentioned in Section 3, this can be easily corrected by adding an extra n rounds where one of the parties broadcasts its output over the channel. Together with these n rounds, the $2T$ rounds in Line 5 imply that $\|\Pi'\| \leq 2 \cdot \|\Pi\| + n$. It remains to show that Π' computes $\mathsf{GapMaj}^{\theta,\theta n}_{\theta,n}$ with probability p. For this, we have to show Equation (1). We do this next.

Fix $x' = \big(x'^{(1)}, \dots, x'^{(n)}\big)$ in the domain of $\mathsf{GapMaj}^{\theta,m}_{\theta,n}$ as in Equation (1) and define $\hat{x} = \mathsf{GapMaj}^{\theta,m}_{\theta,n}(x')$. As $\mathsf{ID}_n(\hat{x}) = \hat{x}$ by definition, Equation (1) follows if we show a θ-limited adversary Adv for Π such that for all $r_{\leq T}$ and all $i \in [n]$, we have that:

$$\Pi_{\mathsf{Adv},i}(\hat{x}, r_{\leq T}) = \Pi'_i(x', r_{\leq T}),$$

where, as usual, we pad the output of Π' with zeros to be of length n. Indeed, the above implies $\Pi_{\mathsf{Adv},i}(\hat{x}, r_{\leq T}) = \hat{x} \iff \Pi'_i(x', r_{\leq T}) = \hat{x}$, and Equation (1) follows from the fact that Π computes ID_n with probability p resilient to θ-adversarial noise.

To start, we first note that it suffices to define a different Adv for every randomness $r_{\leq T}$ as the property we want is determined solely by the value of Adv on the randomness $r_{\leq T}$. Fix an arbitrary $r_{\leq T}$ and note that, as we already fixed x', fixing $r_{\leq T}$ fixes the value of all variables in the execution of Algorithm 2. Henceforth, we abuse notation and use the name of the variable to also denote its fixed value at the end of the protocol. We define the adversary Adv as:

$$\mathsf{Adv}_{i,j}(\hat{x}, r_{\leq j}) = \pi'^{(i)}_j \oplus \tau_{j,\hat{x}_{\sigma_j}},$$

and we set it to 0 everywhere else. We now show why this adversary satisfies $\Pi_{\mathsf{Adv},i}(\hat{x}, r_{\leq T}) = \Pi'_i(x', r_{\leq T})$ for all $i \in [n]$. Due to Line 8, this follows if we show that for all i, the transcript $\pi'^{(i)}$ equals the transcript $\pi^{(i)}$ received by party i when Π is executed in the presence of Adv. As both $\pi'^{(i)}$ and $\pi^{(i)}$ have length T, this follows if we show by induction that, for all $0 \leq j \leq T$, we have $\pi'^{(i)}_{\leq j} = \pi^{(i)}_{\leq j}$. The base case $j = 0$ is straightforward. To prove the statement for $j > 0$, we assume it holds for $j-1$ and prove that $\pi'^{(i)}_j = \pi^{(i)}_j$. We have:

$$\begin{aligned}
\pi^{(i)}_j &= M_j\Big(\hat{x}_{\sigma_j}, \pi^{(\sigma_j)}_{<j}, r_{\leq j}\Big) \oplus \mathsf{Adv}_{i,j}(\hat{x}, r_{\leq j}) \\
&= M_j\Big(\hat{x}_{\sigma_j}, \pi^{(\sigma_j)}_{<j}, r_{\leq j}\Big) \oplus \pi'^{(i)}_j \oplus \tau_{j,\hat{x}_{\sigma_j}} && \text{(Definition of } \mathsf{Adv}\text{)} \\
&= M_j\Big(\hat{x}_{\sigma_j}, \pi'^{(\sigma_j)}_{<j}, r_{\leq j}\Big) \oplus \pi'^{(i)}_j \oplus \tau_{j,\hat{x}_{\sigma_j}} && \text{(Induction hypothesis)} \\
&= \pi'^{(i)}_j. && \text{(Line 4)}
\end{aligned}$$

It remains to show that Adv is θ-limited. For this we show the two inequalities in Definition 5. As Adv is 0 everywhere else, it suffices to show it for the arguments $(\hat{x}, r_{\leq T})$. For the first inequality, we have for all $i \in [n]$ that:

$$\begin{aligned}
\sum_{j\in[T]} \mathsf{Adv}_{i,j}(\hat{x}, r_{\leq j}) &\leq \sum_{j\in[T]} \mathbb{1}\left(x'^{(i)}_{\sigma_j} \neq \hat{x}_{\sigma_j}\right) && \text{(Definition of } \mathsf{Adv} \text{ and Line 6)}\\
&= \sum_{i'=1}^{n} \mathbb{1}\left(x'^{(i)}_{i'} \neq \hat{x}_{i'}\right) \cdot \left|\sigma^{-1}(i')\right| \\
&= \sum_{i'=1}^{m} \mathbb{1}\left(x'^{(i)}_{i'} \neq \hat{x}_{i'}\right) \cdot \left|\sigma^{-1}(i')\right| && \text{(The other coordinates are paddings)}\\
&\leq \sum_{i'=1}^{m} \mathbb{1}\left(x'^{(i)}_{i'} \neq \hat{x}_{i'}\right) \cdot \frac{T}{\theta n} && \text{(Definition of } I\text{)}\\
&\leq \theta T. && \text{(Definition of } m \text{ and } \mathsf{GapMaj}^{\theta,m}_{\theta,n}\text{)}
\end{aligned}$$

For the second inequality, we have for all $i \in [n]$ that:

$$\begin{aligned}
\sum_{j\in\sigma^{-1}(i)} \sum_{i'\in[n]} \mathsf{Adv}_{i',j}(\hat{x}, r_{\leq j}) &\leq \sum_{j\in\sigma^{-1}(i)} \sum_{i'\in[n]} \mathbb{1}\left(x'^{(i')}_{\sigma_j} \neq \hat{x}_{\sigma_j}\right) \\
& && \text{(Definition of } \mathsf{Adv} \text{ and Line 6)}\\
&\leq \sum_{j\in\sigma^{-1}(i)} \theta n && \text{(Definition of } \mathsf{GapMaj}^{\theta,m}_{\theta,n}\text{)}\\
&\leq \theta n \cdot \left|\sigma^{-1}(i)\right|.
\end{aligned}$$

◀

6 Lower Bound for Direct Sum Gap-Majority

The goal of this section is to show Theorem 7. As we already proved Theorem 9, it suffices to show the following result.

▶ **Theorem 10.** *For all $0 < \theta < \frac{1}{3}$, there exists $\kappa > 0$ such that for all $n > 0$ large enough and $m = \theta n$, any (possibly randomized) protocol Π computing $\mathsf{GapMaj}^{\theta,\theta n}_{\theta,n}$ with probability $1 - \kappa^n$ satisfies $\|\Pi\| \geq \kappa n^2$.*

Indeed, Theorem 7 follows easily from Theorems 9 and 10. Moreover, as $\mathsf{GapMaj}^{\theta,\theta n}_{\theta,n}$ is an easier problem than $\mathsf{GapMaj}^{1,\theta n}_{\theta,n}$, Theorem 10 implies the following result about the direct-sum of gap-majority, that may be of independent interest.

▶ **Theorem 11.** *For all $0 < \theta < \frac{1}{3}$, there exists $\kappa > 0$ such that for all $n > 0$ large enough and $m = \theta n$, any (possibly randomized) protocol Π computing $\mathsf{GapMaj}^{1,\theta n}_{\theta,n}$ with probability $1 - \kappa^n$ satisfies $\|\Pi\| \geq \kappa n^2$.*

Henceforth, we focus on proving Theorem 10, whose proof spans this entire section. Fix θ as in the theorem statement and define $\kappa = \theta^{1000}$. Let $n > 0$ be sufficiently large and define a distribution $\mathcal{D}$ over inputs for $\mathsf{GapMaj}^{\theta,m}_{\theta,n}$ as follows: For all players $i \in [n]$ and all $j \in [m]$, the bit $x_{i,j}$ is sampled independently of all other bits and is 1 with probability θ^{25} and 0 with probability $1 - \theta^{25}$. We will show that, any deterministic protocol Π with $\|\Pi\| < \kappa n^2$ satisfies:

$$\Pr_{\mathsf{X}\sim\mathcal{D}}\left(\Pi(\mathsf{X}) \in \mathsf{GapMaj}^{\theta,m}_{\theta,n}(\mathsf{X})\right) \leq 1 - \kappa^n.$$

Theorem 10 then follows from Yao's minimax principle. For brevity sake, we henceforth keep the distribution $\mathcal{D}$ implicit. To show this bound, we shall focus on a testing version of Gap Majority, where the parties are only required to determine whether or not the output is the all zeros vector 0^m. Specifically, let $\mathsf{flag} : \{0,1\}^m \to \{0,1\}$ be the indicator function that outputs 0 if and only if the Hamming weight of its input is at most $\frac{\theta m}{2}$ and 1 otherwise. Next, define the set-valued function $\mathsf{GapMaj}'^m_{\theta,n}$ as follows:

$$\mathsf{GM\text{-}Test}^m_{\theta,n}(x) = \begin{cases} \{0,1\}^m, & \text{if } \exists i \in [n] : \mathsf{flag}(x_i) = 1 \\ \{0^m\}, & \text{else if } \mathsf{GapMaj}^{\theta,m}_{\theta,n}(x) = \{0^m\} \\ \{0,1\}^m \setminus \{0^m\}, & \text{else if } \left|\mathsf{GapMaj}^{\theta,m}_{\theta,n}(x)\right| = 1 \\ \{0,1\}^m, & \text{otherwise} \end{cases}. \tag{2}$$

This definition implies that $\mathsf{GapMaj}^{\theta,m}_{\theta,n}(x) \subseteq \mathsf{GM\text{-}Test}^m_{\theta,n}(x)$ for all x and thus, it suffices to show that any deterministic protocol Π with $\|\Pi\| < \kappa n^2$.

$$\Pr\big(\Pi(\mathsf{X}) \in \mathsf{GM\text{-}Test}^m_{\theta,n}(\mathsf{X})\big) \leq 1 - \kappa^n. \tag{3}$$

Fix a protocol Π as above and let $T = \kappa n^2$ so that $\|\Pi\| < T$ and $T' = T/\theta^{500}$. We first augment Π to get another protocol Π_{aug} that reveals some extra information about the parties' inputs. The protocol Π_{aug} is defined below in Algorithm 3 where we use the symbol $\perp$ to denote a special symbol saying "I skip". Also, for a distribution D on the parties' inputs and a subset $S \subseteq [n] \times [m]$, we use $D_{|S}$ to denote the marginal distribution of D over the coordinates in S. If we are writing a set, say $S = \{(i_1, j_1), (i_2, j_2)\}$, explicitly, we may omit the $\{\}$ and simply write $D_{|(i_1,j_1),(i_2,j_2)}$.

Algorithm 3 The protocol Π_{aug}. All lines except Lines 8 and 11 executed by all the players. Any message sent is automatically appended to π_{aug}.

```
Input: Player $i$'s input is a vector $x_i \in \{0,1\}^m$.
 1: Run $\Pi$ to get a transcript $\pi \in \{0,1\}^T$. Set $\pi_{\mathsf{aug}} \leftarrow \pi$.
 2: All players $i \in [n]$ speak. Player $i$ sends $\mathsf{flag}(x_i)$.
 3: For all $i \in [n]$, $j \in [m]$, we set $\mathsf{R}_{i,j} \leftarrow 0$.
 4: for $t \in [T']$ do
 5:     Compute the sets:
            $S_{\mathsf{cell}} = \big\{(i,j) \mid \mathsf{R}_{i,j} = 0 \wedge \mathbb{D}\big((\mathcal{D} \mid \pi_{\mathsf{aug}})_{|(i,j)} \mid\mid \mathcal{D}_{|(i,j)}\big) \geq \theta^{200}\big\}$,
            $S_{\mathsf{pair}} = \big\{(i,j,j') \mid j \neq j' \wedge \mathsf{R}_{i,j} = \mathsf{R}_{i,j'} = 0 \wedge \mathbb{D}\big((\mathcal{D} \mid \pi_{\mathsf{aug}})_{|(i,j),(i,j')} \mid\mid \mathcal{D}_{|(i,j),(i,j')}\big) \geq \theta^{100}\big\}$,
            $S_{\mathsf{col}} = \big\{(i,j) \mid \mathsf{R}_{i,j} = 0 \wedge \big|\big\{i' \in [n] \mid \mathsf{R}_{i',j} = 1\big\}\big| \geq \theta^{100} \cdot n\big\}$.
 6:     if $S_{\mathsf{cell}} \cup S_{\mathsf{col}} \neq \emptyset$ then
 7:         Let $(i_1, j_1)$ be the smallest element in $S_{\mathsf{cell}} \cup S_{\mathsf{col}}$. Set $\mathsf{R}_{i_1,j_1} \leftarrow 1$.
 8:         All players $i \in [n]$ speak. If $i \neq i_1$, they send $(\perp, \perp)$. Else, they send $(x_{i_1,j_1}, \perp)$.
 9:     else if $S_{\mathsf{pair}} \neq \emptyset$ then
10:         Let $(i_2, j_2, j_2')$ be the smallest element in $S_{\mathsf{pair}}$. Set $\mathsf{R}_{i_2,j_2}, \mathsf{R}_{i_2,j_2'} \leftarrow 1$.
11:         All players $i \in [n]$ speak. If $i \neq i_2$, they send $(\perp, \perp)$. Else, they send $\big(x_{i_2,j_2}, x_{i_2,j_2'}\big)$.
12:     else
13:         All players send $(\perp, \perp)$.
14:     end if
15: end for
```

Intuitively, the protocol Π_{aug} "cleans" the protocol Π by revealing the functions $\mathsf{flag}(\cdot)$ and then, iteratively revealing all coordinates, pairs of coordinates, *etc.* for which the marginal distribution changed significantly. We describe this formally in the following section, but before that, a word on the notations we use.

Throughout this proof, we will use sans-serif letters, e.g., X to denote random variables and the corresponding lower case letters, e.g., x to denote their values. When it is clear from context, we may abbreviate the event $\mathsf{X} = x$ as just x. Note that the protocol Π_{aug} is deterministic and the only randomness we have is the randomness of the distribution $\mathcal{D}$ of inputs. All our random variables and probabilities are defined over this randomness.

For a variable *var* in Algorithm 3 and $t \in [T']$, we use var^t to denote the random variable (over the randomness of the inputs in $\mathcal{D}$) whose value equals the value of the variable at the end of iteration t of the loop in Line 4. When $t = 0$, we mean the corresponding value at the beginning of the loop, *i.e.*, after Line 3. We may omit writing the subscript when $t = T'$. We also define the additional set-valued variable $\mathsf{R} = \{(i, j) \mid \mathsf{R}_{i,j} = 1\}$ and use the same notation. Note that the set R can only grow.

6.1 Properties of Π_{aug}

In this subsection, we establish some useful properties of Π_{aug}.

▶ **Lemma 12.** *For all $0 \leq t \leq T'$, the value of π_{aug}^{t-1} determines*[8] *the values of* $\mathsf{R}^0, \mathsf{R}^1, \ldots, \mathsf{R}^t$.

Proof. Proof by induction on t. The base case $t = 0$ is because $\mathsf{R}^0 = \emptyset$ by definition. We prove the lemma for $t > 0$ assuming it holds for $t - 1$. Consider iteration t for the loop in Line 4 and let π_{aug}^{t-1} be an arbitrary value of the variable π_{aug} at the beginning of the iteration. As π_{aug}^{t-1} determines π_{aug}^{t-2}, we have by the induction hypothesis that it also determines the values of $\mathsf{R}^0, \mathsf{R}^1, \ldots, \mathsf{R}^{t-1}$. In particular, it determines R^{t-1}, the value of R at the beginning of this execution. Therefore, it also determines the value of the sets computed in Line 5. Now, using Lines 6, 8, 9, and 11, we get that it also determines the value of R^t, as desired. ◀

Observe from Algorithm 3 that that the variables S_{cell}^t, S_{pair}^t, S_{col}^t, (i_1^t, j_1^t), $(i_2^t, j_2^t, j_2'^t)$ are all determined by π_{aug}^{t-1} and R^{t-1}. Thus, we get:

▶ **Corollary 13.** *For all $0 \leq t \leq T'$, the value of π_{aug}^{t-1} determines the values of S_{cell}^t, S_{pair}^t, S_{col}^t, (i_1^t, j_1^t), (i_2^t, j_2^t, j_2^t).*

▶ **Lemma 14.** *For all $0 \leq t \leq T'$, the value of π_{aug}^t determines the values of $x_{i,j}$ for all $(i, j) \in \mathsf{R}^t$.*

Proof. Proof by induction on t. The base case $t = 0$ is because $\mathsf{R}^0 = \emptyset$ by definition. We prove the lemma for $t > 0$ assuming it holds for $t - 1$. Consider iteration t for the loop in Line 4. As π_{aug}^t determines π_{aug}^{t-1}, we have by the induction hypothesis that π_{aug}^t determines the values of $x_{i,j}$ for all $(i, j) \in \mathsf{R}^{t-1}$. Moreover, from Lines 8 and 11, we have that, for all $(i, j) \in \mathsf{R}^t \setminus \mathsf{R}^{t-1}$, the value of $x_{i,j}$ is determined by π_{aug}^t. ◀

▶ **Lemma 15.** *For all $1 < t \leq T'$, if $S_{\mathsf{cell}}^{t-1} = S_{\mathsf{pair}}^{t-1} = S_{\mathsf{col}}^{t-1} = \emptyset$, then $S_{\mathsf{cell}}^t = S_{\mathsf{pair}}^t = S_{\mathsf{col}}^t = \emptyset$ (with probability 1).*

[8] We define π_{aug}^{t-1} to be some dummy value when $t = 0$.

Proof. Recall from Corollary 13 that, for all $1 < t \le T'$, the value of π_{aug}^{t-2} determines the values of S_{cell}^{t-1}, S_{pair}^{t-1}, S_{col}^{t-1}. Fix an arbitrary $1 < t \le T'$ and an arbitrary π_{aug}^{t-2} such that $S_{\mathsf{cell}}^{t-1} = S_{\mathsf{pair}}^{t-1} = S_{\mathsf{col}}^{t-1} = \emptyset$ and consider iteration $t-1$ of the loop in Algorithm 3. As Lines 8 and 11 are never executed in this iteration we get that $\mathsf{R}^{t-2} = \mathsf{R}^{t-1}$ and that π_{aug}^{t-2} determines π_{aug}^{t-1}. This means that both π_{aug}^{t-2} and π_{aug}^{t-1} determine each other implying that $\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-2} = \mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}$. Combine this with $\mathsf{R}^{t-2} = \mathsf{R}^{t-1}$ and use Line 5 to finish the proof. ◀

▶ **Lemma 16.** *For all $0 \le j \le \left|\pi_{\mathsf{aug}}^{T'}\right|$, the random variables $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are mutually independent conditioned on $\pi_{\mathsf{aug},\le j}^{T'}$.*

Proof. Proof by induction on j. The base case $j = 0$ is trivial. We prove the lemma for $j > 0$ assuming it holds for $j - 1$. By the induction hypothesis, we have that $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are mutually independent conditioned on $\pi_{\mathsf{aug},<j}^{T'}$. This means that for all $i \in [n]$, and all functions f and all values z in the range of f, we have that $\mathsf{X}_1, \ldots, \mathsf{X}_n$ are mutually independent conditioned on $\pi_{\mathsf{aug},<j}^{T'}, f(\mathsf{X}_i) = z$. As conditioned on $\pi_{\mathsf{aug},<j}^{T'}$, the value of $\pi_{\mathsf{aug},j}^{T'}$ is just a function of exactly one of $\mathsf{X}_1, \ldots, \mathsf{X}_n$, the lemma follows. ◀

▶ **Lemma 17.** *We have:*

$$\mathbb{E}\left[\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\right)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\right)\right] = \mathbb{I}\left(\mathsf{X} : \Pi_{\mathsf{aug}}^{0}\right) \le T + n \le 2T.$$

Proof. The inequality follows from Fact 35 and Lemma 33. We now show the equality. As $\mathsf{R}^0 = \emptyset$ and π_{aug}^{0} is just π appended with the values $(\mathsf{flag}(x_i))_{i\in[n]}$, we have:

$$\begin{aligned}
\mathbb{E}\left[\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\right)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\right)\right] &= \mathbb{E}\left[\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\right) \mid\mid \mathcal{D}\right)\right] \\
&= \sum_{\pi_{\mathsf{aug}}^{0}} \sum_{x} \Pr\left(\pi_{\mathsf{aug}}^{0}\right) \cdot \Pr\left(x \mid \pi_{\mathsf{aug}}^{0}\right) \cdot \log \frac{\Pr\left(x \mid \pi_{\mathsf{aug}}^{0}\right)}{\Pr(x)} \\
&\qquad \text{(Definition 37)} \\
&= \mathbb{I}\left(\mathsf{X} : \Pi_{\mathsf{aug}}^{0}\right). \qquad \text{(Lemma 36)}
\end{aligned}$$

◀

Recall from Lemma 12 and Corollary 13 that fixing π_{aug}^{t-1} fixes the values of many variables in Algorithm 3. We now show:

▶ **Lemma 18.** *For all $t \in [T']$ and all π_{aug}^{t-1}, we have:*

$$\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\overline{\mathsf{R}^{t-1}}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^{t-1}}}\right) = \mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\right) + \mathbb{E}\left[\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t}\right)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}}\right) \mid \pi_{\mathsf{aug}}^{t-1}\right].$$

Proof. This essentially is just from the chain rule for KL-divergence. We give the details below. Note that:

$$\begin{aligned}
&\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\overline{\mathsf{R}^{t-1}}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^{t-1}}}\right) \\
&\qquad = \sum_{x_{|\overline{\mathsf{R}^{t-1}}}} \Pr\left(x_{|\overline{\mathsf{R}^{t-1}}} \mid \pi_{\mathsf{aug}}^{t-1}\right) \cdot \log \frac{\Pr\left(x_{|\overline{\mathsf{R}^{t-1}}} \mid \pi_{\mathsf{aug}}^{t-1}\right)}{\Pr\left(x_{|\overline{\mathsf{R}^{t-1}}}\right)} \qquad \text{(Definition 37)} \\
&\qquad = \sum_{x_{|\overline{\mathsf{R}^{t-1}}}} \Pr\left(x_{|\overline{\mathsf{R}^{t-1}}} \mid \pi_{\mathsf{aug}}^{t-1}\right) \cdot \log \frac{\Pr\left(x_{|\overline{\mathsf{R}^t}} \mid x_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}, \pi_{\mathsf{aug}}^{t-1}\right) \Pr\left(x_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid \pi_{\mathsf{aug}}^{t-1}\right)}{\Pr\left(x_{|\overline{\mathsf{R}^t}} \mid x_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\right) \Pr\left(x_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\right)}
\end{aligned}$$

$$\begin{aligned}
&= \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}} \Big) \\
&\quad + \sum_{x_{|\overline{\mathsf{R}^{t-1}}}} \Pr\Big(x_{|\overline{\mathsf{R}^{t-1}}} \mid \pi^{t-1}_{\mathsf{aug}}\Big)\cdot \log \frac{\Pr\Big(x_{|\overline{\mathsf{R}^t}} \mid x_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}}, \pi^{t-1}_{\mathsf{aug}}\Big)}{\Pr\Big(x_{|\overline{\mathsf{R}^t}} \mid x_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}}\Big)}. \qquad \text{(Definition 37)}
\end{aligned}$$

To continue, recall that all the coordinates of all players are mutually independent in the distribution $\mathcal{D}$. Moreover, we have from Lines 6, 8, 9, and 11 that conditioned on π^{t-1}_{aug}, the event $x_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}}$ is the same as the corresponding event π^{t}_{aug}. We get:

$$\begin{aligned}
&\mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^{t-1}}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^{t-1}}} \Big) \\
&= \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}} \Big) + \sum_{x_{|\overline{\mathsf{R}^t}}} \sum_{\pi^t_{\mathsf{aug}}} \Pr\Big(x_{|\overline{\mathsf{R}^t}}, \pi^t_{\mathsf{aug}} \mid \pi^{t-1}_{\mathsf{aug}}\Big) \cdot \log \frac{\Pr\Big(x_{|\overline{\mathsf{R}^t}} \mid \pi^t_{\mathsf{aug}}\Big)}{\Pr\Big(x_{|\overline{\mathsf{R}^t}}\Big)} \\
&= \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}} \Big) + \sum_{\pi^t_{\mathsf{aug}}} \Pr\big(\pi^t_{\mathsf{aug}} \mid \pi^{t-1}_{\mathsf{aug}}\big) \cdot \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big) \\
&\qquad\qquad \text{(Definition 37)} \\
&= \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}} \Big) + \mathbb{E}\Big[\mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big) \mid \pi^{t-1}_{\mathsf{aug}} \Big].
\end{aligned}$$

◀

▶ **Lemma 19.** *Let $0 \le t \le T'$ and π^0_{aug} be given. Let P^t be any set of transcripts π^t_{aug} that all have π^0_{aug} as a prefix. It holds that:*

$$\sum_{\pi^t_{\mathsf{aug}} \in P^t} \Pr\big(\pi^t_{\mathsf{aug}} \mid \pi^0_{\mathsf{aug}}\big) \cdot \nabla\big(\pi^t_{\mathsf{aug}}\big) \le \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{0}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}} \Big),$$

where:

$$\nabla\big(\pi^t_{\mathsf{aug}}\big) = \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big) + \sum_{t'=1}^{t} \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t'-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^{t'}\setminus\mathsf{R}^{t'-1}} \Big).$$

Proof. We prove the lemma by induction on t. The base case $t = 0$ is straightforward. We prove the lemma for $t > 0$ assuming it holds for $t - 1$. Fix a set P^t as in the lemma statement and define, for all $0 \le t' < t$, the set $P^{t'}$ to be the set of all $\pi^{t'}_{\mathsf{aug}}$ that are prefixes of a $\pi^t_{\mathsf{aug}} \in P^t$. We have:

$$\begin{aligned}
&\sum_{\pi^t_{\mathsf{aug}} \in P^t} \Pr\big(\pi^t_{\mathsf{aug}} \mid \pi^0_{\mathsf{aug}}\big) \cdot \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big) \\
&\quad = \sum_{\pi^{t-1}_{\mathsf{aug}} \in P^{t-1}} \Pr\big(\pi^{t-1}_{\mathsf{aug}} \mid \pi^0_{\mathsf{aug}}\big) \cdot \mathbb{E}\Big[\mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big) \mid \pi^{t-1}_{\mathsf{aug}} \Big] \\
&\quad = \sum_{\pi^{t-1}_{\mathsf{aug}} \in P^{t-1}} \Pr\big(\pi^{t-1}_{\mathsf{aug}} \mid \pi^0_{\mathsf{aug}}\big) \cdot \\
&\qquad\qquad \Big(\mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^{t-1}}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^{t-1}}} \Big) - \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t-1}_{\mathsf{aug}}\big)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t\setminus\mathsf{R}^{t-1}} \Big) \Big). \\
&\qquad\qquad \text{(Lemma 18)}
\end{aligned}$$

To continue, we use the induction hypothesis on the first term. We get:

$$\sum_{\pi^t_{\mathsf{aug}} \in P^t} \Pr\big(\pi^t_{\mathsf{aug}} \mid \pi^0_{\mathsf{aug}}\big) \cdot \mathbb{D}\Big(\big(\mathcal{D} \mid \pi^{t}_{\mathsf{aug}}\big)_{|\overline{\mathsf{R}^t}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^t}} \Big)$$

$$\begin{aligned}
&\leq \mathbb{D}\Big(\big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\big)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\Big) \\
&\qquad - \sum_{\pi_{\mathsf{aug}}^{t-1} \in P^{t-1}} \Pr\big(\pi_{\mathsf{aug}}^{t-1} \mid \pi_{\mathsf{aug}}^{0}\big) \cdot \sum_{t'=1}^{t} \mathbb{D}\Big(\Big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t'-1}\Big)_{|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}}\Big) \\
&\leq \mathbb{D}\Big(\big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\big)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\Big) \\
&\qquad - \sum_{\pi_{\mathsf{aug}}^{t} \in P^{t}} \Pr\big(\pi_{\mathsf{aug}}^{t} \mid \pi_{\mathsf{aug}}^{0}\big) \cdot \sum_{t'=1}^{t} \mathbb{D}\Big(\Big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t'-1}\Big)_{|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}}\Big).
\end{aligned}$$

Rearranging gives the result. ◀

▶ **Corollary 20.** *Let π_{aug}^{0} be given and $P^{T'}$ be any set of transcripts $\pi_{\mathsf{aug}}^{T'}$ that all have π_{aug}^{0} as a prefix. It holds that:*

$$\sum_{\pi_{\mathsf{aug}}^{T'} \in P^{T'}} \Pr\Big(\pi_{\mathsf{aug}}^{T'} \mid \pi_{\mathsf{aug}}^{0}\Big) \cdot \sum_{t=1}^{T'} \mathbb{D}\Big(\big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\big)_{|\mathsf{R}^{t} \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^{t} \setminus \mathsf{R}^{t-1}}\Big) \leq \mathbb{D}\Big(\big(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\big)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\Big).$$

6.2 Many Clean Transcripts

Recall from Corollary 13 that fixing π_{aug} fixes the values of the values computed in Line 5. By definition, it also fixes the values computed in Line 2. Using this, we define the following events that are just some subsets of all possible $\pi_{\mathsf{aug}}^{T'}$:

1. Define the event $\mathcal{E}_{\mathsf{clean},\mathsf{flag}}$ to be the set of all $\pi_{\mathsf{aug}}^{T'}$ such that for all $i \in [n]$, we have:

   $$\mathsf{flag}(x_i) = 0.$$

2. Define the event $\mathcal{E}_{\mathsf{clean},x}$ to be the set of all $\pi_{\mathsf{aug}}^{T'}$ such that for all $j \in [m]$, we have:

   $$\Pr\left(\sum_{i=1}^{n} x_{i,j} \geq \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) \leq 2^{-\theta^5 n}.$$

3. Define the event $\mathcal{E}_{\mathsf{clean},S}$ to be the set of all $\pi_{\mathsf{aug}}^{T'}$ such that:

   $$S_{\mathsf{cell}}^{T'} = S_{\mathsf{pair}}^{T'} = S_{\mathsf{col}}^{T'} = \emptyset.$$

The goal of this section is to show that a randomly sampled transcript $\pi_{\mathsf{aug}}^{T'}$ is likely to be clean. Namely, if we define $\mathcal{E}_{\mathsf{clean}} = \mathcal{E}_{\mathsf{clean},\mathsf{flag}} \wedge \mathcal{E}_{\mathsf{clean},x} \wedge \mathcal{E}_{\mathsf{clean},S}$, we have:

▶ **Lemma 21.** *It holds that:*

$$\Pr\big(\overline{\mathcal{E}_{\mathsf{clean}}}\big) < \frac{1}{2}.$$

Lemma 21 follows from Lemmas 22–24 proven below.

▶ **Lemma 22.** *It holds that:*

$$\Pr\big(\overline{\mathcal{E}_{\mathsf{clean},\mathsf{flag}}}\big) \leq \theta^{30}.$$

Proof. We have:

$$\begin{aligned}
\Pr(\exists i \in [n] : \mathsf{flag}(x_i) = 1) &\leq n \cdot \Pr(\mathsf{flag}(x_1) = 1) && \text{(As } x_i \text{ are identically distributed)} \\
&\leq n \cdot 2^{-\theta^4 m} && \text{(Lemma 3)} \\
&\leq 2^{-\theta^5 m}.
\end{aligned}$$

◀

▶ **Lemma 23.** *It holds that:*

$$\Pr\left(\overline{\mathcal{E}_{\mathsf{clean},x}}\right) \le \theta^{30}.$$

Proof. We have:

$$\begin{aligned}
&\Pr\left(\exists j \in [m] : \Pr\left(\sum_{i=1}^{n} x_{i,j} \ge \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) > 2^{-\theta^5 n}\right) \\
&\quad \le \sum_{j=1}^{m} \Pr\left(\Pr\left(\sum_{i=1}^{n} x_{i,j} \ge \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) > 2^{-\theta^5 n}\right) && \text{(Union bound)} \\
&\quad \le 2^{\theta^5 n} \cdot \sum_{j=1}^{m} \mathbb{E}\left[\Pr\left(\sum_{i=1}^{n} x_{i,j} \ge \theta n \mid \pi_{\mathsf{aug}}^{T'}\right)\right] && \text{(Markov's Inequality)} \\
&\quad \le 2^{\theta^5 n} \cdot \sum_{j=1}^{m} \Pr\left(\sum_{i=1}^{n} x_{i,j} \ge \theta n\right) \\
&\quad \le 2^{\theta^5 n} \cdot m \cdot 2^{-\theta^4 n} && \text{(Lemma 3)} \\
&\quad \le 2^{-\theta^5 n}.
\end{aligned}$$

◀

▶ **Lemma 24.** *It holds that:*

$$\Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}}\right) \le \theta^{30}.$$

Proof. To start, define the event $\mathcal{E}$ to be the set of all π_{aug}^{0} such that $\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\right)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\right) \le T \cdot \theta^{-40}$. By Markov's inequality and Lemma 17, we have $\Pr\left(\overline{\mathcal{E}}\right) \le \theta^{35}$. Using the chain rule, this implies that:

$$\Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}}\right) \le \Pr\left(\overline{\mathcal{E}}\right) + \Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}} \mid \mathcal{E}\right) \le \theta^{35} + \Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}} \mid \mathcal{E}\right).$$

Thus, it suffices to show that the last term is bounded by θ^{35}. We will show this holds even under a stronger conditioning. Specifically, we fix an arbitrary π_{aug}^{0} such that $\mathcal{E}$ happens and show that $\Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}} \mid \pi_{\mathsf{aug}}^{0}\right) \le \theta^{35}$. Fix any such π_{aug}^{0}. We first claim that:

▷ Claim 25. For all $\pi_{\mathsf{aug}}^{T'}$ that extend π_{aug}^{0} for which $\mathcal{E}_{\mathsf{clean},S}$ does not happen, we have:

$$\sum_{t=1}^{T'} \mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\right) \ge T'\theta^{400}.$$

The lemma now follows as, defining $P^{T'}$ to be the set of all $\pi_{\mathsf{aug}}^{T'}$ that extend π_{aug}^{0} for which $\mathcal{E}_{\mathsf{clean},S}$ does not happen, we get:

$$\begin{aligned}
\Pr\left(\overline{\mathcal{E}_{\mathsf{clean},S}} \mid \pi_{\mathsf{aug}}^{0}\right) &= \sum_{\pi_{\mathsf{aug}}^{T'} \in P^{T'}} \Pr\left(\pi_{\mathsf{aug}}^{T'} \mid \pi_{\mathsf{aug}}^{0}\right) \\
&\le \frac{1}{T'\theta^{400}} \sum_{\pi_{\mathsf{aug}}^{T'} \in P^{T'}} \Pr\left(\pi_{\mathsf{aug}}^{T'} \mid \pi_{\mathsf{aug}}^{0}\right) \cdot \sum_{t=1}^{T'} \mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\right) \\
& && \text{(Claim 25)} \\
&\le \frac{1}{T'\theta^{400}} \mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{0}\right)_{|\overline{\mathsf{R}^0}} \mid\mid \mathcal{D}_{|\overline{\mathsf{R}^0}}\right) && \text{(Corollary 20)} \\
&\le \frac{T}{T'\theta^{440}} && \text{(Choice of } \pi_{\mathsf{aug}}^{0}\text{)} \\
&\le \theta^{50}.
\end{aligned}$$

It remains to show Claim 25

Proof of Claim 25. Fix an arbitrary $\pi_{\mathsf{aug}}^{T'}$ as in the statement of the claim. As $\mathcal{E}_{\mathsf{clean},S}$ does not happen, we have that at least one of $S_{\mathsf{cell}}^{T'}$, $S_{\mathsf{pair}}^{T'}$, $S_{\mathsf{col}}^{T'}$ is non-empty. Applying Lemma 15, we get that for all $t \in [T']$, at least one of S_{cell}^{t}, S_{pair}^{t}, S_{col}^{t} is non-empty. Due to Lines 6 and 9, this means that in all iterations $t \in [T']$, the parties either execute Line 8 or they execute Line 11. Let $Z_{\mathsf{col}} \subseteq [T']$ be the set of all iterations t where parties execute Line 8 and $(i_1^t, j_1^t) \in S_{\mathsf{col}}^t \setminus S_{\mathsf{cell}}^t$. We claim that $|Z_{\mathsf{col}}| \leq T' \cdot \left(1 - \theta^{120}\right)$.

We prove this claim later, but assuming it for now, we have that either the parties execute Line 11 or the execute Line 8 and $(i_1^t, j_1^t) \in S_{\mathsf{cell}}^t$. In either case, note from Line 5 that $\mathbb{D}\Big(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\Big) \geq \theta^{200}$. This means that:

$$T'\theta^{120} \leq T' - |Z_{\mathsf{col}}| \leq \theta^{-200} \cdot \sum_{t \in [T'] \setminus Z_{col}} \mathbb{D}\Big(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{t-1}\right)_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}} \mid\mid \mathcal{D}_{|\mathsf{R}^t \setminus \mathsf{R}^{t-1}}\Big).$$

As the KL-divergence is non-negative, we have the claim. It remains to show that $|Z_{\mathsf{col}}| \leq T' \cdot \left(1 - \theta^{120}\right)$. For this, consider the following relation $M \subseteq Z_{\mathsf{col}} \times ([T'] \setminus Z_{\mathsf{col}})$. For $t \in Z_{\mathsf{col}}$ and $t' \in [T'] \setminus Z_{\mathsf{col}}$, we have $(t, t') \in M$ if and only if there exists $i' \in [n]$ such that $(i', j_1^t) \in \mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}$. We will show that:

(a) For all $t \in Z_{\mathsf{col}}$, there are at least $\theta^{100}n$ values of $t' \in [T'] \setminus Z_{\mathsf{col}}$ such that $(t, t') \in M$.

(b) For all $t' \in [T'] \setminus Z_{\mathsf{col}}$, there are at most $2n$ values of $t \in Z_{\mathsf{col}}$ such that $(t, t') \in M$.

Using Items a and b, we get:

$$|Z_{\mathsf{col}}| \cdot \theta^{100}n \leq |M| \leq (T' - |Z_{\mathsf{col}}|) \cdot 2n.$$

It follows that $T' \cdot \left(1 - \theta^{120}\right)$. It remains to show Items a and b. For Item a, fix an arbitrary $t \in Z_{\mathsf{col}}$ and, for all $0 \leq t' \leq T'$, define the value $Y(t') = \left|\left\{i' \in [n] \mid \mathsf{R}_{i', j_1^t}^{t'} = 1\right\}\right|$. Observe that from Algorithm 3 that

1. $Y(0) = 0$.
2. For all $t' \in [T']$, we have $Y(t'-1) \leq Y(t') \leq Y(t'-1) + 1$.
3. If $t' \in [T']$ is such that $Y(t') \leq Y(t'-1) + 1$, there exists $i' \in [n]$ such that $(i', j_1^t) \in \mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}$.

To see why Item a follows, consider the smallest $t^* \in Z_{\mathsf{col}}$ such that $j_1^t = j_1^{t^*}$. As $t^* \in Z_{\mathsf{col}}$, we have $\left(i_1^{t^*}, j_1^t\right) \in S_{\mathsf{col}}^{t^*}$. By Line 5, this means that $Y(t^* - 1) \geq \theta^{100} \cdot n$. Together with Items 1 and 2, this means that there are at least $\theta^{100}n$ values of $t' \in [t^* - 1]$ such that $Y(t') = Y(t'-1) + 1$. Moreover, none of these values are in Z_{col} as otherwise, Item 3 implies that $j_1^{t'} = j_1^t$, a contradiction to the choice of t^*. Using Item 3 again, it follows that all these values satisfy $(t, t') \in M$, as desired.

For Item b, fix an arbitrary $t' \in [T'] \setminus Z_{\mathsf{col}}$ and suppose for contradiction that there are at least $2n + 1$ values of $t \in Z_{\mathsf{col}}$ such that $(t, t') \in M$. Each of these $2n + 1$ values of t has a corresponding value of (i_1^t, j_1^t) which are all distinct (due to the fact that Line 5 only adds (i, j) to S_{col} if $\mathsf{R}_{i,j} = 0$). As all $i_1^t \in [n]$, the fact that there are $2n + 1$ distinct values of (i_1^t, j_1^t) imply that these values must contain at least 3 distinct values of j_1^t. This implies that there are at least three distinct values of j_1^t such that there exists $i' \in [n]$ for which $(i', j_1^t) \in \mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}$. This is a contradiction as Algorithm 3 guarantees that $\left|\mathsf{R}^{t'} \setminus \mathsf{R}^{t'-1}\right| \leq 2$. ◁

◀

6.3 Properties of Clean Transcripts

Lemma 21 shows that the probability that a transcript is not clean is small. Thus, a randomly sampled transcript is likely to be clean. We now establish some properties of clean transcripts. Throughout this subsection, we fix a transcript $\pi_{\mathsf{aug}}^{T'}$ for which $\mathcal{E}_{\mathsf{clean}}$ happens. This also fixes the values of $\mathsf{R}^0, \dots, \mathsf{R}^{T'}$ and other variables that are determined by $\pi_{\mathsf{aug}}^{T'}$. As $\mathcal{E}_{\mathsf{clean},S}$, $\mathcal{E}_{\mathsf{clean},x}$ and $\mathcal{E}_{\mathsf{clean,flag}}$ happen, we have $S_{\mathsf{cell}}^{T'} = S_{\mathsf{pair}}^{T'} = S_{\mathsf{col}}^{T'} = \emptyset$ and for all $j \in [m]$, it holds that $\Pr\left(\sum_{i \in [n]} x_{i,j} \geq \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) \leq 2^{-\theta^5 n}$ and that $\mathsf{flag}(x_i) = 0$ for all $i \in [n]$. Moreover, as the size R increases by at most 2 in any iteration, we have that:

$$\left|R^{T'}\right| \leq 2T' \leq \theta^{250} mn. \tag{4}$$

We claim that:

▶ **Lemma 26.** *For all $j \in [m]$, we have:*

$$\left|\mathsf{R}^{T'} \cap ([n] \times \{j\})\right| < n \implies \left|\mathsf{R}^{T'} \cap ([n] \times \{j\})\right| < \theta^{100} \cdot n.$$

Proof. Fix such a j. As $S_{\mathsf{cell}}^{T'} = S_{\mathsf{pair}}^{T'} = S_{\mathsf{col}}^{T'} = \emptyset$, we have by Lines 6 and 9 that $\mathsf{R}^{T'} = \mathsf{R}^{T'-1}$. Also, as $S_{\mathsf{col}}^{T'} = \emptyset$, we have by Line 5 that:

$$\left|\mathsf{R}^{T'-1} \cap ([n] \times \{j\})\right| < \theta^{100} \cdot n \vee \forall i' \in [n] : (i', j) \in \mathsf{R}^{T'-1}.$$

As $\mathsf{R}^{T'} = \mathsf{R}^{T'-1}$, we are done. ◀

Due to Lemma 26, we can partition the values $j \in [m]$ into two sets as follows:

$$J_{\mathsf{fix}} = \left\{ j \in [m] \mid \left|\mathsf{R}^{T'} \cap ([n] \times \{j\})\right| = n \right\}.$$
$$J_{\mathsf{unfix}} = \left\{ j \in [m] \mid \left|\mathsf{R}^{T'} \cap ([n] \times \{j\})\right| < \theta^{100} \cdot n \right\}. \tag{5}$$

Recall from Lemma 14 that $\pi_{\mathsf{aug}}^{T'}$ determines the values of $x_{i,j}$ for all $(i, j) \in \mathsf{R}^{T'}$. We have:

▶ **Lemma 27.** *For all $j \in J_{\mathsf{fix}}$, it holds that $\sum_{i=1}^{n} x_{i,j} < \theta n$.*

Proof. For all $j \in J_{\mathsf{fix}}$, we have by Equation (5) that $(i, j) \in \mathsf{R}^{T'}$ for all $i \in [n]$. By Lemma 14, this means that $\pi_{\mathsf{aug}}^{T'}$ determines $x_{i,j}$ for all $i \in [n]$. The lemma then follows as we have $\Pr\left(\sum_{i \in [n]} x_{i,j} \geq \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) \leq 2^{-\theta^5 n}$. ◀

▶ **Lemma 28.** *We have:*

1. *For all $(i, j) \notin \mathsf{R}^{T'}$ and all $b \in \{0, 1\}$,*

$$\Pr\left(x_{i,j} = b \mid \pi_{\mathsf{aug}}^{T'}\right) \geq \theta^{26}.$$

2. *For all $i \in [n]$, $j \neq j' \in [m]$ such that $(i, j), (i, j') \notin \mathsf{R}^{T'}$, and all $b \in \{0, 1\}$:*

$$\Pr\left((x_{i,j}, x_{i,j'}) = (1, b) \mid \pi_{\mathsf{aug}}^{T'}\right) \leq \theta^{20} \cdot \Pr\left(x_{i,j'} = b \mid \pi_{\mathsf{aug}}^{T'}\right).$$

Proof. Recall that $S_{\mathsf{cell}}^{T'} = S_{\mathsf{pair}}^{T'} = S_{\mathsf{col}}^{T'} = \emptyset$ and also recall from Corollary 13 that these sets are determined by $\pi_{\mathsf{aug}}^{T'-1}$. The fact that these sets are empty imply that $\pi_{\mathsf{aug}}^{T'-1}$ also determines $\pi_{\mathsf{aug}}^{T'}$, which means that they both determine one another. It follows that the distributions $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'-1}$ and $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}$ are identical. We now prove each part in turn.

1. As $\mathsf{R}^{T'-1} \subseteq \mathsf{R}^{T'}$, we have $(i,j) \notin \mathsf{R}^{T'-1}$. Recall that $S_{\mathsf{cell}}^{T'} = \emptyset$ implying from Line 5 that $\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'-1}\right)_{|(i,j)} \mid\mid \mathcal{D}_{|(i,j)}\right) < \theta^{200}$. As the distributions $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'-1}$ and $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}$ are identical, we get $\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}\right)_{|(i,j)} \mid\mid \mathcal{D}_{|(i,j)}\right) < \theta^{200}$. By Fact 39, this means that $\left\|\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}\right)_{|(i,j)} - \mathcal{D}_{|(i,j)}\right\|_{\mathrm{TV}} < \theta^{100}$ which by Definition 38 and the definition of $\mathcal{D}$ implies the result.
2. As $\mathsf{R}^{T'-1} \subseteq \mathsf{R}^{T'}$, we have $(i,j),(i,j') \notin \mathsf{R}^{T'-1}$. Recall that $S_{\mathsf{pair}}^{T'} = \emptyset$ implying from Line 5 that $\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'-1}\right)_{|(i,j),(i,j')} \mid\mid \mathcal{D}_{|(i,j),(i,j')}\right) < \theta^{100}$. As the distributions $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'-1}$ and $\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}$ are identical, we get $\mathbb{D}\left(\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}\right)_{|(i,j),(i,j')} \mid\mid \mathcal{D}_{|(i,j),(i,j')}\right) < \theta^{100}$. By Fact 39, this means that $\left\|\left(\mathcal{D} \mid \pi_{\mathsf{aug}}^{T'}\right)_{|(i,j),(i,j')} - \mathcal{D}_{|(i,j),(i,j')}\right\|_{\mathrm{TV}} < \theta^{50}$. It follows that:

$$\begin{aligned}
\Pr\left((x_{i,j}, x_{i,j'}) = (1,b) \mid \pi_{\mathsf{aug}}^{T'}\right) &\le \Pr((x_{i,j}, x_{i,j'}) = (1,b)) + \theta^{50} && \text{(Definition 38)} \\
&\le \theta^{25} \cdot \Pr(x_{i,j'} = b) + \theta^{50} \\
&\le \theta^{24} \cdot \Pr(x_{i,j'} = b) \\
&\le \theta^{24} \cdot \left(\Pr\left(x_{i,j'} = b \mid \pi_{\mathsf{aug}}^{T'}\right) + \theta^{50}\right) && \text{(Definition 38)} \\
&\le \theta^{20} \cdot \Pr\left(x_{i,j'} = b \mid \pi_{\mathsf{aug}}^{T'}\right). && \text{(Item 1)}
\end{aligned}$$

◀

Moreover, we have from Equation (4) that $J_{\mathsf{unfix}} \neq \emptyset$. Fix an arbitrary $j^* \in J_{\mathsf{unfix}}$. We now use j^* to define some important sets of the parties' inputs. Let x' be an input in the support of $\mathcal{D}$. We say that x' is relevant if for all $j \neq j^* \in [m]$, we have $\sum_{i=1}^{n} x_{i,j} \le \theta n$. We define $\mathcal{X}_{\mathsf{rel}}$ to be the set of all relevant inputs. We say that x' sets j^* to zero (respectively, one) if for all $i \in [n]$, we have $x'_{i,j^*} = x_{i,j^*}$ if $(i,j^*) \in \mathsf{R}^{T'}$ (recall from Lemma 14 that $\pi_{\mathsf{aug}}^{T'}$ determines the values of $x_{i,j}$ for all $(i,j) \in \mathsf{R}^{T'}$) and $x'_{i,j^*} = 0$ (resp. $x'_{i,j^*} = 1$) otherwise. We let $\mathcal{X}_{\mathsf{zero}}$ and $\mathcal{X}_{\mathsf{one}}$ be the set of all inputs that set j^* to zero and one respectively. We claim that:

▶ **Lemma 29.** *For all $x' \in \mathcal{X}_{\mathsf{rel}} \cap \mathcal{X}_{\mathsf{one}}$ such that $\Pr\left(x' \mid \pi_{\mathsf{aug}}^{T'}\right) > 0$, we have $\mathsf{GM\text{-}Test}_{\theta,n}^{m}(x') = \{0,1\}^m \backslash \{0^m\}$. For all $x' \in \mathcal{X}_{\mathsf{rel}} \cap \mathcal{X}_{\mathsf{zero}}$ such that $\Pr\left(x' \mid \pi_{\mathsf{aug}}^{T'}\right) > 0$, we have $\mathsf{GM\text{-}Test}_{\theta,n}^{m}(x') = \{0^m\}$.*

Proof. We only prove the former as the latter is analogous. For this, fix $x' \in \mathcal{X}_{\mathsf{rel}} \cap \mathcal{X}_{\mathsf{one}}$ and examine the cases in Equation (2). Recall that $\Pr\left(x' \mid \pi_{\mathsf{aug}}^{T'}\right) > 0$ implies that $\mathsf{flag}(x'_i) = 0$ for all $i \in [n]$. We finish the proof by showing that:

$$\mathsf{GapMaj}_{\theta,n}^{\theta,m}(x') = \{(\underbrace{0,\ldots,0}_{j^*-1 \text{ times}}, 1, \underbrace{0,\ldots,0}_{m-j^* \text{ times}})\}. \tag{6}$$

Indeed, we have from $x' \in \mathcal{X}_{\mathsf{rel}}$ that $\sum_{i=1}^{n} x'_{i,j} \le \theta n$ for all $j \neq j^* \in [m]$. We also have from $j^* \in J_{\mathsf{unfix}}$ and $x' \in \mathcal{X}_{\mathsf{one}}$ that $\sum_{i=1}^{n} x'_{i,j} \ge n - \theta^{100} n$. Moreover, we also have from $\mathsf{flag}(x'_i) = 0$ for all $i \in [n]$ that the Hamming weight of x'_i is at most $\frac{\theta m}{2}$ for all $i \in [n]$. Combine these to get Equation (6). ◀

▶ **Lemma 30.** *For all $\mathcal{X} \in \{\mathcal{X}_{\mathsf{one}}, \mathcal{X}_{\mathsf{zero}}\}$, it holds that:*

$$\Pr\Big(\mathsf{X} \in \mathcal{X} \cap \mathcal{X}_{\mathsf{rel}} \mid \pi_{\mathsf{aug}}^{T'}\Big) \geq \theta^{30n}.$$

Proof. We show the result for $\mathcal{X} = \mathcal{X}_{\mathsf{one}}$ as the proof for $\mathcal{X}_{\mathsf{zero}}$ is analogous. For $i \in [n]$, define $x^*_{i,j^*} = x_{i,j^*}$ if $(i, j^*) \in \mathsf{R}^{T'}$ (recall from Lemma 14 that $\pi_{\mathsf{aug}}^{T'}$ determines the values of $x_{i,j}$ for all $(i, j) \in \mathsf{R}^{T'}$) and $x^*_{i,j^*} = 0$ otherwise. Thus, we have:

$$\begin{aligned}\Pr\Big(\mathsf{X} \in \mathcal{X}_{\mathsf{one}} \mid \pi_{\mathsf{aug}}^{T'}\Big) &= \Pr\Big(\forall i \in [n] : x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ &= \prod_{i=1}^{n} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big),\end{aligned} \tag{7}$$

where the last step uses Lemma 16. We also have

$$\begin{aligned}&\Pr\Big(\mathsf{X} \in \mathcal{X}_{\mathsf{one}} \cap \overline{\mathcal{X}_{\mathsf{rel}}} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ &= \Pr\left(\forall i \in [n] : x'_{i,j^*} = x^*_{i,j^*} \land \exists j \neq j^* \in [m] : \sum_{i=1}^{n} x'_{i,j} > \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) \\ &\leq \sum_{j \neq j^* \in [m]} \Pr\left(\forall i \in [n] : x'_{i,j^*} = x^*_{i,j^*} \land \sum_{i=1}^{n} x'_{i,j} > \theta n \mid \pi_{\mathsf{aug}}^{T'}\right) && \text{(Union bound)} \\ &\leq \sum_{j \neq j^* \in [m]} \sum_{\substack{Z \subseteq [n] \\ |Z| = \theta n}} \Pr\Big(\forall i \in [n] : x'_{i,j^*} = x^*_{i,j^*} \land \forall i \in Z : x'_{i,j} = 1 \mid \pi_{\mathsf{aug}}^{T'}\Big) && \text{(Union bound)} \\ &\leq \sum_{j \neq j^* \in [m]} \sum_{\substack{Z \subseteq [n] \\ |Z| = \theta n}} \prod_{i \in \overline{Z}} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big) \prod_{i \in Z} \Pr\Big(\big(x'_{i,j}, x'_{i,j^*}\big) = \big(1, x^*_{i,j^*}\big) \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ & && \text{(Lemma 16)} \\ &\leq \sum_{j \neq j^* \in [m]} \sum_{\substack{Z \subseteq [n] \\ |Z| = \theta n}} \theta^{20 \cdot \theta n} \cdot \prod_{i=1}^{n} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big). && \text{(Lemma 28, Item 2)}\end{aligned}$$

Now, note that there are at most $n \cdot \left(\frac{3}{\theta}\right)^{\theta n} \leq \left(\frac{4}{\theta}\right)^{\theta n}$ terms in the sum (as $\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$). We get using $\theta < 1/2$ that:

$$\begin{aligned}&\Pr\Big(\mathsf{X} \in \mathcal{X}_{\mathsf{one}} \cap \overline{\mathcal{X}_{\mathsf{rel}}} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ &\leq \left(\frac{4}{\theta}\right)^{\theta n} \cdot \theta^{20 \cdot \theta n} \cdot \prod_{i=1}^{n} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ &\leq \frac{1}{2} \cdot \prod_{i=1}^{n} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big).\end{aligned}$$

Combining with Equation (7), we get:

$$\begin{aligned}\Pr\Big(\mathsf{X} \in \mathcal{X}_{\mathsf{one}} \cap \mathcal{X}_{\mathsf{rel}} \mid \pi_{\mathsf{aug}}^{T'}\Big) &\geq \frac{1}{2} \cdot \prod_{i=1}^{n} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ &= \frac{1}{2} \cdot \prod_{i : (i, j^*) \notin \mathsf{R}^{T'}} \Pr\Big(x'_{i,j^*} = x^*_{i,j^*} \mid \pi_{\mathsf{aug}}^{T'}\Big) \\ & \qquad \text{(As } \pi_{\mathsf{aug}}^{T'} \text{ determines } x_{i,j} \text{ for all } (i, j) \in \mathsf{R}^{T'}) \\ &\geq \theta^{30n}. && \text{(Lemma 28, Item 1)}\end{aligned}$$

◀

6.4 Finishing the Proof

We are now ready to finish the proof of Theorem 10.

Proof of Theorem 10. Recall that it suffices to show Equation (3). We have:

$$\begin{aligned}\Pr\big(\Pi(\mathsf{X}) \in \textsf{GM-Test}^m_{\theta,n}(\mathsf{X})\big) &\le \Pr\big(\overline{\mathcal{E}_{\mathsf{clean}}}\big) + \Pr(\mathcal{E}_{\mathsf{clean}}) \cdot \Pr\big(\Pi(\mathsf{X}) \in \textsf{GM-Test}^m_{\theta,n}(\mathsf{X}) \mid \mathcal{E}_{\mathsf{clean}}\big) & \text{(Union bound)}\\ &= 1 - \Pr(\mathcal{E}_{\mathsf{clean}}) \cdot \Pr\big(\Pi(\mathsf{X}) \notin \textsf{GM-Test}^m_{\theta,n}(\mathsf{X}) \mid \mathcal{E}_{\mathsf{clean}}\big)\\ &\le 1 - \frac{1}{2} \cdot \Pr\big(\Pi(\mathsf{X}) \notin \textsf{GM-Test}^m_{\theta,n}(\mathsf{X}) \mid \mathcal{E}_{\mathsf{clean}}\big). & \text{(Lemma 21)}\end{aligned}$$

Thus, it suffices to lower bound the last probability by θ^{30n}. We will show this holds even under a stronger conditioning. Specifically, we fix an arbitrary $\pi^{T'}_{\mathsf{aug}}$ such that $\mathcal{E}_{\mathsf{clean}}$ happens and show that $\Pr\Big(\Pi(\mathsf{X}) \notin \textsf{GM-Test}^m_{\theta,n}(\mathsf{X}) \mid \pi^{T'}_{\mathsf{aug}}\Big) \ge \theta^{30n}$. Fix any such $\pi^{T'}_{\mathsf{aug}}$ and recall that fixing $\pi^{T'}_{\mathsf{aug}}$ also fixes the output of the protocol. As the sets in the two cases of Lemma 29 are disjoint, we have:

$$\begin{aligned}&\Pr\Big(\Pi(\mathsf{X}) \notin \textsf{GM-Test}^m_{\theta,n}(\mathsf{X}) \mid \pi^{T'}_{\mathsf{aug}}\Big)\\ &\quad\ge \min\Big(\Pr\Big(\Pi(\mathsf{X}) \in \mathcal{X}_{\mathsf{rel}} \cap \mathcal{X}_{\mathsf{one}} \mid \pi^{T'}_{\mathsf{aug}}\Big), \Pr\Big(\Pi(\mathsf{X}) \in \mathcal{X}_{\mathsf{rel}} \cap \mathcal{X}_{\mathsf{zero}} \mid \pi^{T'}_{\mathsf{aug}}\Big)\Big)\\ &\quad\ge \theta^{30n}. & \text{(Lemma 30)}\end{aligned}$$

◀

References

1 Abhinav Aggarwal, Varsha Dani, Thomas P Hayes, and Jared Saia. Distributed computing with channel noise. *arXiv preprint*, 2016. arXiv:1612.05943.

2 Noga Alon, Mark Braverman, Klim Efremenko, Ran Gelles, and Bernhard Haeupler. Reliable communication over highly connected noisy networks. In *Symposium on Principles of Distributed Computing (DISC)*, pages 165–173. ACM, 2016.

3 Ziv Bar-Yossef, Thathachar S Jayram, Ravi Kumar, and D Sivakumar. An information statistics approach to data stream and communication complexity. *Journal of Computer and System Sciences*, 68(4):702–732, 2004.

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

5 Mark Braverman, Klim Efremenko, Ran Gelles, and Bernhard Haeupler. Constant-rate coding for multiparty interactive communication is impossible. In *Symposium on Theory of Computing (STOC)*, pages 999–1010. ACM, 2016.

6 Mark Braverman and Anup Rao. Information equals amortized communication. In Rafail Ostrovsky, editor, *Symposium on Foundations of Computer Science (FOCS)*, pages 748–757. IEEE Computer Society, 2011.

7 Mark Braverman, Anup Rao, Omri Weinstein, and Amir Yehudayoff. Direct product via round-preserving compression. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, pages 232–243. Springer, 2013.

8 Mark Braverman, Anup Rao, Omri Weinstein, and Amir Yehudayoff. Direct products in communication complexity. In *Symposium on Foundations of Computer Science (FOCS)*, pages 746–755. IEEE, 2013.

9 Keren Censor-Hillel, Ran Gelles, and Bernhard Haeupler. Making asynchronous distributed computations robust to noise. *Distributed Computing*, 32:405–421, 2019.

10 Keren Censor-Hillel, Bernhard Haeupler, D Ellis Hershkowitz, and Goran Zuzic. Broadcasting in noisy radio networks. In *Symposium on Principles of Distributed Computing (PODC)*, pages 33–42, 2017.

11 Keren Censor-Hillel, Bernhard Haeupler, D Ellis Hershkowitz, and Goran Zuzic. Erasure correction for noisy radio networks. In *International Symposium on Distributed Computing (DISC)*, 2019.

12 Amit Chakrabarti, Yaoyun Shi, Anthony Wirth, and Andrew Yao. Informational complexity and the direct sum problem for simultaneous message complexity. In *Symposium on Foundations of Computer Science (FOCS)*, pages 270–278. IEEE, 2001.

13 Imrich Chlamtac and Shay Kutten. On broadcasting in radio networks-problem analysis and protocol design. *IEEE Trans. Communications*, 33(12):1240–1246, 1985.

14 Klim Efremenko, Gillat Kol, Dmitry Paramonov, and Raghuvansh R. Saxena. Computation over the noisy broadcast channel with malicious parties. In *Innovations in Theoretical Computer Science Conference, (ITCS)*, volume 185, pages 82:1–82:19, 2021.

15 Klim Efremenko, Gillat Kol, Dmitry Paramonov, and Raghuvansh R. Saxena. Tight bounds for general computation in noisy broadcast networks. In *Symposium on Foundations of Computer Science (FOCS)*, pages 634–645, 2021.

16 Klim Efremenko, Gillat Kol, Dmitry Paramonov, and Raghuvansh R. Saxena. Protecting single-hop radio networks from message drops. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 261 of *LIPIcs*, pages 53:1–53:20, 2023.

17 Klim Efremenko, Gillat Kol, and Raghuvansh Saxena. Interactive coding over the noisy broadcast channel. In *Symposium on Theory of Computing (STOC)*, pages 507–520. ACM, 2018.

18 Klim Efremenko, Gillat Kol, and Raghuvansh Saxena. Radio network coding requires logarithmic overhead. In *Foundations of Computer Science (FOCS)*, pages 348–369, 2019.

19 Klim Efremenko, Gillat Kol, and Raghuvansh Saxena. Interactive error resilience beyond 2/7. In *Symposium on Theory of Computing (STOC)*. ACM, 2020.

20 Klim Efremenko, Gillat Kol, and Raghuvansh R. Saxena. Noisy beeps. In Yuval Emek and Christian Cachin, editors, *Symposium on Principles of Distributed Computing (PODC)*, pages 418–427, 2020.

21 Klim Efremenko, Gillat Kol, and Raghuvansh R. Saxena. Optimal error resilience of adaptive message exchange. In *Symposium on Theory of Computing (STOC)*, pages 1235–1247. ACM, 2021.

22 Tomás Feder, Eyal Kushilevitz, Moni Naor, and Noam Nisan. Amortized communication complexity. *SIAM Journal on computing*, 24(4):736–750, 1995.

23 Uriel Feige and Joe Kilian. Finding OR in a noisy broadcast network. *Information Processing Letters*, 73(1-2):69–75, 2000.

24 Robert G. Gallager. Finding parity in a simple broadcast network. *IEEE Transactions on Information Theory*, 34(2):176–180, 1988.

25 Abbas El Gamal. Open problems presented at the 1984 workshop on specific problems in communication and computation sponsored by bell communication research. *"Open Problems in Communication and Computation", by Thomas M. Cover and B. Gopinath (editors). Springer-Verlag*, 1987.

26 Anat Ganor, Gillat Kol, and Ran Raz. Exponential separation of information and communication for boolean functions. *Journal of the ACM*, 63(5):46:1–46:31, 2016.

27 Anat Ganor, Gillat Kol, and Ran Raz. Exponential separation of communication and external information. *SIAM Journal on computing*, 50(3), 2021.

28 Ran Gelles and Yael T Kalai. Constant-rate interactive coding is impossible, even in constant-degree networks. *IEEE Transactions on Information Theory*, 65(6):3812–3829, 2019.

29 Ran Gelles, Yael Tauman Kalai, and Govind Ramnarayan. Efficient multiparty interactive coding—part i: Oblivious insertions, deletions and substitutions. *IEEE Transactions on Information Theory*, 67(6):3411–3437, 2021.

30 Ran Gelles, Yael Tauman Kalai, and Govind Ramnarayan. Efficient multiparty interactive coding—part ii: Non-oblivious noise. *IEEE Transactions on Information Theory*, 68(7):4723–4749, 2022.

31 Ran Gelles, Ankur Moitra, and Amit Sahai. Efficient and explicit coding for interactive communication. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 768–777. IEEE, 2011.

32 Navin Goyal, Guy Kindler, and Michael Saks. Lower bounds for the noisy broadcast problem. *SIAM Journal on Computing*, 37(6):1806–1841, 2008.

33 Bernhard Haeupler. Interactive channel capacity revisited. In *Foundations of Computer Science (FOCS)*, pages 226–235. IEEE, 2014.

34 Prahladh Harsha, Rahul Jain, David McAllester, and Jaikumar Radhakrishnan. The communication complexity of correlation. In *Conference on Computational Complexity (CCC)*, pages 10–23. IEEE, 2007.

35 William M. Hoza and Leonard J. Schulman. The adversarial noise threshold for distributed protocols. In *Symposium on Discrete Algorithms (SODA)*, pages 240–258, 2016.

36 Abhishek Jain, Yael Tauman Kalai, and Allison Bishop Lewko. Interactive coding for multiparty protocols. In *Symposium on Theory of computing (STOC)*, pages 1–10, 2015.

37 Rahul Jain. New strong direct product results in communication complexity. *Journal of the ACM*, 62(3):1–27, 2015.

38 Rahul Jain, Attila Pereszlényi, and Penghui Yao. A direct product theorem for two-party bounded-round public-coin communication complexity. *Algorithmica*, 76:720–748, 2016.

39 Rahul Jain, Jaikumar Radhakrishnan, and Pranab Sen. A direct sum theorem in communication complexity via message compression. In *Colloquium on Automata, Languages, and Programming (ICALP)*, pages 300–315. Springer, 2003.

40 Mauricio Karchmer, Eyal Kushilevitz, and Noam Nisan. Fractional covers and communication complexity. *SIAM Journal on Discrete Mathematics*, 8(1):76–92, 1995.

41 Hartmut Klauck. A strong direct product theorem for disjointness. In *Symposium on Theory of Computing (STOC)*, pages 77–86, 2010.

42 Eyal Kushilevitz and Yishay Mansour. Computation in noisy radio networks. *SIAM Journal on Discrete Mathematics (SIDMA)*, 19(1):96–108, 2005.

43 Allison Lewko and Ellen Vitercik. Balancing communication for multi-party interactive coding. *arXiv preprint*, 2015. `arXiv:1503.06381`.

44 Manuj Mukherjee and Ran Gelles. Multiparty interactive coding over networks of intersecting broadcast links. *IEEE Journal on Selected Areas in Information Theory*, 2(4):1078–1092, 2021.

45 Ilan Newman. Computing in fault tolerance broadcast networks. In *Computational Complexity Conference (CCC)*, pages 113–122, 2004.

46 Sridhar Rajagopalan and Leonard J. Schulman. A coding theorem for distributed computation. In *Symposium on the Theory of Computing (STOC)*, pages 790–799, 1994.

47 Leonard J Schulman. Communication on noisy channels: A coding theorem for computation. In *Foundations of Computer Science (FOCS)*, pages 724–733. IEEE, 1992.

48 Andrew Chi-Chih Yao. On the complexity of communication under noise. *invited talk in the 5th ISTCS Conference*, 1997.

49 Huacheng Yu. Strong xor lemma for communication with bounded rounds. In *Symposium on Foundations of Computer Science (FOCS)*, pages 1186–1192. IEEE, 2022.

A Information Theory Preliminaries

Recall that we use sans-serif letters to denote random variables. We reserve E to denote an arbitrary event. All random variables will be assumed to be discrete and we shall adopt the convention $0 \log \frac{1}{0} = 0$. When it is clear from context, we may abbreviate the event $\mathsf{X} = x$ as just x. All logarithms are taken with base 2.

A.1 Entropy

▶ **Definition 31** (Entropy). *The (binary) entropy of* X *is defined as:*

$$\mathbb{H}(\mathsf{X}) = \sum_{x \in \mathsf{supp}(\mathsf{X})} \Pr(x) \cdot \log \frac{1}{\Pr(x)}.$$

The entropy of X *conditioned on* E *is defined as:*

$$\mathbb{H}(\mathsf{X} \mid E) = \sum_{x \in \mathsf{supp}(\mathsf{X})} \Pr(x \mid E) \cdot \log \frac{1}{\Pr(x \mid E)}.$$

▶ **Definition 32** (Conditional Entropy). *We define the conditional entropy of* X *given* Y *and* E *as:*

$$\mathbb{H}(\mathsf{X} \mid \mathsf{Y}, E) = \sum_{y \in \mathsf{supp}(\mathsf{Y})} \Pr(y \mid E) \cdot \mathbb{H}(\mathsf{X} \mid y, E).$$

Henceforth, we shall omit writing the $\mathsf{supp}(\cdot)$ when it is clear from context.

▶ **Lemma 33.** *It holds for all* X *and* E *that:*

$$0 \leq \mathbb{H}(\mathsf{X} \mid E) \leq \log(|\mathsf{supp}(\mathsf{X})|).$$

The second inequality is tight if and only if X *conditioned on* E *is the uniform distribution over* $\mathsf{supp}(\mathsf{X})$.

A.2 Mutual Information

▶ **Definition 34** (Mutual Information). *The mutual information between* X *and* Y *is defined as:*

$$\mathbb{I}(\mathsf{X} : \mathsf{Y}) = \mathbb{H}(\mathsf{X}) - \mathbb{H}(\mathsf{X} \mid \mathsf{Y}) = \mathbb{H}(\mathsf{Y}) - \mathbb{H}(\mathsf{Y} \mid \mathsf{X}).$$

The mutual information between X *and* Y *conditioned on* Z *is defined as:*

$$\mathbb{I}(\mathsf{X} : \mathsf{Y} \mid \mathsf{Z}) = \mathbb{H}(\mathsf{X} \mid \mathsf{Z}) - \mathbb{H}(\mathsf{X} \mid \mathsf{YZ}) = \mathbb{H}(\mathsf{Y} \mid \mathsf{Z}) - \mathbb{H}(\mathsf{Y} \mid \mathsf{XZ}).$$

▶ **Fact 35.** *We have* $0 \leq \mathbb{I}(\mathsf{X} : \mathsf{Y} \mid \mathsf{Z}) \leq \mathbb{H}(\mathsf{X})$.

▶ **Lemma 36.** *We have:*

$$\mathbb{I}(\mathsf{X} : \mathsf{Y} \mid \mathsf{Z}) = \sum_{x,y,z} \Pr(x, y, z) \cdot \log \frac{\Pr(x, y \mid z)}{\Pr(x \mid z) \Pr(y \mid z)}.$$

A.3 KL Divergence

▶ **Definition 37** (KL Divergence). *If μ, ν are two distributions over the same (finite) set Ω, the Kullback-Leibler (KL) Divergence between μ and ν is defined as:*

$$\mathbb{D}(\mu \mid\mid \nu) = \sum_{\omega \in \Omega} \mu(\omega) \cdot \log \frac{\mu(\omega)}{\nu(\omega)}.$$

For a finite non-empty set S, we shall use $\mathcal{U}(S)$ to denote the uniform distribution over S. We omit S from the notation when it is clear from the context. We use $\mathsf{dist}(\mathsf{X} \mid E)$ to denote the distribution of the random variable X conditioned on the event E.

A.4 Total Variation Distance

▶ **Definition 38** (Total variation distance). *Let μ, ν be two distributions over the same (finite) set Ω. The total variation distance between μ and ν is defined as:*

$$\|\mu - \nu\|_{\mathrm{TV}} = \max_{\Omega' \subseteq \Omega} \sum_{\omega \in \Omega'} \mu(\omega) - \nu(\omega).$$

▶ **Fact 39** (Pinsker's inequality). *Let μ, ν be two distributions over the same set Ω. It holds that:*

$$\|\mu - \nu\|_{\mathrm{TV}} \leq \sqrt{\frac{1}{2} \cdot \mathbb{D}(\mu \mid\mid \nu)}.$$