Electronic Colloquium on Computational Complexity, Report No. 160 (2021)

Tight Bounds for General Computation in Noisy Broadcast Networks

Klim Efremenko[*]
Ben-Gurion University

Gillat Kol[†]
Princeton University

Dmitry Paramonov[‡]
Princeton University

Raghuvansh R. Saxena[§]
Princeton University

Abstract

Let Π be a protocol over the n-party *broadcast channel*, where in each round, a pre-specified party broadcasts a symbol to all other parties. We wish to design a scheme that takes such a protocol Π as input and outputs a noise resilient protocol Π' that *simulates* Π over the *noisy broadcast channel*, where each received symbol is flipped with a fixed constant probability, independently. What is the minimum overhead in the number of rounds that is incurred by any such simulation scheme?

A classical result by Gallager from the 80's shows that *non-interactive* T-round protocols, where the bit communicated in every round is independent of the communication history, can be converted to noise resilient ones with only an $\mathcal{O}(\log\log T)$ multiplicative overhead in the number of rounds. Can the same be proved for any protocol? Or, are there protocols whose simulation requires an $\Omega(\log T)$ overhead (which always suffices)?

We answer both the above questions in the negative: We give a simulation scheme with an $\tilde{O}(\sqrt{\log T})$ overhead for every protocol and channel alphabet. We also prove an (almost) matching lower bound of $\Omega(\sqrt{\log T})$ on the overhead required to simulate the pointer chasing protocol with $T = n$ and polynomial alphabet.

[*]klimefrem@gmail.com
[†]gillat.kol@gmail.com
[‡]dp20@cs.princeton.edu
[§]rrsaxena@princeton.edu

ISSN 1433-8092

Contents

1 Introduction

We study the *noisy broadcast model*, a noisy version of the standard *broadcast model* (*a.k.a.*, *shared blackboard* model). In the noisy broadcast model, a set of n parties, each holding a private input x_i, communicate with the mutual goal of computing $f(x_1, \cdots, x_n)$, for some function f that is known to all the parties. The communication is carried out in rounds: In each round, a pre-specified party broadcasts a symbol from some finite alphabet set Γ to all the other parties. However, the symbol received by each party is randomly flipped with some fixed constant probability $\epsilon > 0$, independently for each party and round.

The noisy broadcast model was first suggested by El Gamal [Gam87] in 1984, and later popularized by Yao [Yao97], as a simple abstraction for studying the effect of noise on highly distributed wireless systems. El Gamal posted the following challenge: What is the round complexity of the *message exchange function* (*a.k.a.*, *identity function*) $f(x_1, \cdots, x_n) = (x_1, \cdots, x_n)$ in this noisy broadcast model with binary alphabet and bit inputs? Clearly, when there is no noise ($\epsilon = 0$), this function can be computed in n rounds: The parties simply take turns broadcasting their inputs. An $\mathcal{O}(n \log n)$-round protocol over the noisy broadcast channel is also easy: By a union bound argument, it suffices to have every party broadcast its input bit $\mathcal{O}(\log n)$ times as this means that a majority of the bits received by any other party are correct with high probability.

This simple protocol was shown to be sub-optimal by Gallager [Gal88], who gave an elegant $\mathcal{O}(n \log \log n)$-round protocol computing the message exchange function. Gallager's protocol was later proved to be optimal by a beautiful paper of [GKS08]. In fact, Gallager's protocol extends to any function f that can be computed by a *non-interactive* T-round (noiseless) broadcast protocol with an overhead of $\mathcal{O}(\log \log T)$. By "non-interactive", we mean that the symbol communicated in each round of the protocol is independent of the symbols communicated in all previous rounds[1].

Can *any* function f be computed over the noisy broadcast channel with a similar $\mathcal{O}(\log \log T)$ blowup in the number of rounds? Prior to our work, only the trivial bound of $\mathcal{O}(\log T)$ was known for general functions.

1.1 Our Result

We show that while a Gallager-like scheme with a blowup of $\mathcal{O}(\log \log T)$ is impossible in the general case, it is possible to out-perform the trivial scheme with $\mathcal{O}(\log T)$ blowup mentioned above. Specifically, we give a scheme that compiles any T-round protocol Π over the (n, Γ)-broadcast channel to a protocol Π' over the (n, ϵ, Γ)-noisy broadcast channel with only $\tilde{\mathcal{O}}(\sqrt{\log T})$ overhead. Here, n is the number of communicating parties, and Γ is the channel's alphabet set. In the noisy channel, the symbol received by each of the parties in each of the rounds is the symbol broadcast in that round with probability $1 - \epsilon$, and a

[1] In other words, the communicated symbol is only a function of the round number, the input of the communicating player, and its randomness.

uniformly random symbol[2] with the remaining probability[3]. Theorem 1.1 below gives an informal statement of this result. For a formal statement, see Theorem 4.1.

Theorem 1.1 (Upper Bound, Informal). *Let $n, T > 0$ and Γ be an arbitrary non-empty set. Let Π be a T-round protocol over the (n, Γ)-noisy broadcast channel. For all constant $\epsilon \in [0, 1/2]$, there exists a protocol Π' that simulates Π over the (n, ϵ, Γ)-noisy broadcast channel[4], has $T \cdot \tilde{\mathcal{O}}(\sqrt{\log T})$ rounds, and errs with probability polynomially small in T.*

We complement our upper bound result with an almost matching lower bound, at least for the interesting regime of parameters where $T, |\Gamma| = \text{poly}(n)$ (see additional discussion in Section 1.5). An informal statement of our lower bound is given in Theorem 1.2 below. For a formal statement, see Theorem 5.1.

Theorem 1.2 (Lower Bound, Informal). *Let $n > 0$, $\epsilon \in [0, 1/2]$, and Γ be a set with $|\Gamma| \geq n^{200}$. There exists a deterministic n-round protocol Π computing a boolean function over the (n, Γ)-broadcast channel such that any randomized protocol that simulates Π over the (n, ϵ, Γ)-noisy broadcast channel has $\Omega(n\sqrt{\log n})$ rounds[5].*

The combination of Theorems 1.1 and 1.2 shows that the optimal overhead of a simulation scheme for the broadcast channel is $\tilde{\Theta}(\sqrt{\log n})$, at least for protocols of length polynomial in n. This is unique, as, for other channels, the overhead is either close to $\Omega(\log n)$ (up to lower order factors), which is typically the worst possible, or very close to $\mathcal{O}(1)$, which is the best possible. Thus, we show the first channel where the overhead is in the middle of the two extremes. We also note that the existence of an upper bound significantly better than the trivial $\mathcal{O}(\log n)$ makes the lower bound proof very different from other such proofs in the literature (see Section 2 for a detailed outline).

We mention that Yao [Yao97] (see also [GKS08, New04]) posted the question of whether any boolean function can be computed by an $\mathcal{O}(n)$-round protocol (note that [GKS08]'s lower bound is for the message exchange function that has a large output). However, unlike our setting, in Yao's setting, each party only has a single input bit and the alphabet set is binary. In this setting, linear protocols for several basic functions were given, see Section 1.3, but no lower bounds are known. While Yao's question is still open, Theorem 1.2 is the first super-linear lower bound for computing a boolean function in the noisy broadcast model.

[2] As it is possible for the random symbol to be the same as the sent symbol, each party receives each sent symbol correctly with probability $1 - \epsilon + \epsilon/|\Gamma|$ and receives each of the other $|\Gamma| - 1$ symbols with probability $\epsilon/|\Gamma|$. In particular, for the binary alphabet ($|\Gamma| = 2$), each communicated bit is received correctly by each party with probability $1 - \epsilon/2$.

[3] The choice that the parties receive a uniformly random symbol in the case of noise is made for convenience. Both our upper bound and lower bound work in stronger models, *e.g.*, the lower bound can be made to work in an erasure based model while the upper bound can be made to work even if the parties receive an adversarial symbol with probability ϵ.

[4] By "Π' that simulates Π", we mean that a transcript for Π can be retrieved from a transcript for Π', see Section 3.5 for a formal definition.

[5] In fact, the same lower bound also holds for protocols Π' that can *estimate* the boolean function with a *polynomially small advantage* over random guessing.

1.2 Non-Adaptive *vs.* Adaptive Simulation

Non-adaptive *vs.* adaptive protocols. In the classical noisy broadcast model, defined by [Gam87] and assumed by the current work and by prior work [Gal88, Yao97, KM05, FK00, New04, GKS08, EKPS21], the order of communication in the protocol is *predetermined* and is independent of the players' inputs and the channel's noise (and therefore also independent of the received transcripts). Such protocols are called *non-adaptive* or *oblivious* protocols. Non-adaptive protocols are widely studied as they model certain common types of wireless networks, prevent *signaling*[6], and can trivially ensure that exactly one party is broadcasting in every round.

Inspired by the radio network models in distributed computing [CK85], in a recent work [EKS18] we defined an *adaptive* version of the noisy broadcast model, where a party decides whether to broadcast or not based on its input and its received transcript. As hinted above, such a model is prone to collision rounds (where more than one party broadcasts) and silent rounds (where no party broadcasts). To keep the model as general as possible for protocol design purposes, we assumed that the communication in collision and silent rounds is governed by an adversary that can arbitrarily corrupt the symbol received by each party.

Separation of non-adaptive and adaptive simulations. The main result of [EKS18] is that any protocol over the broadcast channel (which is, by definition, *non-adaptive* and noiseless) can be simulated by a protocol in the *adaptive* noisy broadcast model with only a constant blowup in the number of rounds. This result circumvents the lower bound of [GKS08] and gives a separation between adaptive and non-adaptive simulations: For every (non-adaptive) broadcast protocol Π, there exists a protocol Π' over the adaptive noisy broadcast model that simulates Π with $\mathcal{O}(1)$ blowup. However, there exists a (non-adaptive) broadcast protocol Π such that every (non-adaptive) noisy broadcast protocol that simulates Π has a blowup of $\Omega(\log\log n)$. Our Theorem 1.2 *amplifies the gap* in this separation to be $\mathcal{O}(1)$ blowup for an adaptive simulation *vs.* $\Omega(\sqrt{\log n})$ blowup for a non-adaptive simulation.

1.3 Additional Related Work

Noise tolerant protocols for specific n-bit functions. Prior works on the noisy broadcast model mostly focused on the case where each party has a single input bit and the parties wish to evaluate a specific n-bit function over the binary noisy broadcast channel. Since Gallager's result [Gal88] implies an $\mathcal{O}(n\log\log n)$-round protocol for all such functions (parties communicate to exchange their inputs and then each evaluates the target function by itself), linear or near-linear protocols were targeted.

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

For instance, an $\mathcal{O}(n)$-round protocol for computing the majority function (or any other threshold function) in the related, but stronger, *statistical* noisy broadcast model[7] was given by [KM05]. An $\mathcal{O}(n \log^* n)$-round protocol for the *or* function was given by [FK00]. An improved $\mathcal{O}(n)$-round protocol for the *or* function and other boolean functions was given by [New04]. An $\mathcal{O}(n)$-round protocol for the *parity* function and all other functions whose value only depends on the Hamming weight of the input was given by [GKS08].

The message exchange function was also considered under variants of the noisy broadcast model. [GHM18] give an $\mathcal{O}(n \cdot \log^* n)$-round protocol for the message exchange function in the *erasure* noisy broadcast model, which is a relaxed model where receptions are randomly erased (replaced with a '?') instead of flipped. In a recent work [EKPS21], we give a noise resilient message exchange protocol that works even if an adversary controls a constant fraction of the parties.

Interactive coding. The field of interactive coding aims to convert (general) protocols designed to work over noiseless channels to noise resilient protocols. The study of interactive codes was initiated by a seminal paper of Schulman [Sch92] that considered (non-adaptive) two-party protocols. The two-party adaptive channel was later studied by [Hae14, GHS14, AGS16, EKS20a, EKS21]. Interactive codes for multi-party distributed channels were also studied, including codes for peer-to-peer networks [RS94, ABE+16, BEGH16] and codes for various adaptive broadcast channels [CHHZ17, EKS18, EKS19, EKS20b, AGL20]. While the round complexity of some specific functions was studied over the classical (non-adaptive) noisy broadcast channel, as discussed above, our result is the first to consider general interactive coding over this channel.

As mentioned in Section 1.2, in [EKS18], we give a constant rate simulation scheme over the adaptive broadcast channel. In contrast, for other multi-party channels, an $\tilde{\Omega}(\log n)$ blowup in the length of the protocol was shown to be unavoidable. One such example is [BEGH16], proving a near-logarithmic blowup for the peer-to-peer model. Another example is [EKS19], where we show that a logarithmic blowup is inherent in the case that the parties are broadcasting over a general network and a message broadcast by a party can only be received by its neighbors in the network. A similar blowup is also shown for the beeping channel, which is an adaptive channel that models very basic signal-based communication[8] [EKS20b, AGL20]).

1.4 Our Techniques

We next give a very high level survey of our efforts. For a detailed overview, see Section 2.

[7] In the statistical model, it is assumed that the bit received by every party in every round is incorrect with probability *exactly* $\epsilon/2$ (as opposed to *at most* $\epsilon/2$). We mention that our lower bound in Theorem 1.2 holds even in the statistical model.

[8] In every round of the beeping model, each party may "beep" and a beep is received by all parties if at least one party beeped. Equivalently, each party communicates a bit in every round and the logical *or* of the bits is received by all parties.

Upper bound. We design our interactive coding scheme by first breaking the protocol into chunks of length k, for a suitable k, and simulating it chunk-by-chunk. As there are k rounds in any one chunk, a chunk involves at most k parties. Thus, a straightforward way to simulate a chunk is to simply repeat each message $\mathcal{O}(\log k)$ times and have the parties decode by majority. The error probability of this simulation is $\frac{1}{\text{poly}(k)}$, which is exponentially small in the number of times a *single* party broadcasts in the simulation. To get our upper bound, we show a way of simulating one chunk with an error probability that is exponentially small in the total *length* of the simulation, which is much smaller than that of the straightforward simulation above.

In fact, if the chunk being simulated is *non-interactive*, then the main idea from Gallager's protocol [Gal88] can be used to get a simulation scheme with such an error probability. At a high level, the property of non-interactive chunks used here is that the k parties only need to know the simulated transcript when they are outputting it at the end of the simulation, and, in particular, do not need to know it while computing what symbols to broadcast *during* the simulation.

However, chunks may be interactive, and we extend Gallager's protocol to this case by designing a *transformation from interactive to non-interactive chunks*. The key observation is that the party broadcasting the first symbol in the chunk does not need to know the simulated transcript to compute the symbol it wants to broadcast. Thus, the first symbol in the chunk is still amenable to a Gallager style argument. To get the remaining symbols, we re-execute the simulation $\mathcal{O}(k)$ times, each time getting an extra symbol of the transcript. As a small fraction of the executions may be corrupted, we also use ideas from *interactive coding* to get that the overall simulation is correct except with probability exponentially small in the length of the simulation, which is now $\tilde{\mathcal{O}}(k^2)$.

Finally, note that our simulation is $\mathcal{O}(k)$ times longer than the noiseless chunk it simulates, but our lower bound shows that this blowup is necessary.

Lower bound. The lower bound proof is more involved and it first reduces the problem to a problem in the *adversarial* setting: We observe that if the failure probability of the protocol over the noisy broadcast channel with error rate $1/2$ is at most $2^{-\Delta}$, then the protocol is also resilient to Δ adversarial corrupted receptions. The reason is that the probability of any Δ corruptions under the noisy broadcast channel is $2^{-\Delta}$. The main ingredient in our lower bound is an argument showing that if an interactive coding scheme is resilient to Δ adversarial corruptions, then it must have length at least $\Omega(n \cdot \sqrt{\Delta})$.

We prove this by contradiction, showing that one can *trade rounds against adversarial resilience* in the following sense: For any $t > 0$ one can either "remove" $\Omega(n)$ rounds from the protocol without degrading the amount of adversarial corruptions it is resilient to by more than t, or for most $i \in [n]$, the last time party i broadcasts is at least t rounds after the last time party $i-1$ broadcasts. As the latter implies the length of the protocol is at least $\Omega(nt)$, if $t = \sqrt{\Delta}$, any protocol of length at most $\mathcal{O}(n \cdot \sqrt{\Delta})$ must be in the former case. But, if this happens, we can keep removing rounds from the protocol, and eventually get a protocol

with no rounds that is resilient to a non-zero number of corruptions, a contradiction.

We wish to take $\Delta = \log n$. However, this by itself only shows that protocols with length at most $\mathcal{O}(n\sqrt{\log n})$ are resilient to at most $\mathcal{O}(\log n)$ adversarial corruptions, or have a failure probability of at least $n^{-\mathcal{O}(1)}$. We now *boost the failure probability* and show that there exist $\tilde{\Theta}(n)$ "disjoint" sets of corruptions that can fail the protocol. As disjoint corruptions are independent of each other, showing the existence of such sets boosts the failure probability to be almost 1.

1.5 Discussion and Future Directions

Our work suggests several directions for future work:

Simulating noiseless protocols of large length T. When $T \gg n$, our interactive coding scheme has a blowup of $\tilde{\Theta}(n)$ (see Theorem 4.1 for an exact statement). At a high level, this is because one can always get an adaptive simulation with a blowup of $\mathcal{O}(1)$ [EKS18], and simulate the adaptive simulation over the non-adaptive channel round by round with a blowup of $\tilde{\Theta}(n)$. It is interesting to see if this blowup can be avoided. Specifically, can we get an $\mathcal{O}(\operatorname{poly}\log n)$ blowup for every T? A good place to start may be to analyze the blowup needed by a protocol where the order in which the players broadcast is random.

Adaptive-to-adaptive simulation. Our work shows that the overhead of simulating a protocol in the non-adaptive noiseless broadcast model over the non-adaptive noisy broadcast channel is $\Theta(\sqrt{\log n})$, at least for polynomial length protocols. Prior work [EKS18] has shown that if one wants to simulate such a protocol over the *adaptive* noisy broadcast channel, then one can do this with a constant overhead. However, both works leave open the interesting question of simulating *adaptive* noiseless protocols over the adaptive channel. We mention that, as both the noiseless and the noisy protocol are more powerful, this question is incomparable to the current work.

Binary lower bound. Lastly, we mention that the proof of our lower bound (Theorem 1.2) makes use of the fact that the alphabet of the channel is polynomially large in the number of parties. This comes in because our lower bound analyses a *pointer-chasing* type problem where the input of each party is a large vector, but only one coordinate in the vector is relevant to the output (and the party does not know which). If *all* the parties know which is the relevant coordinate in advance, then a protocol like Gallager's [Gal88] would be applicable dashing all hopes of a lower bound. We exploit the large alphabet to easily establish that *none* of the parties can guess which is the right coordinate with any significant probability. This does simplify our argument substantially, but is not crucial to it, as even with a smaller alphabet (say a large constant), *most* of the parties will not be able to guess which is the relevant coordinate. Finding a way to formalize this is an interesting open question.

2 Overview of Our Protocol

We now overview our main result, highlighting the major ideas. We start by overviewing the upper bound in Section 2.1. The lower bound (which is more technically involved) is overviewed later in Section 2.2.

2.1 Upper Bound: Our Interactive Coding Scheme

For the purpose of this section, we shall assume that the channel uses a binary alphabet, the noise parameter $\epsilon = 0.1$, the length of the noiseless protocol being simulated is $T = n$, and for all $i \in [n]$, party i broadcasts in round i of the protocol. To start, observe that one can always break the protocol into $\frac{n}{k}$ *chunks* of k rounds each and simulate each chunk individually. Thus, if one can simulate k rounds of the noiseless protocol correctly except with probability polynomially small in n using at most k' rounds, then, by a union bound, one can also simulate the entire noiseless protocol correctly except with probability polynomially small in n using at most $\frac{n}{k} \cdot k'$ rounds.

We next show that any chunk of size $k = o(n)$ can be simulated correctly in $\tilde{\mathcal{O}}(k^2)$ rounds, except with probability exponentially small in k^2. Following the above observation, we choose $k = \Theta(\sqrt{\log n})$, so that the error probability will be polynomially small in n and we can union bound over all the chunks.

Simulating chunks of length $k = o(n)$. We seek a simulation scheme that takes a k-round noiseless protocol and converts it to an $\tilde{\mathcal{O}}(k^2)$-round noise resilient protocol with a *super-low* failure probability of $2^{-\Omega(k^2)}$. Indeed, note that this target failure probability is asymptotically close to the best possible, as the probability that any one party does not receive any of the messages correctly is at least $2^{-\tilde{\mathcal{O}}(k^2)}$. We also note that if the target failure probability was $2^{-\Omega(k)}$ (which we call *low*), instead of the super-low $2^{-\Omega(k^2)}$, then an $\mathcal{O}(k^2)$-round simulation can easily be obtained by repeating each round in the noiseless protocol $\mathcal{O}(k)$ times and having the parties decode by majority[9]. Putting it differently, we can easily obtain a simulation whose error probability is exponentially small in the *number of repetitions of a single symbol*. However, we wish to design a simulation with error that is exponentially small in the *length of the simulation*.

Low error simulation for non-interactive protocols. We first consider a very restricted set of protocols, which we call *non-interactive* protocols, where the symbol broadcast in any round of the protocol is independent of the symbols broadcast in the previous rounds. Implicit in Gallager's work is the following simulation scheme that achieves low error probability of $2^{-\Omega(k)}$ with only $\tilde{\mathcal{O}}(k)$ rounds (*cf.* $\mathcal{O}(k^2)$ rounds for the easy protocol above), and

[9] To see that this works, observe that, as the noise in the channel is independent across different rounds, the decoding for each player and each round will be correct except with probability $2^{-\Omega(k)}$. We can now union bound over all k parties and all k rounds and get that the overall simulation is correct except with probability $2^{-\Omega(k)}$.

we include its description here for completeness. His simulation consists of repeating every message only $\Theta(\log k)$ times and then having the parties jointly broadcast an $\mathcal{O}(k)$-length encoding of the transcript with an error correcting code, in a way that will be explained later.

The crucial property of non-interactive protocols that is exploited by this simulation is that the symbol broadcast in each round of the simulation is always "*correct*", in the sense that even if the transcript received by the communicating party is noisy, this party will broadcast the same symbol that it would have broadcast in the noiseless protocol, as the correct symbol does not depend on the received transcript.

Since every message is repeated $\Theta(\log k)$ times by the simulation, at the end of the execution, all parties know the correct transcript, except with probability $\frac{1}{\text{poly}(k)}$. Moreover, using the fact that the symbol broadcast by any party is always correct, the event that any given party does not know the correct transcript is independent of all other parties. As there are k parties, one can use standard concentration inequalities to conclude that, except with probability $2^{-\Omega(k)}$, at least 0.99 fraction of the parties have the correct transcript.

Had we known ahead of time of one specific party that is going to have the correct transcript (call this transcript X), then in order to get all parties to know X, we would just schedule this party to broadcast $C(X)$ at the end of the simulation, where C is a "good" error correcting code (say, C has distance and rate of $1/10$). Gallager's *hybrid trick* allows us to do something similar without knowing of any party with the correct X: Each party i encodes its received transcript, X_i, with C. Party 1 broadcasts symbols $1, \cdots, 10$ of $C(X_1)$, party 2 broadcasts symbols $11, \cdots, 20$ of $C(X_2)$, *etc.* All parties then receive a noisy hybrid of $C(X_1), \cdots, C(X_k)$. Since except with probability $2^{-\Omega(k)}$, 0.99 fraction of the parties have the correct transcript, it holds that except with probability $2^{-\Omega(k)}$, 0.99 fraction of the symbols broadcast for this hybrid will correspond to the encoding of the correct transcript. In turn, except with probability $2^{-\Omega(k)}$, 0.9 fraction of the symbols received for this hybrid by any party will correspond to $C(X)$, the encoding of the correct transcript, and the parties can all decode correctly.

From low error to super-low error for non-interactive protocols. Given a low error simulation with $\tilde{\mathcal{O}}(k)$ rounds, like the above, we can get a super-low error simulation (which is our goal) with $\tilde{\mathcal{O}}(k^2)$ rounds by repeating the low error simulation $\mathcal{O}(k)$ times independently and taking majority. Indeed, standard concentration inequalities say that a majority of the repetitions are correct except with probability $2^{-\Omega(k^2)}$.

"Reduction" to general protocols. Unfortunately, our low error simulation with $\tilde{\mathcal{O}}(k)$ rounds does not extend to general protocols. However, the following observation regarding the above low error simulation allows us to use it "as-is" to design our target super-low error simulation for general protocols. The observation is that at the end of an execution of the above low error simulation, all parties know the symbols *broadcast* by all other parties. (Of course, for general protocols, the broadcast symbols may not be the "correct" ones).

Equipped with this observation, let us consider what happens when we repeat the low error simulation to get a super-low error simulation as suggested above: The first out of the k parties did not need a transcript to compute the symbol it broadcasts. Thus, after the first execution of the low error protocol, all parties know the symbol of the first party, except with probability $2^{-\Omega(k)}$. Now, in the next repetition, all parties other than the first one can continue the simulation conditioned on the high probability event that the first symbol is correct, and a similar argument would show that, when this event indeed occurs, the second symbol is correct except with probability $2^{-\Omega(k)}$, and so on.

Interactive coding rewind-if-error mechanism. Thus, if all these high probability events occur, then after every execution of the low error simulation all parties learn the symbol of an additional party, and after k executions, the entire transcript of the noiseless protocol will be known to all[10]. However, our target failure probability is $2^{-\Omega(k^2)} \ll 2^{-\Omega(k)}$ and to achieve this super-low probability, we implement a simple *"rewind-if-error" interactive coding mechanism* inspired by Schulman's original work [Sch92], that allows a party whose symbol was received incorrectly in one of the executions to raise a flag during the subsequent executions[11] indicating that an error was made. When this happens, the parties "rewind" one iteration, erasing one incorrect symbol from their transcript. Otherwise, the parties add one correct symbol to the transcript.

Overall, one unit of "progress" is made in either case, except with probability $2^{-\Omega(k)}$. This implies that either the parties will make k units of progress after $\mathcal{O}(k)$ executions, which means that the transcript they output is correct, or there are $\Omega(k)$ executions where progress was not made, an event that can only happen with probability $2^{-\Omega(k^2)}$, as desired. Since the parties make roughly one unit of progress after executing an $\tilde{\mathcal{O}}(k)$-round scheme, the blowup of our scheme is $\tilde{\mathcal{O}}(k)$, which is shown to be unavoidable by the lower bound.

2.2 Lower Bound: Optimality of Our Scheme

We shall now show that our interactive coding scheme above is essentially optimal by showing a lower bound of $\Omega(\sqrt{\log n})$ on the overhead for any interactive coding scheme compiling protocols of length $T = n$. We note that this lower bound is much higher than Gallager's $\mathcal{O}(\log \log n)$ upper bound for the message exchange function [Gal88], and therefore we at least need a communication task that is significantly harder.

[10] We note that we cannot expect to learn (with high probability) the symbol of more than one party in a single execution of the low error simulation. For example, after the first execution, all parties know the symbol of the first party with high probability of $1 - 2^{-\Omega(k)}$. However, this only holds due to the hybrid trick at the end of this execution, and during the execution, after the first party repeats its symbol $\Theta(\log k)$ times, the second party only knows the first party's symbol with probability $1 - \frac{1}{\text{poly}(k)}$. Therefore, we cannot count on the second party to broadcast its correct symbol with high probability.

[11] We do not have a special symbol for this "flag" in our actual protocol. The party simply broadcasts the correct symbol, and the remaining parties check if it is the same as what they had in mind.

The hard communication task. Define $m = n^{200}$ to be a large enough polynomial function of n. In our *pointer-chasing* type communication task, party 1 will have as input a value $f_1 \in [m]$, while party i for all $1 < i \leq n$ will have function $f_i : [m] \to [m]$ mapping the set $[m]$ to itself. The goal of the parties is to output the value[12]:

$$\mathtt{PC}_{n,m} = f_n(f_{n-1}(\cdots(f_2(f_1))\cdots)).$$

Namely, the parties want to output the value of the composed function $f_n \circ \cdots \circ f_2$ on party 1's input f_1. A noiseless protocol over the broadcast channel with alphabet $[m]$ can easily compute $\mathtt{PC}_{n,m}$ in n rounds: In the first round, party 1 will broadcast its input f_1. Party 2 can use this value to broadcast $f_2(f_1)$ in the second round. Party 3 can then broadcast $f_3(f_2(f_1))$ in the third round, and so on until party n broadcasts $\mathtt{PC}_{n,m}$ in the last round.

Observe that the function $\mathtt{PC}_{n,m}$ is much "harder" than the message exchange function in the sense that the optimal noiseless protocol for $\mathtt{PC}_{n,m}$ is fully interactive: When the parties are computing the identity function, they know exactly what they have to broadcast ahead of time, whereas if the parties are computing the function $\mathtt{PC}_{n,m}$, only party 1 knows what it should broadcast ahead of time. All the other parties have as input a function, but only need its value on one of its m coordinates (and they do not know which). Moreover, as m is much larger than the total number of communication rounds, it is futile for any party to broadcast anything about its input unless it has some non-trivial information about which is the right coordinate[13].

A formula for failing a simulation round. Let Π be any protocol in the noisy broadcast channel that computes the function $\mathtt{PC}_{n,m}$ and let $T = \|\Pi\|$. As a randomized protocol is simply a distribution over deterministic protocols, we can assume Π to be deterministic without loss of generality. For the rest of this section, fix the inputs for the parties, and thus also the (noiseless) transcript of Π.

Let $\mathcal{T} \in [m]^{Tn}$. We think of $\mathcal{T}$ as a (possibly corrupted) transcript for Π (each of the n parties receives a symbol from $[m]$ in each of the T rounds). We say that $\mathcal{T}$ *can be obtained with t errors* if an *adversary* can corrupt t out of the Tn symbols received in Π and ensure that the transcript of the execution is $\mathcal{T}$. Let $r \in [T]$ and assume that party i broadcasts in round r of Π. We say that *round r fails with t corruptions* if there is a transcript $\mathcal{T}$ that can be obtained with at most t corruptions, and conditioned on the transcript that party i received for the first $r-1$ rounds being as in $\mathcal{T}$, each of the m options for the value $\mathtt{PC}_{i-1,m} = f_{i-1}(f_{i-2}(\cdots(f_2(f_1))\cdots))$ is (roughly) equally likely. In other words, when party i

[12]Our lower bound shall hold not only for *computing* the value $\mathtt{PC}_{n,m}$ but also for *estimating* any *single bit* in it with a non-negligible advantage. Thus, our lower bound result also holds for a large family of boolean functions.

[13]We mention that this is the only place where we use the fact that the alphabet of the protocol is a large polynomial. If it were smaller, it is possible that some parties can guess the right coordinate and/or broadcast the entire function over many rounds. While we do not believe that the function $\mathtt{PC}_{n,m}$ is easy for small m, assuming a large value of m significantly simplifies our already involved analysis.

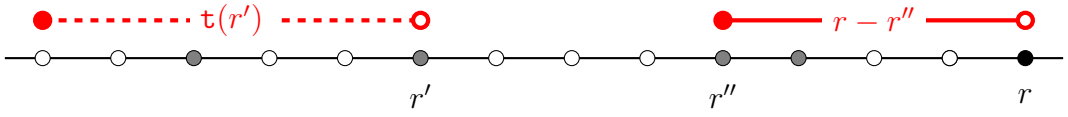

Figure 1: Round r is a round where a party $i > 1$ broadcasts. The gray rounds are rounds where party $i-1$ broadcasts. Failing any such gray round r' using $\mathtt{t}(r')$ corruptions, and then corrupting player i from the next gray round (denoted by r'') onwards suffices to fail round r. Thus, $\mathtt{t}(r) \leq \mathtt{t}(r') + r - r''$ for all r'. Observe that r' can be 0, in which case $\mathtt{t}(r') = 0$ and can also be the last gray round, in which case $r - r'' = 0$.

broadcasts in round r, it has (almost) no information about the coordinate it is supposed to apply f_i to. Finally, let $\mathtt{t}(r)$ be the *minimum number of corruptions required to fail round* r. (We set $\mathtt{t}(r) = \infty$ for all rounds r where party 1 broadcasts, as party 1 always knows f_1 and therefore no amount of corruptions will fail this round.)

We will next show that since the order of turns in Π is pre-specified and since no party can broadcast anything meaningful in Π till it knows which is the right coordinate, we can write a "clean" recursive formula for $\mathtt{t}(r)$. Let $r \in [T]$ be a round where some party $i > 1$ broadcasts. Let $r' < r$ be a round where party $i-1$ broadcasts, or 0. Let r'' be the first round after r' where party $i-1$ broadcasts, satisfying $r' < r'' < r$ (see Fig. 1). (If r' is the last round where party $i-1$ broadcasts before round r, we set $r'' = r$). The main observation here is that every transcript that fails round r' (and thus, party $i-1$ does not know its correct coordinate in round r') and also corrupts all the symbols received by party i in rounds r'' through r, also fails round r. Thus, we get the following formula for $\mathtt{t}(r)$[14]:

$$\mathtt{t}(r) = \min_{r'}\{\mathtt{t}(r') + r - r''\}. \tag{1}$$

A reduction to an adversarial model. As the symbol received by any party in any round is corrupted with some constant probability, say $\frac{1}{2}$, by the noisy broadcast channel, we have by definition of $\mathtt{t}(\cdot)$ that, for all rounds r, round r fails with probability at least $2^{-\mathtt{t}(r)}$. We shall show that if the protocol is short, then, for most of the parties, the last round r where they broadcast satisfies $\mathtt{t}(r) \leq \frac{\log n}{10}$. This directly implies that the protocol fails with probability at least $n^{-0.1}$ when it is run over the noisy broadcast channel. However, a lower bound of $n^{-0.1}$ on the failure probability is rather weak, and we now show how to *boost* this all the way to $1 - \frac{1}{n^{100}}$.

Error amplification. In a nutshell, we boost by showing that there are many different corrupted transcripts that fail round r instead of just one. In more detail, first note that if $\mathtt{t}(r) \leq \frac{\log n}{10}$, then there are at most $\frac{\log n}{10}$ parties whose sent symbols were affected by the noise. Consider now the exact same protocol but with the input of these $\frac{\log n}{10}$ parties fixed.

[14]Our above argument only shows an upper bound on $\mathtt{t}(r)$, but a matching lower bound can also be shown.

This does not change the length of the protocol and decreases the number of parties only marginally. Thus, we get that for this protocol with less parties, it is still (roughly) the case $\mathtt{t}(r) \leq \frac{\log n}{10}$, implying that there is another *disjoint* set of (roughly) $\frac{\log n}{10}$ corruptions that will fail the protocol. Continuing this way, we get that there are at least $\Theta\left(\frac{n}{\log n}\right)$ disjoint sets of corruptions that fail the protocol. As the noise of the noisy broadcast channel affects disjoint sets independently, we can conclude that the protocol fails with probability at least $1 - \frac{1}{n^{100}}$, as claimed.

A lower bound for the adversarial model. We still need to show that if the protocol is short, then, for most parties $i \in [n]$, the last round r_i where they broadcast satisfies $\mathtt{t}(r_i) \leq \frac{\log n}{10}$. We shall show this by showing a stronger form of the contrapositive, namely, that for all integers $t, \Delta > 0$, if $\mathtt{t}(r_i) > t \cdot \Delta$ for most parties $i \in [n]$, then the length of the protocol is larger than the minimum of $\frac{nt}{4}$ and $\frac{n\Delta}{4}$. Plugging $t = \Delta = \sqrt{\frac{\log n}{10}}$ shows the lower bound.

We show this by induction on t. The base case $t = 1$ is because a protocol needs at least n rounds to compute $\mathtt{PC}_{n,m}$. For the inductive step, assume that the rounds r_i form an increasing sequence, which can be shown to be without loss of generality. Define r'_i to be the second to last round where party i broadcasts. Assume that $\mathtt{t}(r) > t \cdot \Delta$ for most parties $i \in [n]$. Owing to Eq. (1) applied with $r = r_i$, $r' = r'_{i-1}$ and, by the fact that r_i form an increasing sequence, $r'' = r_{i-1}$, it holds that $\mathtt{t}(r'_{i-1}) + r_i - r_{i-1} > t \cdot \Delta$ for most parties $i \in [n]$. This implies that the protocol must satisfy at least one of the following conditions:

1. For at least a quarter of the values of i, we have $\mathtt{t}(r'_{i-1}) > (t-1) \cdot \Delta$.
2. For at least a quarter of the values of i, we have $r_i - r_{i-1} > \Delta$.

As mentioned in Section 1.4, these two cases will allow us to deduce the lower bound as they imply a *tradeoff between rounds and adversarial resilience*. More formally, if Item 2 holds, then since the monotonicity of r_i implies that the intervals $(r_{i-1}, r_i]$ are disjoint, the protocol is of length at least $\sum_{i \in [n]} r_i - r_{i-1} > \frac{n}{4} \cdot \Delta$, and we are done. Assume that Item 1 holds. Observe that Item 1 says that $\mathtt{t}(r'_{i-1})$, the number of corruptions needed to fail the *second to last* round where party $i-1$ broadcasts, is large for many i's (this is stronger than stating the same for the last round where party $i-1$ broadcasts). Thus, for all these parties, we can delete the last round where they broadcast from the protocol[15], and the induction hypothesis with $t-1$ can be applied to bound the length of the new protocol (*i.e.*, the protocol after the deletions where r'_{i-1} is the last time party $i-1$ broadcasts). We will get

[15] This needs to be done carefully, as deleting rounds from the protocol can affect the values of $\mathtt{t}(\cdot)$. In the actual proof, we merely "mark" these rounds and restrict r' and r'' in Eq. (1) to be unmarked. We show that if $\mathtt{t}(r) > (t-1) \cdot \Delta$ for some r before the rounds were marked, then the same holds after the rounds were marked and also change the induction hypothesis to be that either the number of unmarked rounds is at least $\frac{nt}{4}$ or the length of the protocol is at least $\frac{n}{4} \cdot \Delta$.

Such a careful analysis is also needed while "deleting" players in the error amplification step.

that the length of the protocol is larger than the minimum of $\frac{n(t-1)}{4}$ and $\frac{n}{4} \cdot \Delta$ plus the $\frac{n}{4}$ deleted rounds, finishing the proof.

3 Preliminaries and Formal Problem Definition

All logarithms in this work are to the base 2, unless specified otherwise. For an integer $n > 0$, the notation $[n]$ will denote the set $\{1, 2, \cdots, n\}$.

3.1 Technical Preliminaries

Throughout this subsection, we use sans-serif letters to denote random variables and reserve E to denote an arbitrary event. All random variables will be assumed to be discrete and we shall adopt the convention $0 \log \frac{1}{0} = 0$. All logarithms are taken with base 2.

3.1.1 Entropy

Definition 3.1 (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 3.2 (Conditional Entropy)**.** *We define the conditional entropy of* X *given* Y *and* E *as:*

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

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

Lemma 3.3 (Chain Rule for Entropy)**.** *It holds for all* X*,* Y*,* Z *and* E *that:*

$$\mathbb{H}(\mathsf{XY} \mid \mathsf{Z}, E) = \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) + \mathbb{H}(\mathsf{Y} \mid \mathsf{X}, \mathsf{Z}, E).$$

Proof. We have:

$$\begin{aligned}
\mathbb{H}(\mathsf{XY} \mid \mathsf{Z}, E) &= \sum_{z} \Pr(z \mid E) \cdot \mathbb{H}(\mathsf{XY} \mid z, E) \\
&= \sum_{z} \Pr(z \mid E) \cdot \sum_{x,y} \Pr(x, y \mid z, E) \cdot \log \frac{1}{\Pr(x, y \mid z, E)}
\end{aligned}$$

$$= \sum_z \Pr(z \mid E) \cdot \sum_{x,y} \Pr(x, y \mid z, E) \cdot \left(\log \frac{1}{\Pr(x \mid z, E)} + \log \frac{1}{\Pr(y \mid x, z, E)} \right)$$
$$= \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) + \sum_{x,z} \Pr(x, z \mid E) \cdot \sum_y \Pr(y \mid x, z, E) \cdot \log \frac{1}{\Pr(y \mid x, z, E)}$$
$$= \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) + \mathbb{H}(\mathsf{Y} \mid \mathsf{X}, \mathsf{Z}, E). \qquad \square$$

Lemma 3.4 (Conditioning reduces Entropy)**.** *It holds for all* X*,* Y*,* Z *and* E *that:*

$$\mathbb{H}(\mathsf{X} \mid \mathsf{Y}, \mathsf{Z}, E) \leq \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E).$$

Equality holds if and only if X *and* Y *are independent conditioned on* Z*,* E*.*

Proof. We have:

$$\begin{aligned}
\mathbb{H}(\mathsf{X} \mid \mathsf{Y}, \mathsf{Z}, E) &= \sum_{y,z} \Pr(y, z \mid E) \cdot \mathbb{H}(\mathsf{X} \mid \mathsf{Y} = y, \mathsf{Z} = z, E) \\
&= \sum_{x,y,z} \Pr(y, z \mid E) \cdot \Pr(x \mid y, z, E) \cdot \log \frac{1}{\Pr(x \mid y, z, E)} \\
&= \sum_{x,y,z} \Pr(x, z \mid E) \cdot \Pr(y \mid x, z, E) \cdot \log \frac{\Pr(y, z \mid E)}{\Pr(x, z \mid E) \cdot \Pr(y \mid x, z, E)} \\
&\leq \sum_{x,z} \Pr(x, z \mid E) \cdot \log \frac{\Pr(z \mid E)}{\Pr(x, z \mid E)} \qquad \text{(Concavity of } \log(\cdot)\text{)} \\
&= \sum_z \Pr(z \mid E) \cdot \sum_x \Pr(x \mid z, E) \cdot \log \frac{1}{\Pr(x \mid z, E)} \\
&= \sum_z \Pr(z \mid E) \cdot \mathbb{H}(\mathsf{X} \mid \mathsf{Z} = z, E) \\
&= \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E). \qquad \square
\end{aligned}$$

Lemma 3.5. *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* $\mathsf{supp}(\mathsf{X})$*.*

Proof. The first inequality is direct. For the second, we have by the concavity of $\log(\cdot)$ that:

$$\mathbb{H}(\mathsf{X} \mid E) = \sum_x \Pr(x \mid E) \cdot \log \frac{1}{\Pr(x \mid E)} \leq \log(|\mathsf{supp}(\mathsf{X})|). \qquad \square$$

Lemma 3.6. *It holds for all* X, Y, Z *and* E *that:*

$$\mathbb{H}(\mathsf{X} \mid \mathsf{Y}, \mathsf{Z}, E) \geq \mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) - \mathbb{H}(\mathsf{Y} \mid \mathsf{Z}, E).$$

Proof. Using Lemma 3.5 and Lemma 3.3, conclude that $\mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) \leq \mathbb{H}(\mathsf{XY} \mid \mathsf{Z}, E)$. Use Lemma 3.3 again to get:

$$\mathbb{H}(\mathsf{X} \mid \mathsf{Z}, E) \leq \mathbb{H}(\mathsf{Y} \mid \mathsf{Z}, E) + \mathbb{H}(\mathsf{X} \mid \mathsf{Y}, \mathsf{Z}, E).$$

Rearranging yields the lemma. □

3.1.2 KL Divergence

Definition 3.7 (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.

Lemma 3.8. *Let* X *be a random variable uniformly distributed over a set* Ω *and* $S \subseteq \Omega$ *be given:*

$$\mathbb{D}(\mathsf{dist}(\mathsf{X} \mid \mathsf{X} \in S) \mid\mid \mathcal{U}) = \log \frac{|\Omega|}{|S|}.$$

Proof. As X is distributed uniformly, we have:

$$\mathbb{D}(\mathsf{dist}(\mathsf{X} \mid \mathsf{X} \in S) \mid\mid \mathcal{U}) = \sum_{x \in S} \frac{1}{|S|} \cdot \log \frac{|\Omega|}{|S|} = \log \frac{|\Omega|}{|S|}.$$ □

Lemma 3.9. *It holds for all* X *and* E *that:*

$$\mathbb{D}(\mathsf{dist}(\mathsf{X} \mid E) \mid\mid \mathcal{U}) = \log(|\mathsf{supp}(\mathsf{X})|) - \mathbb{H}(\mathsf{X} \mid E).$$

Proof. We have:

$$\begin{aligned}
\mathbb{D}(\mathsf{dist}(\mathsf{X} \mid E) \mid\mid \mathcal{U}) &= \sum_{x \in \mathsf{supp}(\mathsf{X})} \Pr(x \mid E) \cdot \log(\Pr(x \mid E) \cdot |\mathsf{supp}(\mathsf{X})|) \\
&= \sum_{x \in \mathsf{supp}(\mathsf{X})} \Pr(x \mid E) \cdot \log \Pr(x \mid E) + \sum_{x \in \mathsf{supp}(\mathsf{X})} \Pr(x \mid E) \cdot \log(|\mathsf{supp}(\mathsf{X})|) \\
&= \log(|\mathsf{supp}(\mathsf{X})|) - \mathbb{H}(\mathsf{X} \mid E).
\end{aligned}$$ □

Lemma 3.10. *It holds for all* X, Y *and* E *that:*

$$\mathbb{D}(\mathsf{dist}(\mathsf{XY} \mid E) \mid\mid \mathcal{U}) \geq \mathbb{D}(\mathsf{dist}(\mathsf{X} \mid E) \mid\mid \mathcal{U}) + \mathbb{D}(\mathsf{dist}(\mathsf{Y} \mid E) \mid\mid \mathcal{U}).$$

Proof. We have:

$$\begin{aligned}
\mathbb{D}(\mathsf{dist}(\mathsf{XY} \mid E) \mid\mid \mathcal{U}) &= \log(|\mathsf{supp}(\mathsf{X})|) + \log(|\mathsf{supp}(\mathsf{Y})|) - \mathbb{H}(\mathsf{XY} \mid E) && \text{(Lemma 3.9)} \\
&\geq \log(|\mathsf{supp}(\mathsf{X})|) + \log(|\mathsf{supp}(\mathsf{Y})|) - \mathbb{H}(\mathsf{X} \mid E) - \mathbb{H}(\mathsf{Y} \mid E) \\
& && \text{(Lemma 3.3 and Lemma 3.4)} \\
&\geq \mathbb{D}(\mathsf{dist}(\mathsf{X} \mid E) \mid\mid \mathcal{U}) + \mathbb{D}(\mathsf{dist}(\mathsf{Y} \mid E) \mid\mid \mathcal{U}). && \text{(Lemma 3.9)}
\end{aligned}$$

□

3.1.3 Total Variation Distance

Definition 3.11 (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 3.12 (Pinsker's inequality)**.** *Let* μ, ν *be two distributions over the same set* Ω*. It holds that:*

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

Corollary 3.13. *Let* X*,* E *be given and consider a set* $S \subseteq \mathsf{supp}(\mathsf{X})$*. It holds that:*

$$\left| \Pr(\mathsf{X} \in S \mid E) - \frac{|S|}{|\mathsf{supp}(\mathsf{X})|} \right| \leq \sqrt{\frac{1}{2} \cdot \mathbb{D}(\mathsf{dist}(\mathsf{X} \mid E) \mid\mid \mathcal{U})}.$$

3.2 String Operations

Throughout this section, let Γ be some non-empty alphabet set.

For two strings $x, x' \in \Gamma^*$, we denote by $x \| x'$ their concatenation, and by $\|x\|$ the length of x. Furthermore, if $\|x\| = \|x'\|$, we denote their Hamming distance by

$$\Delta(x, x') = |\{i \in [\|x\|] : x_i \neq x'_i\}|.$$

For a string $x \in \Gamma^*$ and $j \leq \|x\|$, we denote by $x_{\leq j}$ the string $x_1 \| x_2 \| \cdots \| x_j$ and by $x_{<j}$ the string $x_1 \| x_2 \| \cdots \| x_{j-1}$. For $j < j' \leq \|x\|$, we denote by $x_{(j,j']}$ the string $x_{j+1} \| x_{j+2} \| \cdots x_{j'}$.

For two strings $x, x' \in \Gamma^*$, we denote by $\mathsf{LCP}(x, x')$ their longest common prefix. More precisely, let $\ell = \max\big(\{0 \leq j \leq \min(|x|, |x'|) : x_{\leq j} = x'_{\leq j}\}\big)$. Then,

$$\mathsf{LCP}(x, x') = x_{\leq \ell}.$$

3.3 Concentration Bounds

We shall use the following version of the Chernoff bound.

Lemma 3.14 (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,*

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

In particular, we have that:

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

3.4 Error Correcting Codes

We shall use the following types of codes for our algorithms.

Definition 3.15 (Error-Correcting Codes)**.** *Fix $\delta \in (0, 1/2)$, $k, m \in \mathbb{Z}^+$, and a non-empty alphabet Γ. A (δ, Γ, k, m)-error correcting code is a function $\mathsf{ECC} : \Gamma^k \to \Gamma^{mk}$ with the property that for all $z, z' \in \Gamma^k$, if $z \neq z'$, then*

$$\Delta(\mathsf{ECC}(z), \mathsf{ECC}(z')) \geq \delta mk.$$

These codes are useful because noisy versions of these codes can still be decoded correctly with high probability, as will be shown later.

We also consult the literature to show that such codes do exist.

Lemma 3.16. *For all Γ with $|\Gamma| \geq 2$ and for all sufficiently large k, there exists a $(0.4, \Gamma, k, 10^4)$-error correcting code.*

Proof. Let $f : \Gamma^k \to \Gamma^{10^4 k}$ be a random function. We wish to prove that f is a $(0.4, \Gamma, k, 10^4)$-error correcting code with non-zero probability. For $z, z' \in \Gamma^k$, let $\mathcal{X}_{z,z'}$ be the event that $\Delta(f(z), f(z')) < 0.4 \cdot 10^4 k$. We can then express the event that f is not such a code as

$$\Pr\big(f \text{is not a } (0.4, \Gamma, k, 10^4)\text{-error correcting code}\big) = \Pr\left(\bigcup_{z\in\Gamma^k}\bigcup_{\substack{z'\in\Gamma^k \\ z'\neq z}} \mathcal{X}_{z,z'}\right)$$

$$\leq \sum_{z\in\Gamma^k} \Pr\left(\bigcup_{\substack{z'\in\Gamma^k\\ z'\neq z}} \mathcal{X}_{z,z'}\right)$$
$$\leq \sum_{z\in\Gamma^k} \sum_{\substack{z'\in\Gamma^k\\ z'\neq z}} \Pr(\mathcal{X}_{z,z'}).$$

We thus have to analyze $\Pr\left(\mathcal{X}^{z,z'}\right)$ for arbitrary z and z'. Fix z and z' such that $z \neq z'$. For $i \in [10^4 k]$, let $\mathcal{Z}_i$ be the event that $f_i(z) = f_i(z')$. Note that each $\mathcal{Z}_i$ is independent for different i, and occurs with probability $|\Gamma|^{-1}$. Furthermore, note that $\mathcal{X}_{z,z'}$ occurs if and only if at least $0.6 \cdot 10^4 k$ of the $\mathcal{Z}_i$ occur.

Now, we will bound the probability $\Pr(\mathcal{X}_{z,z'})$ under two separate cases, depending on $|\Gamma|$.

Case 1: $|\Gamma| \leq 3$**.** Let $\mathbb{1}[\mathcal{Z}_i]$ be an indicator random variable for $\mathcal{Z}_i$. The probability of $\mathcal{X}_{z,z'}$ occuring is then bounded as follows:

$$\begin{aligned}
\Pr(\mathcal{X}_{z,z'}) &= \Pr\left(\sum_{i=1}^{10^4 k} \mathbb{1}[\mathcal{Z}_i] \geq 0.6 \cdot 10^4 k\right) \\
&= \Pr\left(\sum_{i=1}^{10^4 k} \mathbb{1}[\mathcal{Z}_i] \geq 0.6|\Gamma| \cdot |\Gamma|^{-1} 10^4 k\right) \\
&\leq \mathrm{e}^{-\frac{(0.6|\Gamma|-1)^2 \cdot |\Gamma|^{-1} 10^4 k}{2+(0.6|\Gamma|-1)}} && \text{(Lemma 3.14)} \\
&\leq \mathrm{e}^{-\frac{(0.2|\Gamma|)^2 \cdot |\Gamma|^{-1} 10^4 k}{3}} && (2 \leq |\Gamma| \leq 3) \\
&= \mathrm{e}^{-\frac{400}{3}|\Gamma|k} \\
&\leq |\Gamma|^{-100k} && (\mathrm{e}^x \geq x) \\
&\leq |\Gamma|^{-50k}.
\end{aligned}$$

Case 2: $|\Gamma| \geq 4$**.** Consider $\mathcal{X}_{z,z'}$. Note that this occurs if and only if there exists some set S of size at least $0.6 \cdot 10^4 k$ such that there $\mathcal{Z}_i$ is true for all $i \in S$. We then get the following:

$$\begin{aligned}
\Pr(\mathcal{X}_{z,z'}) &= \Pr\left(\bigcup_{\substack{S\subseteq[10^4 k]\\ |S|=0.6\cdot 10^4 k}} \bigcap_{i\in S} \mathcal{Z}_i\right) \\
&\leq \sum_{\substack{S\subseteq[10^4 k]\\ |S|=0.6\cdot 10^4 k}} \Pr\left(\bigcap_{i\in S} \mathcal{Z}_i\right)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{\substack{S \subseteq [10^4 k] \\ |S| = 0.6 \cdot 10^4 k}} \prod_{i \in S} \Pr(\mathcal{Z}_i) \\
&= \sum_{\substack{S \subseteq [10^4 k] \\ |S| = 0.6 \cdot 10^4 k}} \prod_{i \in S} |\Gamma|^{-1} \\
&= \sum_{\substack{S \subseteq [10^4 k] \\ |S| = 0.6 \cdot 10^4 k}} |\Gamma|^{-0.6 \cdot 10^4 k} \\
&\leq \sum_{S \subseteq [10^4 k]} |\Gamma|^{-0.6 \cdot 10^4 k} \\
&= 2^{10^4 k} |\Gamma|^{-0.6 \cdot 10^4 k} \\
&\leq |\Gamma|^{0.5 \cdot 10^4 k} |\Gamma|^{-0.6 \cdot 10^4 k} && (|\Gamma| \geq 4) \\
&= |\Gamma|^{-10^3 k} \\
&\leq |\Gamma|^{-50k}.
\end{aligned}
$$

Thus, in both cases, we see that $\Pr(\mathcal{X}_{z,z'}) \leq |\Gamma|^{-50k}$. The rest of the proof continues identically for both cases. We thus return to our earlier work to get that:

$$
\begin{aligned}
\Pr\big(f \text{is not a } (0.4, \Gamma, k, 10^4)\text{-error correcting code}\big) &\leq \sum_{z \in \Gamma^k} \sum_{\substack{z' \in \Gamma^k \\ z' \neq z}} \Pr(\mathcal{X}_{z,z'}) \\
&\leq \sum_{z \in \Gamma^k} \sum_{\substack{z' \in \Gamma^k \\ z' \neq z}} |\Gamma|^{-50k} \\
&\leq |\Gamma|^k |\Gamma|^k |\Gamma|^{-50k} \\
&= |\Gamma|^{-25k}.
\end{aligned}
$$

Thus, with probability at least $1 - 2^{-25k} > 0$, for any two distinct $z, z' \in \Gamma^k$, $\Delta(f(z), f(z')) \geq 0.4 \cdot 10^4 k$. Thus, we see that with non-zero probability, f is a $(0.4, \Gamma, k, 10^4)$-error correcting code, so there must exist such an error correcting code. □

We also state how to decode error correcting codes.

Definition 3.17. *Fix $\delta \in (0, 1/2)$, $k, m \in \mathbb{Z}^+$, and an alphabet Γ. Given a (δ, Γ, k, m)-error correcting code $\mathsf{ECC} : \Gamma^k \to \Gamma^{mk}$, define $\mathsf{D\text{-}ECC} : \Gamma^{mk} \to \Gamma^k$ to be the function given by*

$$
\mathsf{D\text{-}ECC}(w) = \arg\min_{z' \in \Gamma^k} \Delta(\mathsf{ECC}(z'), w),
$$

with ties broken lexicographically.

We then claim that this decoding function works correctly in the presence of few corruptions.

Lemma 3.18. *Fix $\delta \in (0, 1/2)$, $k, m \in \mathbb{Z}^+$, and an alphabet Γ. Let $\mathsf{ECC} : \Gamma^k \to \Gamma^{mk}$ be a (δ, Γ, k, m)-error correcting code. Then, for any $z \in \Gamma^k$ and $w \in \Gamma^{mk}$ where $\Delta(\mathsf{ECC}(z), w) < \frac{\delta}{2}mk$, $\mathsf{D\text{-}ECC}(w) = z$.*

Proof. Fix z and w as above. Consider any $z' \neq z$. By Definition 3.15, $\Delta(\mathsf{ECC}(z), \mathsf{ECC}(z')) \geq \delta mk$. Then, we can use a triangle inequality to see that

$$\begin{aligned}\Delta(\mathsf{ECC}(z'), w) &\geq \Delta(\mathsf{ECC}(z), \mathsf{ECC}(z')) - \Delta(\mathsf{ECC}(z), w) \\ &> \delta mk - \frac{\delta}{2}mk \\ &= \frac{\delta}{2}mk \\ &> \Delta(\mathsf{ECC}(z), w).\end{aligned}$$

Thus, for any $z' \neq z$, $\Delta(\mathsf{ECC}(z'), w) > \Delta(\mathsf{ECC}(z), w)$. As such, we see that $\mathsf{D\text{-}ECC}(w) = \arg\min_{z' \in \Gamma^k} \Delta(\mathsf{ECC}(z'), w) = z$. □

3.5 The Noisy Broadcast Model

The noisy broadcast model is defined by a number $n > 0$ of parties, a noise parameter $\epsilon \in [0, 1]$, and a finite, non-empty alphabet set Γ. When the noise parameter $\epsilon = 0$, we say that the broadcast model is *noiseless* and may drop ϵ from the notation. A (deterministic) protocol over the (n, ϵ, Γ)-noisy broadcast model is defined by a tuple:

$$\Pi = \big(T, \mathtt{p}, \mathcal{X}^1, \cdots, \mathcal{X}^n, \mathcal{Y}, M_1, \cdots, M_T, \mathtt{out}\big), \tag{2}$$

where: (1) $T = \|\Pi\| > 0$ is a parameter denoting the length of the protocol, (2) $\mathtt{p} : [T] \to [n]$ is a function that determines which party speaks when (*i.e.*, for all $j \in [T]$, party $\mathtt{p}(j)$ is the unique party speaking in round j), (3) $\mathcal{X}^i$ for all $i \in [n]$ is the input set of party i, (4) $\mathcal{Y}$ is the set of possible outputs of the protocol, (5) For all $j \in [T]$, $M_j : \mathcal{X}^{\mathtt{p}(j)} \times \Gamma^{j-1} \to \Gamma$ is a function that computes the message sent in round j based on the input of the party $\mathtt{p}(j)$ speaking in round j and the transcript $\in \Gamma^{j-1}$ received by party $\mathtt{p}(j)$ in the first $j-1$ rounds, (6) $\mathtt{out} : \Gamma^T \to \mathcal{Y}$ is a function that computes the output from the transcript of the protocol. We suppress items on the right hand side of Eq. (2) when they are clear from context. We define a randomized protocol to be a distribution over deterministic protocols.

Execution of a protocol. A protocol Π as defined above starts with all parties $i \in [n]$ having an input $x^i \in \mathcal{X}^i$ and proceeds in T rounds, maintaining the invariant that before round $j \in [T]$, all parties $i \in [n]$ have a transcript $\Pi^i_{<j} \in \Gamma^{j-1}$. For all $j \in [T]$, in round

j, party $\mathtt{p}(j)$ sends the symbol $\Pi_j = M_j\left(x^{\mathtt{p}(j)}, \Pi^{\mathtt{p}(j)}_{<j}\right)$. All parties $i \in [n]$ then receive an independent noisy copy Π^i_j of Π_j that is equal to Π_j with probability $1-\epsilon$ and a uniformly random symbol from Γ with probability ϵ. They append Π^i_j to $\Pi^i_{<j}$ to obtain a transcript $\Pi^i_{\leq j} = \Pi^i_{<j+1}$ and continue executing the protocol. After T rounds are over, party 1 outputs $\mathtt{out}(\Pi^1_{\leq T}) \in \mathcal{Y}$[16].

Observe that the execution of the protocol is determined by the inputs $X = (x^1, x^2, \cdots, x^n)$ and the noise in the channel. For $i \in [n]$ and $j \in [T]$, we use $\mathsf{\Pi}^i_j(X)$ (respectively, $\mathsf{\Pi}_j(X)$ and $\mathsf{\Pi}(X)$) to denote the random variable (over the randomness in the channel noise) corresponding to the value of Π^i_j (resp, Π_j and the output) in the execution of the protocol Π when the inputs are given by X. We define $\mathsf{\Pi}^i_{\leq j}(X) = \left(\mathsf{\Pi}^i_1(X), \cdots, \mathsf{\Pi}^i_j(X)\right)$ and $\mathsf{\Pi}_{\leq j}(X) = (\mathsf{\Pi}_1(X), \cdots, \mathsf{\Pi}_j(X))$, and $\mathsf{\Pi}^i_{<j}(X), \mathsf{\Pi}_{<j}(X)$ are defined similarly.

Channel noise. We now define some helpful notation regarding the noise in the channel. Consider an execution of a protocol Π as above. For a party $i \in [n]$ in round $j \in [T]$, we denote the noise witnessed by party i in round j by N^i_j. Recall that party i can either receive the symbol sent (by party $\mathtt{p}(j)$) in round j correctly, and event we denote by $N^i_j = \star$, or can receive a uniformly random symbol from Γ, when we set N^i_j to be the symbol received. Using this notation, we get that $\left\{N^i_j\right\}_{i\in[n], j\in[T]}$ are mutually independent and for all $i \in [n], j \in [T]$:

$$N^i_j = \begin{cases} \star, & \text{with probability } 1-\epsilon \\ \gamma, & \forall \gamma \in \Gamma \text{ with probability } \frac{\epsilon}{|\Gamma|} \end{cases} \tag{3}$$

We shall use N to denote the tuple $N = \left(N^i_j\right)_{i\in[n], j\in[T]}$.

4 The Upper Bound

In this section, we prove the upper bound part of our results. We begin by giving the formal statement of Theorem 1.1. Most importantly, we prove a stronger result than stated earlier, getting stronger bounds on the length of the resulting protocol.

We will prove this statement for all deterministic protocols. As a randomized protocol is a distribution over deterministic protocols, this thus proves it for all randomized protocols as well. Without loss of generality, we also assume that the output of any noiseless protocol is simply the transcript that was observed in that protocol.

Theorem 4.1. *Let $n > 0$, $\epsilon \in [0, 1/2]$, and Γ be an non-empty set. Let Π be a protocol over the (n, Γ)-noisy broadcast model of length $\|\Pi\| = T$. Then, there exists a protocol Π' over the*

[16] Having only party 1 output (instead of all parties outputting) is without loss of generality, as party 1 can simply encode its output using an error correcting code and broadcast the encoding. This would allow, except with small probability, all the parties who broadcast at least once in the protocol to know the output correctly.

(n, ϵ, Γ)-*noisy broadcast model such that* $\|\Pi'\| \leq T \cdot \tilde{\mathcal{O}}\big(\min\{\sqrt{\log T}, n\}\big)$ *and for all inputs* $X \in \mathcal{X}^1 \times \cdots \mathcal{X}^n$*, we have:*

$$\Pr(\Pi'(X) \neq \Pi(X)) \leq 1/T^2,$$

where the probability is over the noise in the channel.

The crux of this theorem is captured within the following smaller result, which we prove first. We defer proving Theorem 4.1 until later (in Section 4.1).

Theorem 4.2. *Let* $n \geq 3k > 0$ *be large enough and* Γ *be an arbitrary non-empty set. Let* Π *be a protocol over the* (n, Γ)*-noisy broadcast model of length* $\|\Pi\| = k$*. For all constant* $\epsilon \in [0, 1/2]$*, there exists a protocol* Π' *over* (n, ϵ, Γ)*-noisy broadcast model such that* $\|\Pi'\| \leq 10^{25} \cdot k^2 \log k$ *and for all inputs* $X \in \mathcal{X}^1 \times \cdots \mathcal{X}^n$*, we have:*

$$\Pr(\Pi'(X) \neq \Pi(X)) \leq 2^{-k^2},$$

where the probability is over the noise in the channel.

Proof. We assume $|\Gamma| > 1$ without loss of generality. We shall define the protocol Π' over the $(n, \frac{1}{10}, \Gamma)$-noisy broadcast model with $\|\Pi'\| \leq 10^{20} \cdot k^2 \log k$. This then implies Theorem 4.2 due to standard noise reduction techniques. Our protocol shall use a $(0.4, \Gamma, k, 10^4)$-error correcting code ECC, as given by Lemma 3.16.

As $\|\Pi\| = k \leq n/3$, there are at least $2k$ parties that do not participate in Π. Out of these non-participating parties, we select one leader Ld and k repeaters, numbered from 1 to k[17]. We assume without loss of generality that Ld is the party that outputs in Π.

We describe our protocol formally in Algorithm 1 that uses Algorithm 2 as a subroutine.

The claim about $\|\Pi'\|$ is straightforward from these definitions, and we focus on showing that Π' indeed simulates Π.

Fix $X \in \mathcal{X}^1 \times \cdots \mathcal{X}^n$. For a variable *var* in Algorithms 1 and 2 and $t \in [10^5 k]$, we shall use $var(t)$ to denote the value of *var* at the end of iteration t. We shall use $t = 0$ to denote the values at the start of the protocol.

For every $t \in [10^5 k]$, we define the following events for the purposes of our analysis:

- For $r \in [k]$, let $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}(t)$ be the event that during the call to SHARE in iteration t, there exists some $j \in [k]$ such that at least $1/2$ of the broadcasts from player $\mathsf{p}(j)$ to repeater r at Line 14 are affected by noise.
- We define $\mathcal{E}_{\mathsf{sh\text{-}Rpt}}(t)$ to be the event that at least $0.05k$ of the k different $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}(t)$ occurred.
- We define $\mathcal{E}_{\mathsf{sh\text{-}Ld}}(t)$ to be the event that during the call to SHARE in iteration t, at least 0.15 fraction of the broadcasts from the repeaters to the leader at Line 15 are affected by noise.

[17] These repeaters do not have to be distinct from the players, but the algorithm becomes clearer when they are explicitly listed as distinct.

Algorithm 1 The protocol Π'.

Input: All parties $i \in [n]$ have an input $x^i \in \mathcal{X}^i$.

Output: Ld outputs a transcript $\pi^{\mathsf{Ld}} \in \Gamma^k$.

Ld initializes π^{Ld} as the empty string.
for $t \in [10^5 k]$ **do**
    Ld broadcasts $\mathsf{ECC}(\pi^{\mathsf{Ld}})$ (after trimming or padding π^{Ld} to be in Γ^k). For $i \in [n]$, let $\widetilde{\mathsf{ECC}}^i$ be the message received by party i.
    All parties $i \in [n]$ set $\tilde{\pi}^i \leftarrow \mathsf{D\text{-}ECC}\Big(\widetilde{\mathsf{ECC}}^i\Big)$.
    For $j \in [k]$, party $\mathsf{p}(j)$ sets $z^j \leftarrow M_j\Big(x^{\mathsf{p}(j)}, \tilde{\pi}^{\mathsf{p}(j)}_{<j}\Big)$.
    The parties run $\textsc{Share}(z^1, \cdots, z^k)$. Let $\tilde{z}^{\mathsf{Ld}} \leftarrow \big\{\tilde{z}^{\mathsf{Ld}}_j\big\}_{j\in[k]}$ denote Ld's output.
    if $\Big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}, \tilde{z}^{\mathsf{Ld}}\big)\Big\| \geq \min\big(k, \big\|\pi^{\mathsf{Ld}}\big\|\big)$ **then** ▷ The remainder is executed by Ld.
        Set $\pi^{\mathsf{Ld}} \leftarrow \pi^{\mathsf{Ld}} \| \tilde{z}^{\mathsf{Ld}}_{\|\pi^{\mathsf{Ld}}\|+1}$. Let $\tilde{z}^{\mathsf{Ld}}_{\|\pi^{\mathsf{Ld}}\|+1}$ be an arbitrary symbol in Γ if $\big\|\pi^{\mathsf{Ld}}\big\| \geq k$.
    else
        Set $\pi^{\mathsf{Ld}} \leftarrow \pi^{\mathsf{Ld}}_{<\|\pi^{\mathsf{Ld}}\|}$.
    end if
end for
Ld outputs π^{Ld} trimmed (or padded) to be in Γ^k.

Algorithm 2 The subroutine $\textsc{Share}(z^1, \cdots, z^k)$.

Input: For all $j \in [k]$, party $\mathsf{p}(j)$ has a symbol $z^j \in \Gamma$.

Output: For all $j \in [k]$, Ld outputs $\tilde{z}^{\mathsf{Ld}}_j \in \Gamma$.

For all $j \in [k]$, party $\mathsf{p}(j)$ broadcasts z^j for $200 \log k$ rounds. Repeaters $r \in [k]$ decode by majority to get $\tilde{z}^r_j$.
Repeaters $r \in [k]$ set $\tilde{z}^r \leftarrow \tilde{z}^r_1 \| \cdots \| \tilde{z}^r_k$ and broadcast $\mathsf{ECC}_{(10^4(r-1),10^4 r]}(\tilde{z}^r)$. Let $\widetilde{\mathsf{ECC}}_{(10^4(r-1),10^4 r]}$ be the symbols received by Ld.
Leader sets $\widetilde{\mathsf{ECC}} \leftarrow \widetilde{\mathsf{ECC}}_{(0,10^4]} \| \widetilde{\mathsf{ECC}}_{(10^4, 10^4\cdot 2]} \| \cdots \| \widetilde{\mathsf{ECC}}_{(10^4(k-1),10^4 k]}$ and outputs $\mathsf{D\text{-}ECC}\Big(\widetilde{\mathsf{ECC}}\Big)$.

- We define $\mathcal{E}_{\mathsf{sh}}(t)$ as $\mathcal{E}_{\mathsf{sh\text{-}Rpt}}(t) \cup \mathcal{E}_{\mathsf{sh\text{-}Ld}}(t)$.
- We define $\mathcal{E}_{\mathsf{ecc}}(t)$ as the event that there exists some $j \in [k]$ such that at least 0.2 fraction of the broadcasts from the leader to player $\mathtt{p}(j)$ at Line 3 in iteration t are affected by noise.
- Finally, we define $\mathcal{E}(t)$ as $\mathcal{E}_{\mathsf{sh}}(t) \cup \mathcal{E}_{\mathsf{ecc}}(t)$.

We now begin to analyse these events. First, note that for all $t \in [10^5 k]$, $\mathcal{E}_{\mathsf{sh}}(t)$ depends entirely on the noise in the channel during iteration t. In particular, this means that $\mathcal{E}_{\mathsf{sh}}(t)$ and $\mathcal{E}_{\mathsf{sh}}(t')$ are independent for all $t \neq t'$. Furthermore:

Lemma 4.3. *For all $t \in [10^5 k]$, we have:*

$$\Pr(\mathcal{E}_{\mathsf{sh}}(t)) \leq 2^{-5k}.$$

Proof. Fix $t \in [10^5 k]$. For the rest of the proof, all variables are taken to have their values during the execution in round t.

For all $j \in [k]$, we let $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt},j}$ be the event that at least half the broadcasts from player $\mathtt{p}(j)$ at Line 14 are affected by noise for repeater r. Thus, $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}} = \mathcal{E}^r_{\mathsf{sh\text{-}Rpt},1} \cup \cdots \cup \mathcal{E}^r_{\mathsf{sh\text{-}Rpt},k}$. Note that because this event only depends on the noise while player $\mathtt{p}(j)$ broadcasts, these $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt},j}$ are distinct.

Fix $r \in [k]$ and $j \in [k]$. For $i \in [200 \log k]$, let $N^r_{j,i}$ be an indicator random variable for if the ith broadcast at Line 14 from player $\mathtt{p}(j)$ is affected by noise for repeater r. Note that $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt},j}$ is exactly the event that $\sum_{i=1}^{200 \log k} N^r_{j,i} \geq 100 \log k$. Furthermore, note that the distributions of $N^r_{j,i}$ are independent, and that $\mathbb{E}\left[N^r_{j,i}\right] = 1/10$. This thus bounds the probability of $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt},j}$ by

$$\begin{aligned}
\Pr\left(\mathcal{E}^r_{\mathsf{sh\text{-}Rpt},j}\right) &= \Pr\left(\sum_{i=1}^{200 \log k} N^r_{j,i} \geq 100 \log k\right) \\
&= \Pr\left(\sum_{i=1}^{200 \log k} N^r_{j,i} \geq (1+4) \cdot 0.1 \cdot 200 \log k\right) \\
&\leq \mathrm{e}^{-\frac{-4 \cdot 20 \log k}{3}} && \text{(Lemma 3.14)} \\
&\leq k^{-26}.
\end{aligned}$$

We can then apply a simple union bound to see that $\Pr\left(\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right) \leq k \cdot k^{-26} = k^{-25}$.

For $r \in [k]$, let $\mathbb{1}\left[\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right]$ be an indicator random variable for $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}$. Note that $\mathcal{E}_{\mathsf{sh\text{-}Rpt}}$ is then exactly the event that $\sum_{r=1}^{k} \mathbb{1}\left[\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right] \geq 0.05k$. Thus, we can analyse the probability of this event, by considering all possible sets $S \subseteq [r]$ of size $0.05k$. We can thus bound the

probability of $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}$ by

$$\begin{aligned}
\Pr(\mathcal{E}_{\mathsf{sh\text{-}Rpt}}) &= \Pr\left(\sum_{r=1}^{k} \mathbb{1}\left[\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right] \geq 0.05k\right) \\
&= \Pr\left(\bigcup_{\substack{S\subseteq[k]\\|S|=0.05k}} \bigcap_{r\in S} \mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right) \\
&\leq \sum_{\substack{S\subseteq[k]\\|S|=0.05k}} \Pr\left(\bigcap_{r\in S} \mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right) && \text{(union bound)} \\
&= \sum_{\substack{S\subseteq[k]\\|S|=0.05k}} \prod_{r\in S} \Pr\left(\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}\right) \\
&\leq \sum_{\substack{S\subseteq[k]\\|S|=0.05k}} \prod_{r\in S} k^{-25} \\
&= \sum_{\substack{S\subseteq[k]\\|S|=0.05\cdot k}} k^{-25\cdot 0.05k} \\
&= \sum_{\substack{S\subseteq[k]\\|S|=0.05\cdot k}} k^{-1.25k} \\
&\leq \sum_{S\subseteq[k]} k^{-1.25k} \\
&= 2^k k^{-1.25\cdot k} \\
&\leq 2^{-6k}.
\end{aligned}$$

Now, let us analyse $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$. For $i \in [10^4 k]$, let N'_i be the an indicator random variable for the event that the ith broadcast at Line 15 is affected by noise for the leader. Note that $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$ is exactly the event that $\sum_{i=1}^{10^4 k} N'_i \geq 0.15 \cdot 10^4 k$. Furthermore, note that the distributions of N'_i are independent, and that $\mathbb{E}[N'_i] = 1/10$. We can thus analyse the probability of $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$: This thus bounds the probability of $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$ by

$$\begin{aligned}
\Pr(\mathcal{E}_{\mathsf{sh\text{-}Ld}}) &= \Pr\left(\sum_{i=1}^{10^4 k} N'_i \geq 0.15 \cdot 10^4 k\right) \\
&= \Pr\left(\sum_{i=1}^{10^4 k} N'_i \geq (1+0.5) \cdot 0.1 \cdot 10^4 k\right) \\
&\leq \mathrm{e}^{-\frac{\frac{1}{2}\cdot\frac{1}{10}10^4 k}{3}} && \text{(Lemma 3.14)}
\end{aligned}$$

$$= \mathrm{e}^{-\frac{10^3}{6}k}$$
$$\leq 2^{-6k}.$$

Finally, we can use a simple union bound to show that $\Pr(\mathcal{E}_{\mathsf{sh}}) \leq 2^{-6k} + 2^{-6k} \leq 2^{-5k}$. □

Similarly, note that for all $t \in [10^5 k]$, $\mathcal{E}_{\mathsf{ecc}}(t)$ depends entirely on the noise in the channel during iteration t. In particular, this means that $\mathcal{E}_{\mathsf{sh}}(t)$ and $\mathcal{E}_{\mathsf{sh}}(t')$ are independent for all $t \neq t'$. Furthermore:

Lemma 4.4. *For all $t \in [10^5 k]$, we have:*

$$\Pr(\mathcal{E}_{\mathsf{ecc}}(t)) \leq 2^{-5k}.$$

Proof. Fix $t \in [10^5 k]$. For the rest of the proof, all variables are taken to have their values during the execution in round t.

For $j \in [n]$, let $\mathcal{E}^j_{\mathsf{ecc}}$ be the event that at least $0.2 \cdot 10^4 k$ of the broadcasts at Line 3 are affected by noise for player $\mathtt{p}(j)$. Note that $\mathcal{E}_{\mathsf{ecc}} = \mathcal{E}^1_{\mathsf{ecc}} \cup \cdots \cup \mathcal{E}^k_{\mathsf{ecc}}$.

For all $j \in [n]$ and $i \in [10^4 k]$, let N^j_i be an indicator random variable for if player $\mathtt{p}(j)$ is affected by noise during the ith broadcast at Line 3. Now that $\mathcal{E}^j_{\mathsf{ecc}}$ is then exactly the event that $\sum_{i=1}^{10^4 k} N^j_i \geq 0.2 \cdot 10^4 k$. Furthermore, note that the distributions of N^j_i are independent, and that $\mathbb{E}\left[N^j_i\right] = 1/10$. This thus bounds the probability of $\mathcal{E}^j_{\mathsf{ecc}}$ by

$$\begin{aligned}
\Pr\left(\mathcal{E}^j_{\mathsf{ecc}}\right) &= \Pr\left(\sum_{i=1}^{10^4 k} N^j_i \geq 0.2 \cdot 10^4 k\right) \\
&= \Pr\left(\sum_{i=1}^{10^4 k} N^j_i \geq (1+1) \cdot 0.1 \cdot 10^4 k\right) \\
&\leq \mathrm{e}^{-\frac{0.1 \cdot 10^4 k}{3}} \\
&\leq 2^{-6k}.
\end{aligned}$$

We can then apply a simple union bound to see that $\Pr(\mathcal{E}_{\mathsf{ecc}}) \leq k \cdot 2^{-6k} \leq 2^{-5k}$. □

Combining the results of Lemma 4.3 and Lemma 4.4 using a simple union bound, we thus see that for each $t \in [10^5 k]$, $\Pr(\mathcal{E}(t)) \leq 2^{-5k} + 2^{-5k} \leq 2^{-4k}$. Furthermore, as $\mathcal{E}(t)$ only depends on the noise in the channel during iteration t, $\mathcal{E}(t)$ and $\mathcal{E}(t')$ are independent for $t \neq t'$. We can then apply this to determine that:

Lemma 4.5. *It holds that:*

$$\Pr\left(\left|\left\{t \in \left[10^5 k\right] : \mathcal{E}(t)\right\}\right| \geq k\right) \leq 2^{-k^2}.$$

Proof. Consider all possible sets of size $S \subseteq [10^5 k]$ of size k. Then $|\{t \in [10^5 k] : \mathcal{E}(t)\}| \geq k$ if and only if there is some set S such that $\mathcal{E}(t)$ holds for all $t \in S$. Thus, we can use a union

bound to analyse the probability of this event:

$$\begin{aligned}
\Pr\left(\left|\left\{t \in \left[10^5 k\right] : \mathcal{E}(t)\right\}\right| \geq k\right) &= \Pr\left(\bigcup_{\substack{S \subseteq \left[10^5 k\right] \\ |S|=k}} \bigcap_{t \in S} \mathcal{E}(t)\right) \\
&\leq \sum_{\substack{S \subseteq \left[10^5 k\right] \\ |S|=k}} \Pr\left(\bigcap_{t \in S} \mathcal{E}(t)\right) && \text{(union bound)} \\
&= \sum_{\substack{S \subseteq \left[10^5 k\right] \\ |S|=k}} \prod_{t \in S} \Pr(\mathcal{E}(t)) \\
&\leq \sum_{\substack{S \subseteq \left[10^5 k\right] \\ |S|=k}} \prod_{t \in S} 2^{-4k} \\
&= \sum_{\substack{S \subseteq \left[10^5 k\right] \\ |S|=k}} \left(2^{-4k}\right)^k \\
&\leq \sum_{S \subseteq [10^5 k]} 2^{-4k^2} \\
&= 2^{10^5 k} 2^{-4k^2} \\
&\leq 2^{-k^2}.
\end{aligned}$$

□

Now, let us show that $\mathcal{E}(t)$ represents an iteration failing, such that if $\mathcal{E}(t)$ does not occur, then our algorithms transmit information successfully.

Lemma 4.6. *For $t \in [10^5 k]$, if $\mathcal{E}(t)$ does not occur, then:*

1. *For all $j \in [k]$, $\tilde{\pi}^{\mathsf{p}(j)}(t) = \pi^{\mathsf{Ld}}(t-1)$.*
2. *For all $j \in [k]$, $\tilde{z}_j^{\mathsf{Ld}}(t) = z^j(t)$.*

Proof. Fix $t \in [10^5 k]$. For the rest of the proof, all variables are taken to have their values during the execution in round t, with the exception of π^{Ld}, which is used to denote $\pi^{\mathsf{Ld}}(t-1)$.

Suppose that $\mathcal{E} = \mathcal{E}_{\mathsf{sh}} \cup \mathcal{E}_{\mathsf{ecc}}$ does not occur. We then know that $\mathcal{E}_{\mathsf{sh}}$ and $\mathcal{E}_{\mathsf{ecc}}$ do not occur. We then know that the following things happen:

1. As $\mathcal{E}_{\mathsf{ecc}}$ does not occur, for each $j \in [k]$, less than $0.2 \cdot 10^4 k$ of the broadcasts by the leader made at Line 3 are corrupted by noise for player $\mathsf{p}(j)$. As such,

$$\Delta\left(\widetilde{\mathsf{ECC}}^i, \mathsf{ECC}\left(\pi^{\mathsf{Ld}}\right)\right) < 0.2 \cdot 10^4 k.$$

Thus, by Lemma 3.18 and because ECC is a $(0.4, \Gamma, k, 10^4)$-error correcting code,

$$\tilde{\pi}^i = \mathsf{D\text{-}ECC}\Big(\widetilde{\mathsf{ECC}}^i\Big) = \pi^{\mathsf{Ld}}.$$

2. As $\mathcal{E}_{\mathsf{sh}}$ does not occur, we know that $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$ and $\mathcal{E}_{\mathsf{sh\text{-}Rpt}}$ do not occur. The latter then implies that $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}$ occurs for less than $0.05k$ of the $r \in [k]$.

   Let $z = z^1 \| \cdots \| z^k$. Note that for each repeater $r \in [k]$ such that $\mathcal{E}^r_{\mathsf{sh\text{-}Rpt}}$ does not occur, for every $j \in [k]$, we get that $\tilde{z}^r_j = z^j$, as the majority of the broadcasts from player $\mathsf{p}(j)$ to repeater r are not affected by noise. Thus, for all such r, $\tilde{z}^r = z$. And as a result, $\mathsf{ECC}_{(10^4(r-1),10^4 r]}(\tilde{z}^r) = \mathsf{ECC}_{(10^4(r-1),10^4 r]}(z)$.

   Let $\mathsf{ECC}' = \mathsf{ECC}_{(0,10^4]}(\tilde{z}^1) \| \mathsf{ECC}_{(10^4,10^4\cdot 2}(\tilde{z}^2) \| \cdots \| \mathsf{ECC}_{(10^4(k-1),10^4 k]}\big(\tilde{z}^k\big)$.

   Thus, as $\mathcal{E}^k_{\mathsf{sh\text{-}Rpt}}$ occurs for less than $0.05k$ of the $r \in [k]$, we thus get that

$$\Delta(\mathsf{ECC}', \mathsf{ECC}(z)) < 0.05 \cdot 10^4 k.$$

   Now, also note that $\mathcal{E}_{\mathsf{sh\text{-}Ld}}$ does not occur. Thus, we know that

$$\Delta\Big(\mathsf{ECC}', \widetilde{\mathsf{ECC}}\Big) < 0.15 \cdot 10^4 k.$$

   Then, by applying a triangle inequality, we see that

$$\begin{aligned}\Delta\Big(\mathsf{ECC}(z), \widetilde{\mathsf{ECC}}\Big) &\leq \Delta(\mathsf{ECC}(z), \mathsf{ECC}') + \Delta\Big(\mathsf{ECC}', \widetilde{\mathsf{ECC}}\Big) \\ &< 0.05 \cdot 10^4 k + 0.15 \cdot 10^4 k \\ &= 0.2 \cdot 10^4 k.\end{aligned}$$

   Then, by applying Lemma 3.18, we get that $\tilde{z}^{\mathsf{Ld}} = \mathsf{D\text{-}ECC}\Big(\widetilde{\mathsf{ECC}}\Big) = z$.

□

Now, we define a potential function $\Phi(\cdot)$ by defining $\Phi(0) = 0$ and, for $t \in [10^5 k]$,

$$\Phi(t) = \begin{cases} \big\|\pi^{\mathsf{Ld}}(t)\big\|, & \text{if } \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\| = k \\ 2 \cdot \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\| - \big\|\pi^{\mathsf{Ld}}(t)\big\|, & \text{if } \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\| < k \end{cases}. \tag{4}$$

To get the intuition behind the definition of Φ, imagine that the correct transcript $\Pi(X)$ and the simulated transcript $\pi^{\mathsf{Ld}}(t)$ match after the first k symbols. Then, one can observe that the potential Φ corresponds to the length of the correct prefix of the simulated transcript minus the length of the incorrect part after the prefix, since

$$2 \cdot \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\| - \big\|\pi^{\mathsf{Ld}}(t)\big\| = \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\| - \big(\big\|\pi^{\mathsf{Ld}}(t)\big\| - \big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(t), \Pi(X)\big)\big\|\big).$$

Moreover, as the leader only appends or removes one symbol from the end of π^{Ld} in each iteration, we get that the value of Φ can change by at most one in each iteration. We shall use this later to get Corollary 4.8 from Lemma 4.7.

Lemma 4.7. *For all $t \in [10^5 k]$ where $\mathcal{E}(t)$ does not occur, we have:*

$$\Phi(t) \geq \Phi(t-1) + 1.$$

Proof. Fix some $t \in [10^5 k]$, and suppose that $\mathcal{E}(t)$ does not occur. Thus, by Lemma 4.6. $\tilde{\pi}^{\mathsf{p}(j)}(t) = \pi^{\mathsf{Ld}}(t-1)$ for all $j \in [k]$, and $\tilde{z}_j^{\mathsf{Ld}}(t) = z^j(t)$ for all $j \in [k]$.

We will consider two cases, depending which of the two cases of Eq. (4) we are in.

Case 1: $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right)\right\| = k$**:** In this case, $\Phi(t-1) = \left\|\pi^{\mathsf{Ld}}(t-1)\right\|$. Furthermore, $\pi^{\mathsf{Ld}}_{\leq k}(t-1) = \Pi(X)$, as $\|\Pi(X)\| = k$.

Thus, at Line 5, for all $j \in [k]$, every party $\mathsf{p}(j)$ sets $z^j(t)$ as

$$\begin{aligned} z^j(t) &= M_j\left(x^{\mathsf{p}(j)}, \tilde{\pi}^{\mathsf{p}(j)}_{<j}(t)\right) \\ &= M_j\left(x^{\mathsf{p}(j)}, \pi^{\mathsf{Ld}}_{<j}(t-1)\right) \\ &= M_j\left(x^{\mathsf{p}(j)}, \Pi_{<j}(X)\right) \\ &= \Pi_j(X) \\ &= \pi^{\mathsf{Ld}}_j(t-1). \end{aligned}$$

Then, the leader will receive $\tilde{z}_j^{\mathsf{Ld}}(t) = z^j(t) = \pi_j^{\mathsf{Ld}}(t-1)$. Therefore, we get that $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \tilde{z}^{\mathsf{Ld}}(t)\right)\right\| = k$, so the leader will extends π^{Ld} in Line 8 by some arbitrary symbol. Therefore, $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right)\right\| = k$ as well, but $\left\|\pi^{\mathsf{Ld}}(t)\right\| = \|\pi^{\mathsf{Ld}}(t-1)\| + 1$. Thus, $\Phi(t) = \Phi(t-1) + 1$.

Case 2: $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right)\right\| < k$**:** Then, $\Phi(t-1) = 2 \cdot \left\|\mathsf{LCP}(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\|$. We will split this case into two further sub-cases, based on if $\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right) = \pi^{\mathsf{Ld}}(t-1)$ or not.

Case 2a: $\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right) = \pi^{\mathsf{Ld}}(t-1)$**:** In this case, $\pi^{\mathsf{Ld}}(t-1) = \Pi(X)_{\leq \left\|\pi^{\mathsf{Ld}}(t-1)\right\|}$.

Thus,

$$\begin{aligned} \Phi(t-1) &= 2 \cdot \left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\| \\ &= 2 \cdot \left\|\pi^{\mathsf{Ld}}(t-1)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\| \\ &= \left\|\pi^{\mathsf{Ld}}(t-1)\right\|. \end{aligned}$$

We can then repeat the analysis from Case 1 to see that for $j \in \left[\left\|\pi^{\mathsf{Ld}}(t-1)\right\| + 1\right]$, $\tilde{z}_j^{\mathsf{Ld}}(t) = z^j(t) = \Pi_j(X)$. Furthermore, for $j \leq \left\|\pi^{\mathsf{Ld}}(t-1)\right\|$, we also have that $\Pi_j(X) = \pi_j^{\mathsf{Ld}}(t-1)$, so $\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \tilde{z}^{\mathsf{Ld}}(t)\right) = \pi^{\mathsf{Ld}}(t-1)$.

Thus, we end up extending π^{Ld} in Line 8 by $\tilde{z}^{\mathsf{Ld}}_{\left\|\pi^{\mathsf{Ld}}(t-1)\right\|+1}(t) = \Pi_{\left\|\pi^{\mathsf{Ld}}(t-1)\right\|+1}(X)$. As such, $\pi^{\mathsf{Ld}}(t)$ becomes equal to $\Pi_{\leq\left\|\pi^{\mathsf{Ld}}(t-1)\right\|+1}(X) = \Pi_{\leq\left\|\pi^{\mathsf{Ld}}(t)\right\|}(X)$. Thus,

$$\Phi(t) = \left\|\pi^{\mathsf{Ld}}(t)\right\| = \left\|\pi^{\mathsf{Ld}}(t-1)\right\| + 1 = \Phi(t-1) + 1.$$

Case 2b: $\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right) \neq \pi^{\mathsf{Ld}}(t-1)$**:** In this case, $\Phi(t-1) = 2\cdot\left\|\mathsf{LCP}(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\|$ and there exists some $j \in \left[\min\left(\left\|\pi^{\mathsf{Ld}}(t-1)\right\|, k\right)\right]$ such that $\pi^{\mathsf{Ld}}_j(t-1) \neq \Pi_j(X)$. Let j' be the smallest such j. We can now analyse the behaviour of party $\mathtt{p}(j')$ at Line 5 to see that

$$\begin{aligned} z^{j'}(t) &= M_{j'}\left(x^{\mathtt{p}(j')}, \tilde{\pi}^{\mathtt{p}(j')}_{<j'}(t)\right) \\ &= M_{j'}\left(x^{\mathtt{p}(j')}, \pi^{\mathsf{Ld}}_{<j'}(t-1)\right) \\ &= M_{j'}\left(x^{\mathtt{p}(j')}, \Pi_{<j'}(X)\right) \\ &= \Pi_{j'}(X). \end{aligned}$$

Furthemore, as established before, $\tilde{z}^{\mathsf{Ld}}_{j'}(t) = z^{j'}(t) = \Pi_{j'}(X)$. Thus, $\tilde{z}^{\mathsf{Ld}}_{j'}(t) \neq \pi^{\mathsf{Ld}}_{j'}(t-1)$. As such, $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \tilde{z}^{\mathsf{Ld}}(t)\right)\right\| < j' \leq \min\left(\left\|\pi^{\mathsf{Ld}}(t-1)\right\|, k\right)$, so the leader will shrink π^{Ld} in Line 10.

Thus, $\pi^{\mathsf{Ld}}(t)$ becomes $\pi^{\mathsf{Ld}}(t-1)$ with the last symbol removed. Thus, $\left\|\pi^{\mathsf{Ld}}(t)\right\| = \left\|\pi^{\mathsf{Ld}}(t-1)\right\| - 1$. As a result, note that $\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right) = \mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right)$, as the last symbol in $\pi^{\mathsf{Ld}}(t-1)$ is not in the prefix. Thus, $\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right)\right\| < k$, and we can compute $\Phi(t)$ as

$$\begin{aligned} \Phi(t) &= 2\cdot\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right)\right\| - \left\|\pi^{\mathsf{Ld}}(t)\right\| \\ &= 2\cdot\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right)\right\| - \left(\left\|\pi^{\mathsf{Ld}}(t-1)\right\| - 1\right) \\ &= 2\cdot\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t), \Pi(X)\right)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\| + 1 \\ &= 2\cdot\left\|\mathsf{LCP}\left(\pi^{\mathsf{Ld}}(t-1), \Pi(X)\right)\right\| - \left\|\pi^{\mathsf{Ld}}(t-1)\right\| + 1 \\ &= \Phi(t-1) + 1. \end{aligned}$$

Thus, in all possible cases, $\Phi(t) = \Phi(t-1) + 1$. □

Corollary 4.8. *Whenever* $|\{t \in [10^5 k] : \mathcal{E}(t)\}| < k$*, we have* $\Phi(10^5 k) > 10^4 k$.

Proof. Whenever $|\{t \in [10^5 k] : \mathcal{E}(t)\}| < k$, there are at least $(10^5 - 1)k$ rounds such that $\mathcal{E}(t)$ does not occur. By Lemma 4.7, we thus get that $\Phi(t)$ increases by at least one in each of those rounds. Furthermore, recall that in the remaining less than k rounds, $\Phi(t)$ decreases by at most one.

Thus, over the span of the algorithm, we get that

$$\Phi(10^5 k) \geq \big(10^5 - 1\big)k - k \geq 10^4 k.$$

□

Thus, conditioned on $|\{t \in [10^5 k] : \mathcal{E}(t)\}| < k$, using Corollary 4.8 and Eq. (4), we get that $\big\|\pi^{\mathsf{Ld}}(10^5 k)\big\| \geq 10^4 k$ and $\big\|\mathsf{LCP}\big(\pi^{\mathsf{Ld}}(10^5 k), \Pi(X)\big)\big\| = k$, so $\pi^{\mathsf{Ld}}_{\leq k}(10^5 k) = \Pi(X)$. Therefore, as the leader finishes by outputting $\pi^{\mathsf{Ld}}_{\leq k}$ in Line 13, we get that $\Pi'(X) = \Pi(X)$.

Finally, Lemma 4.5 gives us that

$$\begin{aligned}\Pr(\Pi'(X) \neq \Pi(X)) &\leq \Pr\big(\big|\big\{t \in \big[10^5 k\big] : \mathcal{E}(t)\big\}\big| \geq k\big) \\ &\leq 2^{-k^2},\end{aligned}$$

which concludes the proof of Theorem 4.2. □

4.1 Proof of Theorem 4.1

We now proceed to prove Theorem 4.1.

Fix some T. Recall that we want our simulation scheme to use $T \cdot \tilde{O}\big(\min\big(\sqrt{\log T}, n\big)\big)$ rounds. When $\sqrt{\log T} \leq n$, this reduces to $T \cdot \tilde{O}\big(\sqrt{\log T}\big)$. We show how to achieve this result in Section 4.1.1. If $\sqrt{\log T} \geq n$, however, we wish to find a simulation scheme that takes $T \cdot \tilde{O}(n)$ communication. We show such a simulation scheme that works for all T (including $\sqrt{\log T} \leq n$) in Section 4.1.2. Combining these two schemes finishes the proof.

4.1.1 "Short" Protocols ($2\sqrt{\log T} \leq n/3$)

Consider the case where $2\sqrt{\log T} \leq n/3$. Then, setting $k = 2\sqrt{\log T}$ means that k satisfies the conditions of Theorem 4.2.[18] Without loss of generality, let k divide T.

We can then split our T rounds into chunks of size k. From there, we will simply run Algorithm 1 on k steps of Π at a time. The leader will then pad the resulting π^{Ld} to have length k^2, and encode it using a $(0.4, \Gamma, k^2, 10^4)$-error correcting code. All players then decode this code, and use that to arrive at a conclusion for what the first k messages in $\Pi(X)$ are. We can then simulate the next chunk using Algorithm 1. We can repeat this for all $\frac{T}{k}$ such chunks, and then concatenate all the resulting transcripts so that every player has a guess for $\Pi(X)$.

Note that this algorithm requires $\frac{T}{k}$ rounds of communication with $10^{25}k^2 \log k + 10^4 k^2$ communication per round. This gives a total communication of $Tk \cdot \mathcal{O}(\log k) = T\tilde{\mathcal{O}}(\sqrt{\log T})$, as desired.

[18] We require $k = 2\sqrt{\log T} \leq n/3$ to hold in Theorem 4.2 because Algorithm 2 requires us to have k different repeaters.

Furthermore, note that the leader for each chunk correctly computes the transcript for $\Pi(X)$ in that chunk except with probability $2^{-k^2} = 2^{-4\log T} = T^{-4}$. Furthermore, the probability of a given player incorrectly decoding π^{Ld} from the error correcting code in a given chunk is at most $\mathrm{e}^{-10^3 k^2/3} = T^{-4}$.

Thus, by taking a union bound over the at most T players who speak and over our T/k chunks, we get that all players know the correct transcript for each chunk except with probability at most T^{-2}.

Thus, our players can accurately compute $\Pi(X)$ except with probability at most T^{-2}, and the desired result is shown.

4.1.2 "Long" Protocols

Now, we show a way of simulating any protocol of a length T using $T \cdot \tilde{\mathcal{O}}(n)$ rounds. At a high level, we will use the simulation scheme in [EKS18] to first simulate the protocol using an *adaptive* protocol of length $\mathcal{O}(T)$, and then simulate that adaptive protocol by splitting up each adaptive round into $\mathcal{O}(n \log n)$ non-adaptive rounds.

Adaptive simulation. We begin by refering to the main result of [EKS18], which says that any n-party protocol Π taking T rounds can be simulated in a noise-resilient way in $\mathcal{O}(T)$ rounds by an adaptive protocol: One where multiple parties may speak in a single round, and where a party can decide whether to speak in a given round depending on the transcript it heard so far. Our strategy will then be to simulate this adaptive protocol in a non-adaptive setting.

Simulating adaptive protocols. Consider a single round of the adaptive protocol, which might have any number of players speaking. We can replace this by $n \cdot \mathcal{O}(\log n)$ non-adaptive rounds, with each player speaking for $\mathcal{O}(\log n)$ rounds. When a player wishes to send a symbol $z \in \Gamma$ in the adaptive round, that player will send z $\mathcal{O}(\log n)$ times in the non-adaptive rounds. This allows the other players to accurately decode each player's message except with polynomially small probability.

Simulating silences. However, players might also elect to stay silent in an adaptive round, which they communicate by sending a message of length $\mathcal{O}(\log n)$ which is very different from the encoding of any symbol in Γ. As each symbol in Γ is encoded via repeating that symbol $\mathcal{O}(\log n)$ times, when a player wishes to communicate a silence, that player needs to send a string with many of its coordinates being different (*e.g.* $123\ldots123\ldots$).

Note that any two encoded messages a player might choose to send will differ in at least half of the $\mathcal{O}(\log n)$ positions. Thus, decoding each message via least Hamming distance results in a given player decoding another player's intent (whether that is a symbol or a silence) correctly, except with inverse polynomial probability.

Thus, as each player can correctly decode every other player's message except with inverse polynomial probability, the players can simulate whether any collisions occurred, thus allowing the players to simulate any adaptive round with the desired small error probability.

5 Lower Bound

Given integers $n, m > 0$, the n-depth, m-width pointer chasing function $\mathtt{PC}_{n,m}$ takes as input an integer[19] $f_1 \in [m]$ and $n-1$ functions $f_2, f_3, \cdots, f_n : [m] \to [m]$, and outputs the value:

$$\mathtt{PC}_{n,m}(f_1, \cdots, f_n) = f_n(f_{n-1}(\cdots(f_2(f_1))\cdots)). \tag{5}$$

We shall use F to denote the tuple $(f_1, \cdots, f_n)$, $\mathcal{F}$ to denote the uniform distribution over all F, and $\mathsf{F} = (\mathsf{f}_1, \cdots, \mathsf{f}_n)$ to denote a random variable sampled from $\mathcal{F}$. We drop n, m from the subscript when they are clear from context. Let $\mathtt{LSB}(\cdot)$ be the function that takes an integer and outputs the least significant bit in the binary representation of that integer, *i.e.*, the protocol outputs 1 if the integer is odd and 0 if it is even[20]. Observe that, for any n, m, the function $\mathtt{LSB}(\mathtt{PC}_{n,m})$ can be computed by an n-round protocol over the noiseless broadcast channel with n parties and alphabet $[m]$, when the input to player i, for all $i \in [n]$, is the function f_i. For our lower bound, we show that such a protocol requires $\Omega\big(n \cdot \sqrt{\log n}\big)$ rounds in the noisy case, even if it is only required to estimate $\mathtt{LSB}(\mathtt{PC}_{n,m})$ with a small advantage over random guessing.

In more details, we show that, when $m = \mathsf{poly}(n)$, any protocol over the n-party noisy broadcast channel with alphabet $\Gamma = [m]$ and noise parameter[21] $\epsilon = \frac{1}{2}$ requires at least $\Omega\big(n \cdot \sqrt{\log n}\big)$ rounds to estimate $\mathtt{LSB}(\mathtt{PC}_{n,m}(\cdot))$ with advantage $\frac{1}{n}$ when the input to player i, for all $i \in [n]$, is the function f_i. Formally, we show that:

Theorem 5.1. *Let $n > 0$ be large enough and set $m = 2 \cdot n^{200}$ and $\epsilon = \frac{1}{2}$. For any randomized protocol Π over the $(n, \epsilon, [m])$-noisy broadcast model satisfying $\|\Pi\| < \frac{n \cdot \sqrt{\log n}}{10^5}$, we have:*

$$\Pr_{\mathsf{F} \sim \mathcal{F}, \mathsf{N}}(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F}))) \leq \frac{1}{2} + \frac{1}{n},$$

where N is the random variable corresponding to the noise in the channel[22].

We prove Theorem 5.1 in the rest of this section. Let n, m, ϵ, Π be fixed as in the theorem statement. As a randomized protocol is simply a distribution over deterministic protocols, we can assume Π to be deterministic without loss of generality. By adding $n+1$ extra rounds, we can also assume without loss of generality that $\mathtt{p}(1) = 1$ and $\mathtt{p}(T+1-i) = n+1-i$ for

[19] We sometimes treat this integer as a function from the singleton set $\{0\}$ to the set $[m]$.

[20] The choice of the function $\mathtt{LSB}(\cdot)$ is made for concreteness. Any function that is "balanced" would be enough for our proof.

[21] Our proof works for any constant ϵ but we fix it to be $\frac{1}{2}$ for simplicity.

[22] To simplify notation, we henceforth use $\Pr(\cdot)$ instead of $\Pr_{\mathsf{F} \sim \mathcal{F}, \mathsf{N}}(\cdot)$.

all $i \in [n]$. We also define the function $\mathtt{p}^{-1}(\cdot)$ to be the "inverse" of $\mathtt{p}$, *i.e.*, for $i \in [n]$, we have:

$$\mathtt{p}^{-1}(i) = \{j \in [T] \mid \mathtt{p}(j) = i\}.$$

5.1 Basic Definitions

Throughout, we use $\mathcal{P} \subset [n]$ to denote a subset of players and $\mathcal{R} \subseteq [T]$ to denote a subset of rounds. We reserve $P = |\mathcal{P}|$ to denote the size of $\mathcal{P}$ and let $\mathcal{P}(1) < \mathcal{P}(2) < \cdots < \mathcal{P}(P)$ be the elements of $\mathcal{P}$.

Definition 5.2. *We say a pair $(\mathcal{P}, \mathcal{R})$ is nice if $\mathtt{p}(r) \in \mathcal{P}$ for all $r \in \mathcal{R}$.*

Consider a nice pair $(\mathcal{P}, \mathcal{R})$ and for $r \in \mathcal{R}$ and $p \in [P]$, define the set:

$$\mathcal{B}_{\mathcal{P},\mathcal{R},p,r} = \{0\} \cup \{r' \in \mathcal{R} \mid r' < r \wedge \mathtt{p}(r') = \mathcal{P}(p)\}. \tag{6}$$

We reserve $B_{\mathcal{P},\mathcal{R},p,r} = |\mathcal{B}_{\mathcal{P},\mathcal{R},p,r}|$ to denote the size of $\mathcal{B}_{\mathcal{P},\mathcal{R},p,r}$ and let $0 = \mathcal{B}_{\mathcal{P},\mathcal{R},p,r}(1) < \mathcal{B}_{\mathcal{P},\mathcal{R},p,r}(2) < \cdots < \mathcal{B}_{\mathcal{P},\mathcal{R},p,r}(B_{\mathcal{P},\mathcal{R},p,r})$ be the elements of $\mathcal{B}_{\mathcal{P},\mathcal{R},p,r}$. Observe that all these elements are strictly less than r. We adopt the convention $\mathcal{B}_{\mathcal{P},\mathcal{R},p,r}(B_{\mathcal{P},\mathcal{R},p,r} + 1) = r$ and shall drop some of the subscripts $\mathcal{P}, \mathcal{R}, p, r$ when they are clear from context. Next, we define:

Definition 5.3 (Crashing Sets)**.** *Consider a nice pair $(\mathcal{P}, \mathcal{R})$ and inductively define, for $r \in \{0\} \cup \mathcal{R}$, a family of crashing sets $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)$ as follows: Define $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(0)$ to be the family of all subsets of $\mathcal{P} \times [T]$. For $r > 0$ such that $\mathtt{p}(r) = \mathcal{P}(1)$, define $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r) = \emptyset$ to be the empty family. Finally, for $r > 0$ such that $\mathtt{p}(r) = \mathcal{P}(p)$ for some $p > 1$, define $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)$ to be the family containing all sets $C \in \mathcal{P} \times [T]$ for which there exists a value $b \in [B_{p-1,r}]$ and a set $C' \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{p-1,r}(b))$ satisfying*

$$C' \cup \{(\mathcal{P}(p), r') \mid \mathcal{B}_{p-1,r}(b+1) \leq r' < r\} \subseteq C.$$

We shall use $\mathtt{t}_{\mathcal{P},\mathcal{R}}(r) = \min_{C \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)} |C|$ to denote the size of the smallest set in $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)$ and adopt the convention that $\mathtt{t}_{\mathcal{P},\mathcal{R}}(r) = \infty$ if $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r) = \emptyset$. As before, we drop the subscripts $\mathcal{P}, \mathcal{R}$ when they are clear from context. Observe that the sets $\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)$ are all upwards closed, *i.e.*,

Observation 5.4. *Consider a nice pair $(\mathcal{P}, \mathcal{R})$ and $r \in \{0\} \cup \mathcal{R}$. For all $C \subseteq C' \subseteq \mathcal{P} \times [T]$, we have $C \in \mathtt{Crash}(r) \implies C' \in \mathtt{Crash}(r)$.*

Observation 5.5. *Consider a nice pair $(\mathcal{P}, \mathcal{R})$ and $r \in \{0\} \cup \mathcal{R}$. For all sets $C \subseteq \mathcal{P} \times [T]$, we have $C \in \mathtt{Crash}(r) \implies C \cap (\mathcal{P} \times [r]) \in \mathtt{Crash}(r)$.*

5.2 A Reduction to the Adversarial Model

We start by stating the notation used in this subsection. The notations $\mathsf{\Pi}^i_{\leq j}(\cdot)$, $\mathsf{\Pi}_{\leq j}(\cdot)$ are as defined in Section 3.5. We extend these to sets $S \subseteq [T]$ by defining and $\mathsf{\Pi}^i_S(\cdot)$ and $\mathsf{\Pi}_S(\cdot)$ to be $\left\{\mathsf{\Pi}^i_j(\cdot)\right\}_{j\in S}$ and $\{\mathsf{\Pi}_j(\cdot)\}_{j\in S}$ respectively. Recall that $\mathsf{F} = (\mathsf{f}_1, \cdots, \mathsf{f}_n)$ and define, for $i \in [n]$, the notations $\mathsf{F}_{-i} = (\mathsf{f}_1, \cdots, \mathsf{f}_{i-1}, \mathsf{f}_{i+1}, \cdots \mathsf{f}_n)$, $\mathsf{F}_{\leq i} = (\mathsf{f}_1, \cdots, \mathsf{f}_i)$, and $\mathsf{F}_{<i} = (\mathsf{f}_1, \cdots, \mathsf{f}_{i-1})$. Also define, for a set $S \subseteq [m]$ and a function $f : [m] \to [m]$ the set $f^{-1}(S) = \{z \in [m] \mid f(z) \in S\}$. If X is a random variable and x is a value that X can take, we sometimes abbreviate the event $\mathsf{X} = x$ as simply x when it is clear from context, *e.g.*, we may write $\Pi_{\leq j}$ instead of $\mathsf{\Pi}_{\leq j}(\mathsf{F}) = \Pi_{\leq j}$, *etc.* We also sometimes omit the argument F in notations such as $\mathsf{\Pi}_{\leq j}(\mathsf{F})$ when it is clear from context.

5.2.1 Properties of Protocols

In this section, we collect some results about broadcast protocols.

Definition 5.6. *Let $0 \leq j \leq T$ be given. For all $(\Pi_{\leq j}, N)$ and $i \in [n]$, define the set:*

$$\mathtt{Rec}_i(\Pi_{\leq j}, N) = \left\{f_i \in \mathsf{supp}(\mathsf{f}_i) \mid \forall j' \in \mathsf{p}^{-1}(i) \cap [j] : \Pi_{j'} = M_{j'}\left(f_i, \Pi^i_{<j'}\right)\right\},$$

where $\Pi^i_{\leq j}$ is the sequence of symbols received by party i when $\Pi_{\leq j}$ is the sequence of sent symbols and N is the channel noise (see Section 3.5).

Lemma 5.7. *Let $0 \leq j \leq T$ and $(\Pi_{\leq j}, N)$ be given. The following events are equivalent:*

$$(N, \Pi_{\leq j}) \equiv (N, \forall i \in [n] : \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\leq j}, N)).$$

Proof. Proof by induction on j. The base case $j = 0$ is trivial. We show the lemma for $j > 0$ by assuming it holds for $j - 1$. Fix $(\Pi_{\leq j}, N)$ and for all $i \in [n]$, let $\Pi^i_{\leq j}$ be the sequence of symbols received by party i when $\Pi_{\leq j}$ is the sequence of sent symbols and N is the channel noise. Observe that $\Pi^i_{<j}$ is determined by the pair $(\Pi_{<j}, N)$. We have:

$$\begin{aligned}
(N, \Pi_{\leq j}) &\equiv (N, \Pi_{<j}, \Pi_j) \\
&\equiv \left(N, \Pi_{<j}, M_j\left(\mathsf{f}_{\mathsf{p}(j)}, \Pi^{\mathsf{p}(j)}_{<j}\right) = \Pi_j\right) \\
&\equiv \left(N, \forall i \in [n] : \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{<j}, N), M_j\left(\mathsf{f}_{\mathsf{p}(j)}, \Pi^{\mathsf{p}(j)}_{<j}\right) = \Pi_j\right) && \text{(Induction hypothesis)} \\
&\equiv (N, \forall i \in [n] : \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\leq j}, N)). && \text{(Definition 5.6)}
\end{aligned}$$

□

Lemma 5.8. *Let $j \in [T]$ and $(\Pi_{\leq j}, N)$ be given. It holds that:*

$$\Pr(\Pi_j \mid \Pi_{<j}, N) = \frac{\left|\mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{\leq j}, N)\right|}{\left|\mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{<j}, N)\right|}.$$

Proof. We derive:

$$\begin{aligned}
\Pr(\Pi_j \mid \Pi_{<j}, N) &= \frac{\Pr(\Pi_{\le j}, N)}{\Pr(\Pi_{<j}, N)} \\
&= \frac{\Pr(N, \forall i \in [n] : \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\le j}, N))}{\Pr(N, \forall i \in [n] : \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{<j}, N))} && \text{(Lemma 5.7)} \\
&= \frac{\prod_{i\in[n]} \Pr(\mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\le j}, N))}{\prod_{i\in[n]} \Pr(\mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{<j}, N))} && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i\in[n]} \text{ and } \mathsf{N}) \\
&= \frac{\Pr\big(\mathsf{f}_{\mathsf{p}(j)} \in \mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{\le j}, N)\big)}{\Pr\big(\mathsf{f}_{\mathsf{p}(j)} \in \mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{<j}, N)\big)} && \text{(Definition 5.6)} \\
&= \frac{\big|\mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{\le j}, N)\big|}{\big|\mathtt{Rec}_{\mathsf{p}(j)}(\Pi_{<j}, N)\big|}. && \text{(As } \mathsf{f}_{\mathsf{p}(j)} \text{ is uniform)}
\end{aligned}$$

□

Lemma 5.9. *Let $i \in [n]$ and $0 \le j' \le j \le T$ be such that $\mathsf{p}^{-1}(i) \cap [j] = \mathsf{p}^{-1}(i) \cap [j']$. It holds for all $(\Pi_{\le j}, N)$ and $f_i \in \mathsf{supp}(\mathsf{f}_i)$ that:*

$$\Pr(f_i \mid \Pi_{\le j}, N) = \Pr(f_i \mid \Pi_{\le j'}, N).$$

Namely, the random variable f_i is independent of $\mathsf{\Pi}_{\le j}$ conditioned on $\mathsf{\Pi}_{\le j'}$ and N.

Proof. We have:

$$\begin{aligned}
\Pr(f_i \mid \Pi_{\le j}, N) &= \Pr(f_i \mid \forall i' \in [n] : \mathsf{f}_{i'} \in \mathtt{Rec}_{i'}(\Pi_{\le j}, N), N) && \text{(Lemma 5.7)} \\
&= \Pr(f_i \mid \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\le j}, N), N) && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i\in[n]} \text{ and } \mathsf{N}) \\
&= \Pr(f_i \mid \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\le j'}, N), N) && \text{(As } \mathsf{p}^{-1}(i) \cap [j] = \mathsf{p}^{-1}(i) \cap [j']) \\
&= \Pr(f_i \mid \forall i' \in [n] : \mathsf{f}_{i'} \in \mathtt{Rec}_{i'}(\Pi_{\le j'}, N), N) \\
& && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i\in[n]} \text{ and } \mathsf{N}) \\
&= \Pr(f_i \mid \Pi_{\le j'}, N). && \text{(Lemma 5.7)}
\end{aligned}$$

□

Definition 5.10. *For all $C \subseteq [n] \times [T]$, define the event $\mathcal{N}(C)$ over the randomness in N as*

$$\mathcal{N}(C) = \big\{N \mid \forall (i,j) \in C : N^i_j \neq \star\big\}.$$

Lemma 5.11. *Let $i \in [n]$ and $0 \le j' \le j \le T$ be given. Define $A = [j'] \cup ([j] \cap \mathsf{p}^{-1}(i))$ and let $N \in \mathcal{N}(\{i\} \times ([j] \setminus A))$.*

1. *The random variable f_i is independent of $\mathsf{\Pi}_{\le j}$ conditioned on $\mathsf{\Pi}_A$ and N,* i.e., *it holds for all $\Pi_{\le j}$ and $f_i \in \mathsf{supp}(\mathsf{f}_i)$ that:*

$$\Pr(f_i \mid \Pi_{\le j}, N) = \Pr(f_i \mid \Pi_A, N).$$

2. *The random variables* $\left(\{\mathsf{f}_{i'}\}_{i' \neq i \in [n]}, \Pi_A\right)$ *are mutually independent conditioned on* $\Pi_{\leq j'}$ *and* N, i.e., *it holds for all* $F_{-i} = \{f_{i'}\}_{i' \neq i \in [n]}$ *and* Π_A *that:*

$$\Pr(F_{-i}, \Pi_A \mid \Pi_{\leq j'}, N) = \Pr(\Pi_A \mid \Pi_{\leq j'}, N) \cdot \prod_{i' \neq i \in [n]} \Pr(f_{i'} \mid \Pi_{\leq j'}, N).$$

Proof. We prove each part in turn:

1. As $N \in \mathcal{N}(\{i\} \times ([j] \setminus A))$, the transcript received by party i in the first j rounds is determined by Π_A and N. In turn, $\mathtt{Rec}_i(\Pi_{\leq j}, N)$ is determined by Π_A and N. As

$$\begin{aligned}
\Pr(f_i \mid \Pi_{\leq j}, N) &= \Pr(f_i \mid \forall i' \in [n] : \mathsf{f}_{i'} \in \mathtt{Rec}_{i'}(\Pi_{\leq j}, N), N) && \text{(Lemma 5.7)} \\
&= \Pr(f_i \mid \mathsf{f}_i \in \mathtt{Rec}_i(\Pi_{\leq j}, N), N), \\
& && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i \in [n]} \text{ and } \mathsf{N})
\end{aligned}$$

   we can conclude that $\Pr(f_i \mid \Pi_{\leq j}, N)$ is determined by Π_A and N. The result follows.

2. As $N \in \mathcal{N}(\{i\} \times ([j] \setminus A))$, we have that Π_A is determined by f_i, $\Pi_{\leq j'}$, and N. Thus, conditioned on $\Pi_{\leq j'}, N$, Π_A is a function of f_i and it is enough to show that the random variables $\{\mathsf{f}_{i'}\}_{i' \in [n]}$ are mutually independent conditioned on $\Pi_{\leq j'}$ and N. This is because, for all F, we have:

$$\begin{aligned}
\Pr(F \mid \Pi_{\leq j'}, N) &= \Pr(F \mid \forall i' \in [n] : \mathsf{f}_{i'} \in \mathtt{Rec}_{i'}(\Pi_{\leq j'}, N), N) && \text{(Lemma 5.7)} \\
&= \prod_{i' \in [n]} \Pr(f_{i'} \mid \mathsf{f}_{i'} \in \mathtt{Rec}_{i'}(\Pi_{\leq j'}, N)) \\
& && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i \in [n]} \text{ and } \mathsf{N}) \\
&= \prod_{i' \in [n]} \Pr(f_{i'} \mid \forall i'' \in [n] : \mathsf{f}_{i''} \in \mathtt{Rec}_{i''}(\Pi_{\leq j'}, N), N) \\
& && \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i \in [n]} \text{ and } \mathsf{N}) \\
&= \prod_{i' \in [n]} \Pr(f_{i'} \mid \Pi_{\leq j'}, N). && \text{(Lemma 5.7)}
\end{aligned}$$

□

5.2.2 Entropy Bounds on the Parties' Inputs

Definition 5.12. *Let* $0 \leq j \leq T$ *be given. For all noise vectors* N *and* $i \in [n]$, *define the event:*

$$\mathcal{Z}^i_N(j) = \left\{ \Pi_{\leq j} \;\middle|\; \mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \Pi_{\leq j}, N) \;||\; \mathcal{U}) > 10 \cdot \log m \cdot \left|\mathsf{p}^{-1}(i) \cap [j]\right| \right\}.$$

Additionally, define $\mathcal{Z}_N(j) = \bigcup_{i \in [n]} \mathcal{Z}^i_N(j)$.

Lemma 5.13. *Let $0 \leq j \leq T$ be given. For all noise vectors N and $i \in [n]$, we have:*

$$\Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{supp}(\mathsf{f}_i)|} < \frac{1}{m^{10 \cdot |\mathtt{p}^{-1}(i) \cap [j]|}} \;\middle|\; N \right) \leq \frac{|\mathtt{p}^{-1}(i) \cap [j]|}{m^9}.$$

Proof. Proof by induction on j. The base case $j = 0$ is easy to see. We show the result for $j > 0$ assuming it holds for $j - 1$. We have:

$$\begin{aligned}
&\Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{supp}(\mathsf{f}_i)|} < \frac{1}{m^{10 \cdot |\mathtt{p}^{-1}(i) \cap [j]|}} \;\middle|\; N \right) \\
&\leq \Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{< j}, N)|}{|\mathsf{supp}(\mathsf{f}_i)|} < \frac{1}{m^{10 \cdot |\mathtt{p}^{-1}(i) \cap [j-1]|}} \;\middle|\; N \right) + \Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{Rec}_i(\Pi_{< j}, N)|} < \frac{1}{m^{10 \cdot \mathbb{1}(i = \mathtt{p}(j))}} \;\middle|\; N \right) \\
&\hfill \text{(Union bound)} \\
&\leq \frac{|\mathtt{p}^{-1}(i) \cap [j-1]|}{m^9} + \Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{Rec}_i(\Pi_{< j}, N)|} < \frac{1}{m^{10 \cdot \mathbb{1}(\mathtt{p}(j) = i)}} \;\middle|\; N \right). \qquad \text{(Induction hypothesis)}
\end{aligned}$$

If $\mathtt{p}(j) \neq i$, the lemma follows from the foregoing inequality and Definition 5.6. Otherwise, if $\mathtt{p}(j) = i$, and the lemma follows if we show that the rightmost term is upper bounded by $\frac{1}{m^9}$. Next, we show that this holds even when conditioned on an arbitrary value $\Pi_{<j}$. We have:

$$\begin{aligned}
\Pr\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{Rec}_i(\Pi_{< j}, N)|} < \frac{1}{m^{10}} \;\middle|\; \Pi_{<j}, N \right) &= \sum_{\Pi_j} \Pr\left(\Pi_j \;\middle|\; \Pi_{<j}, N \right) \cdot \mathbb{1}\left(\frac{|\mathsf{Rec}_i(\Pi_{\leq j}, N)|}{|\mathsf{Rec}_i(\Pi_{< j}, N)|} < \frac{1}{m^{10}} \right) \\
&= \sum_{\Pi_j} \Pr\left(\Pi_j \;\middle|\; \Pi_{<j}, N \right) \cdot \mathbb{1}\left(\Pr(\Pi_j \mid \Pi_{<j}, N) < \frac{1}{m^{10}} \right) \\
&\hfill \text{(Lemma 5.8)} \\
&\leq \frac{1}{m^9}.
\end{aligned}$$

□

Lemma 5.14. *Let $0 \leq j \leq T$ be given. For all noise vectors N and $i \in [n]$, we have:*

$$\Pr\left(\mathcal{Z}_N^i(j) \mid N \right) \leq \frac{1}{m^5}.$$

Proof. For all $\Pi_{\leq j} \in \mathcal{Z}_N^i(j)$, we have:

$$\begin{aligned}
&\mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \Pi_{\leq j}, N) \,||\, \mathcal{U}) > 10 \cdot \log m \cdot \left|\mathtt{p}^{-1}(i) \cap [j]\right| \\
&\quad \Longrightarrow \mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \forall i' \in [n] : \mathsf{f}_{i'} \in \mathsf{Rec}_{i'}(\Pi_{\leq j}, N), N) \,||\, \mathcal{U}) > 10 \cdot \log m \cdot \left|\mathtt{p}^{-1}(i) \cap [j]\right| \\
&\hfill \text{(Lemma 5.7)} \\
&\quad \Longrightarrow \mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \mathsf{f}_i \in \mathsf{Rec}_i(\Pi_{\leq j}, N)) \,||\, \mathcal{U}) > 10 \cdot \log m \cdot \left|\mathtt{p}^{-1}(i) \cap [j]\right| \\
&\hfill \text{(Mutual Independence of } \{\mathsf{f}_i\}_{i \in [n]} \text{ and } \mathsf{N})
\end{aligned}$$

$$\implies \frac{|\mathsf{Rec}_i(\Pi_{\le j}, N)|}{|\mathsf{supp}(\mathsf{f}_i)|} < \frac{1}{m^{10\cdot|\mathsf{p}^{-1}(i)\cap[j]|}}. \qquad \text{(Lemma 3.8)}$$

The lemma now follows from Lemma 5.13. □

Corollary 5.15. *Let $0 \le j \le T$ be given. For all noise vectors N, we have:*

$$\Pr(\mathcal{Z}_N(j) \mid N) \le \frac{1}{m^4}.$$

5.2.3 Properties of High Entropy Inputs

Definition 5.16. *Let $i > 1 \in [n]$ and $0 \le j \le T$ be given. For all $(\Pi_{\le j}, N)$, define the set:*

$$S^i_N(\Pi_{\le j}) = \left\{ z \in [m] \;\middle|\; \mathbb{D}(\mathsf{dist}(\mathsf{f}_i(z) \mid \Pi_{\le j}, N) \mid\mid \mathcal{U}) > \frac{1}{m^{0.65}} \right\}.$$

Lemma 5.17. *Consider $i > 1 \in [n]$, $0 \le j \le T$, and a noise vector N. For all $\Pi_{\le j} \notin \mathcal{Z}^i_N(j)$, we have*

$$\left|S^i_N(\Pi_{\le j})\right| \le m^{0.66}.$$

Proof. As $\Pi_{\le j} \notin \mathcal{Z}^i_N(j)$, we derive:

$$\begin{aligned}
10T \cdot \log m &\ge \mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \Pi_{\le j}, N) \mid\mid \mathcal{U}) && \text{(Definition 5.12)}\\
&\ge \sum_{z=1}^{m} \mathbb{D}(\mathsf{dist}(\mathsf{f}_i(z) \mid \Pi_{\le j}, N) \mid\mid \mathcal{U}) && \text{(Lemma 3.10)}\\
&\ge \sum_{z \in S^i_N(\Pi_{\le j})} \mathbb{D}(\mathsf{dist}(\mathsf{f}_i(z) \mid \Pi_{\le j}, N) \mid\mid \mathcal{U})\\
&\ge \left|S^i_N(\Pi_{\le j})\right| \cdot \frac{1}{m^{0.65}}. && \text{(Definition 5.16)}
\end{aligned}$$

The lemma follows as $10T \cdot \log m \le m^{0.01}$. □

Lemma 5.18. *Consider $0 \le j \le T$ and a noise vector N. For all $\Pi_{\le j} \notin \mathcal{Z}_N(j)$, $S \subseteq [m]$, and $i > 1 \in [n]$, we have:*

$$\Pr\left(\left|\mathsf{f}_i^{-1}(S)\right| \ge |S| + m^{0.67} \mid \Pi_{\le j}, N\right) \le \frac{1}{m^{0.33}}.$$

Proof. Let X be the indicator random variable that is 1 if and only if $\left|\mathsf{f}_i^{-1}(S)\right| \ge |S| + m^{0.67}$ and $p = \Pr(\mathsf{X} = 1 \mid \Pi_{\le j}, N)$. As $\Pi_{\le j} \notin \mathcal{Z}^i_N(j)$, we have:

$$\begin{aligned}
(m - 10T) \cdot \log m &\le \mathbb{H}(\mathsf{f}_i \mid \Pi_{\le j}, N) && \text{(Definition 5.12 and Lemma 3.9)}\\
&\le 1 + \mathbb{H}(\mathsf{f}_i \mid \mathsf{X}, \Pi_{\le j}, N) && \text{(Lemmas 3.3 and 3.5)}\\
&\le 1 + \sum_{x \in \{0,1\}} \Pr(\mathsf{X} = x \mid \Pi_{\le j}, N) \cdot \mathbb{H}(\mathsf{f}_i \mid \mathsf{X} = x, \Pi_{\le j}, N) && \text{(Definition 3.2)}
\end{aligned}$$

$$\leq 1 + (1-p)\cdot m\log m + p\cdot \mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N) \qquad \text{(Lemma 3.5)}$$
$$\leq m\log m + 1 - p\cdot(m\log m - \mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N)).$$

This rearranges to

$$p\cdot(m\log m - \mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N)) \leq 11T\cdot\log m \leq m^{0.01},$$

and to show the lemma it suffices to show that

$$\mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N) \leq m\log m - m^{0.34}.$$

We show this in the rest of the proof. Note that due to Lemma 3.5, it is enough to upper bound the number of different values of f_i such that $\mathsf{X}=1$. We do this as follows: There are (at most) m possible choices for $\left|f_i^{-1}(S)\right|$. For each value k of $\left|f_i^{-1}(S)\right|$, there are $\binom{m}{k}$ values of $f_i^{-1}(S)$. Once we fixed $f_i^{-1}(S)$ such that $\left|f_i^{-1}(S)\right| = k$, the number of choices of f_i is $|S|^k\cdot(m-|S|)^{m-k}$. Thus, we have:

$$\begin{aligned}
\mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N) &\leq \log m + \mathbb{H}\big(\mathsf{f}_i \mid \left|\mathsf{f}_i^{-1}(S)\right|, \mathsf{X}=1, \Pi_{\leq j}, N\big) && \text{(Lemma 3.5)}\\
&\leq \log m + \max_{k\geq |S|+m^{0.67}} \mathbb{H}\big(\mathsf{f}_i \mid \left|\mathsf{f}_i^{-1}(S)\right| = k, \mathsf{X}=1, \Pi_{\leq j}, N\big) \\
& && \text{(Definition 3.2)}\\
&\leq \log m + \max_{k\geq |S|+m^{0.67}} \log\binom{m}{k} + k\log(|S|) + (m-k)\cdot\log(m-|S|). \\
& && \text{(Lemma 3.5)}
\end{aligned}$$

To continue, let $\mathsf{h}(x) = x\log\frac{1}{x} + (1-x)\log\frac{1}{1-x}$ denote the binary entropy function. From the identity $\binom{m}{k} \leq 2^{\mathsf{h}(k/m)\cdot m}$, we have:

$$\begin{aligned}
&\mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\leq j}, N)\\
&\quad\leq \log m + \max_{k\geq |S|+m^{0.67}} m\cdot\mathsf{h}(k/m) + k\log(|S|) + (m-k)\cdot\log(m-|S|)\\
&\quad\leq (m+1)\cdot\log m + \max_{k\geq |S|+m^{0.67}} m\cdot\mathsf{h}(k/m) + k\log\left(\frac{|S|}{m}\right) + (m-k)\cdot\log\left(1-\frac{|S|}{m}\right)\\
&\quad\leq (m+1)\cdot\log m + \max_{k\geq |S|+m^{0.67}} k\log\frac{|S|/m}{k/m} + (m-k)\cdot\log\frac{1-|S|/m}{1-k/m}\\
& && \text{(Definition of } \mathsf{h}(\cdot))\\
&\quad\leq (m+1)\cdot\log m + \max_{k\geq |S|+m^{0.67}} \left(-m\cdot\left(\frac{k}{m}\log\frac{k/m}{|S|/m} + \left(1-\frac{k}{m}\right)\cdot\log\frac{1-k/m}{1-|S|/m}\right)\right).
\end{aligned}$$

Now observe that the term inside the max is just the KL divergence (see Definition 3.7) between the Bernoulli distribution with parameter k/m and the Bernoulli distribution with

parameter $|S|/m$. We apply Fact 3.12 to get:

$$\begin{aligned}\mathbb{H}(\mathsf{f}_i \mid \mathsf{X}=1, \Pi_{\le j}, N) &\le (m+1)\cdot\log m + \max_{k\ge|S|+m^{0.67}}\left(-2m\cdot\left(\frac{k-|S|}{m}\right)^2\right)\\ &\le (m+1)\cdot\log m - \min_{k\ge|S|+m^{0.67}}\frac{2\cdot(k-|S|)^2}{m}\\ &\le (m+1)\cdot\log m - 2\cdot m^{0.34}\\ &\le m\log m - m^{0.34}.\end{aligned}$$

□

Lemma 5.19. *Consider $0 \le j \le T$ and a noise vector N. For all $\Pi_{\le j} \notin \mathcal{Z}_N(j)$, $S \subseteq [m]$, and $1 < i' \le i \le n$, we have:*

$$\Pr\left(\left|\mathsf{f}_{i'}^{-1}\left(\cdots\left(\mathsf{f}_i^{-1}(S)\right)\right)\right| \ge |S| + m^{0.67}\cdot(i+1-i') \mid \Pi_{\le j}, N\right) \le \frac{i+1-i'}{m^{0.33}}.$$

Proof. Proof by induction of $i-i'$. The base case $i=i'$ is due to Lemma 5.18. We show the lemma for $i>i'$ by assuming it holds for $i'+1$. For notational convenience, define the events:

$$\begin{aligned}\mathcal{E}_{i'} &\equiv \left|\mathsf{f}_{i'}^{-1}\left(\cdots\left(\mathsf{f}_i^{-1}(S)\right)\right)\right| \ge |S| + m^{0.67}\cdot(i+1-i').\\ \mathcal{E}_{i'+1} &\equiv \left|\mathsf{f}_{i'+1}^{-1}\left(\cdots\left(\mathsf{f}_i^{-1}(S)\right)\right)\right| \ge |S| + m^{0.67}\cdot(i-i').\end{aligned}$$

We have to bound:

$$\begin{aligned}\Pr(\mathcal{E}_{i'} \mid \Pi_{\le j}, N) &\le \Pr(\mathcal{E}_{i'+1} \mid \Pi_{\le j}, N) + \Pr\left(\mathcal{E}_{i'} \mid \overline{\mathcal{E}_{i'+1}}, \Pi_{\le j}, N\right)\\ &\le \frac{i-i'}{m^{0.33}} + \Pr\left(\mathcal{E}_{i'} \mid \overline{\mathcal{E}_{i'+1}}, \Pi_{\le j}, N\right). \qquad \text{(Induction hypothesis)}\end{aligned}$$

It is thus enough to bound the rightmost term by $\frac{1}{m^{0.33}}$. We shall show this bound even under the stronger conditioning of $f_{i'+1}, \cdots, f_i, \Pi_{\le j}, N$ for any $f_{i'+1}, \cdots, f_i$ such that $\mathcal{E}_{i'+1}$ does not happen. Letting $S' = f_{i'+1}^{-1}\left(\cdots\left(f_i^{-1}(S)\right)\right)$, we have:

$$\begin{aligned}\Pr(\mathcal{E}_{i'} \mid f_{i'+1}, \cdots, f_i, \Pi_{\le j}, N) &\le \Pr\left(\left|\mathsf{f}_{i'}^{-1}(S')\right| \ge |S| + m^{0.67}\cdot(i+1-i') \mid f_{i'+1}, \cdots, f_i, \Pi_{\le j}, N\right)\\ &\le \Pr\left(\left|\mathsf{f}_{i'}^{-1}(S')\right| \ge |S'| + m^{0.67} \mid f_{i'+1}, \cdots, f_i, \Pi_{\le j}, N\right)\\ &\qquad\qquad \text{(As } \mathcal{E}_{i'+1} \text{ does not happen)}\\ &\le \Pr\left(\left|\mathsf{f}_{i'}^{-1}(S')\right| \ge |S'| + m^{0.67} \mid \Pi_{\le j}, N\right)\\ &\quad \text{(Lemma 5.7 and the mutual independence of } \{\mathsf{f}_i\}_{i\in[n]} \text{ and } \mathsf{N})\\ &\le \frac{1}{m^{0.33}}. \qquad \text{(Lemma 5.18)}\end{aligned}$$

□

Corollary 5.20. *Consider $0 \le j \le T$ and a noise vector N. For all $\Pi_{\le j} \notin \mathcal{Z}_N(j)$, $S \subseteq [m]$, and $1 < i' \le i \le n$, we have:*

$$\Pr\left(\left|\mathsf{f}_{i'}^{-1}\left(\cdots\left(\mathsf{f}_i^{-1}(S)\right)\right)\right| \ge |S| + m^{0.68} \mid \Pi_{\le j}, N\right) \le \frac{1}{m^{0.32}}.$$

5.2.4 Entropy Bounds on the Output

Definition 5.21. *Let $0 \le j \le T$ and a noise vector N be given. For all $i \in [n]$, define the event:*

$$\mathcal{Y}_N^i(j) = \begin{cases} \left\{(\Pi_{\le j}, F_{<i}) \;\middle|\; \mathbb{D}(\mathsf{dist}(\mathsf{f}_i \mid \Pi_{\le j}, N) \mid\mid \mathcal{U}) > \frac{1}{m^{0.65}}\right\}, & \text{if } i = 1 \\ \left\{(\Pi_{\le j}, F_{<i}) \;\middle|\; \mathtt{PC}(F_{<i}) \in S_N^i(\Pi_{\le j})\right\}, & \text{if } i > 1 \end{cases}.$$

Observation 5.22. *For all noise vectors N and $i \in [n]$, we have $\mathcal{Y}_N^i(0) = \emptyset$.*

Before the next lemma, it may be helpful to recall the definition of $\mathtt{Crash}(\cdot)$ in Definition 5.3.

Lemma 5.23. *Let $\mathcal{P} \subseteq [n]$ and $\mathcal{R} = \{j \in [T] \mid \mathtt{p}(j) \in \mathcal{P}\}$. For all $j \in \mathcal{R}$, all crashing sets $C \in \mathtt{Crash}(j)$, and all noise vectors $N \in \mathcal{N}(C)$, we have:*

$$\Pr\left(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid N\right) \le \frac{j}{m^{0.3}}.$$

Proof. Proof by induction on $\mathtt{p}(j)$. For the base case, not that for all j such that $\mathtt{p}(j) = \mathcal{P}(1)$, we have $\mathtt{Crash}(j) = \emptyset$ by Definition 5.3 and there is nothing to show. For the inductive step, we take an arbitrary $p > 1$ and show the result for all j such that $\mathtt{p}(j) = \mathcal{P}(p)$ assuming it holds for all j' such that $\mathtt{p}(j') = \mathcal{P}(p-1)$. As $C \in \mathtt{Crash}(j)$, we have from Definition 5.3 that there exists a value $b \in [B_{p-1,j}]$ and a set $C' \in \mathtt{Crash}(\mathcal{B}_{p-1,j}(b))$ satisfying

$$C' \cup \{(\mathtt{p}(j), j'') \mid \mathcal{B}_{p-1,j}(b+1) \le j'' < j\} \subseteq C.$$

Let $j' = \mathcal{B}_{p-1,j}(b)$ for convenience. Observe that $N \in \mathcal{N}(C')$ and $j' < j$. We first claim that:

$$\Pr\left(\mathcal{Y}_N^{\mathcal{P}(p-1)}(j') \mid N\right) \le \frac{j-1}{m^{0.3}}. \tag{7}$$

Indeed, if $j' = 0$, then Eq. (7) is due to Observation 5.22 and otherwise, it is by the induction hypothesis on j' (note that in the latter case, we have $\mathtt{p}(j') = \mathcal{P}(p-1)$). Next, define $j^* = \mathcal{B}_{p-1,j}(b+1) - 1$ and the event $\mathcal{E}_{\mathsf{Bad}} = \mathcal{Y}_N^{\mathcal{P}(p-1)}(j') \vee \mathcal{Z}_N^{\mathtt{p}(j)}(j) \vee \mathcal{Z}_N(j^*)$ and note by a union bound that:

$$\begin{aligned} \Pr(\mathcal{E}_{\mathsf{Bad}} \mid N) &\le \Pr\left(\mathcal{Y}_N^{\mathcal{P}(p-1)}(j') \mid N\right) + \Pr\left(\mathcal{Z}_N^{\mathtt{p}(j)}(j) \mid N\right) + \Pr(\mathcal{Z}_N(j^*) \mid N) \\ &\le \frac{j - 1/2}{m^{0.3}}. && \text{(Eq. (7), Lemma 5.14, and Corollary 5.15)} \end{aligned}$$

Moreover, letting $A = [j^*] \cup ([j] \cap \mathtt{p}^{-1}(\mathtt{p}(j)))$, note that $j' \le j^* < j$ and any value of the pair $\big(\Pi_A, F_{<\mathcal{P}(p-1)}\big)$ determines whether or not the $\mathcal{E}_{\mathsf{Bad}}$ happens. Indeed, any value of the pair $\big(\Pi_A, F_{<\mathcal{P}(p-1)}\big)$ determines whether or not $\mathcal{Y}_N^{\mathcal{P}(p-1)}(j')$ and $\mathcal{Z}_N(j^*)$ happen as $j' \le j^*$, and also determines whether or not $\mathcal{Z}_N^{\mathtt{p}(j)}(j)$ happens due to Item 1 of Lemma 5.11. Combining, we get that it determines whether or not $\mathcal{E}_{\mathsf{Bad}}$ happens, as claimed. We now derive:

$$\begin{aligned}\Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid N\Big) &\le \Pr(\mathcal{E}_{\mathsf{Bad}} \mid N) + \Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid \overline{\mathcal{E}_{\mathsf{Bad}}}, N\Big) \\ &\le \frac{j-1/2}{m^{0.3}} + \Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid \overline{\mathcal{E}_{\mathsf{Bad}}}, N\Big).\end{aligned}$$

It is thus enough to upper bound the last term by $\frac{3}{m^{0.31}}$. As any value of the pair $\big(\Pi_A, F_{<\mathcal{P}(p-1)}\big)$ determines whether or not the $\mathcal{E}_{\mathsf{Bad}}$ happens, it suffices to do this conditioned on an arbitrary value of $\big(\Pi_A, F_{<\mathcal{P}(p-1)}\big)$ such $\mathcal{E}_{\mathsf{Bad}}$ does not happen. Letting $\mathcal{E} = \Pi_A, F_{<\mathcal{P}(p-1)}, N$, we have from Definition 5.21 that:

$$\Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid \mathcal{E}\Big) = \Pr\Big(\mathsf{PC}\big(\mathsf{F}_{<\mathtt{p}(j)}\big) \in S_N^{\mathtt{p}(j)}(\mathsf{\Pi}_{\le j}) \mid \mathcal{E}\Big).$$

Due to Item 1 of Lemma 5.11, fixing Π_A also fixes the value of $S_N^{\mathtt{p}(j)}(\mathsf{\Pi}_{\le j})$. Denoting this by $S_N^{\mathtt{p}(j)}(\Pi_A)$, we have:

$$\begin{aligned}&\Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid \mathcal{E}\Big) \\ &\quad = \Pr\Big(\mathsf{PC}\big(\mathsf{F}_{<\mathtt{p}(j)}\big) \in S_N^{\mathtt{p}(j)}(\Pi_A) \mid \mathcal{E}\Big) \\ &\quad = \Pr\Big(\mathsf{f}_{\mathtt{p}(j)-1}\big(\cdots\big(\mathsf{f}_{\mathcal{P}(p-1)}\big(\mathsf{PC}\big(F_{<\mathcal{P}(p-1)}\big)\big)\big)\big) \in S_N^{\mathtt{p}(j)}(\Pi_A) \mid \Pi_{\le j^*}, N\Big) \\ &\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{(Lemma 5.11, Item 2)} \\ &\quad = \Pr\Big(\mathsf{f}_{\mathcal{P}(p-1)}\big(\mathsf{PC}\big(F_{<\mathcal{P}(p-1)}\big)\big) \in \mathsf{f}^{-1}_{\mathcal{P}(p-1)+1}\Big(\cdots\Big(\mathsf{f}^{-1}_{\mathtt{p}(j)-1}\Big(S_N^{\mathtt{p}(j)}(\Pi_A)\Big)\Big)\Big) \mid \Pi_{\le j^*}, N\Big) \\ &\quad = \sum_{S\subseteq[m]} \Pr\Big(\mathsf{f}_{\mathcal{P}(p-1)}\big(\mathsf{PC}\big(F_{<\mathcal{P}(p-1)}\big)\big) \in \mathsf{f}^{-1}_{\mathcal{P}(p-1)+1}\Big(\cdots\Big(\mathsf{f}^{-1}_{\mathtt{p}(j)-1}\Big(S_N^{\mathtt{p}(j)}(\Pi_A)\Big)\Big)\Big) = S \mid \Pi_{\le j^*}, N\Big).\end{aligned}$$

To continue, we use Lemma 5.7 and the mutual independence of $\{\mathsf{f}_i\}_{i\in[n]}$ and N. We get:

$$\begin{aligned}\Pr\Big(\mathcal{Y}_N^{\mathtt{p}(j)}(j) \mid \mathcal{E}\Big) = \sum_{S\subseteq[m]} &\Pr\big(\mathsf{f}_{\mathcal{P}(p-1)}\big(\mathsf{PC}\big(F_{<\mathcal{P}(p-1)}\big)\big) \in S \mid \Pi_{\le j^*}, N\big) \\ &\times \Pr\Big(\mathsf{f}^{-1}_{\mathcal{P}(p-1)+1}\Big(\cdots\Big(\mathsf{f}^{-1}_{\mathtt{p}(j)-1}\Big(S_N^{\mathtt{p}(j)}(\Pi_A)\Big)\Big)\Big) = S \mid \Pi_{\le j^*}, N\Big).\end{aligned}$$

Now, as $\mathcal{Z}_N^{\mathtt{p}(j)}(j)$ does not happen, we have by Lemma 5.17 that $\Big|S_N^{\mathtt{p}(j)}(\Pi_A)\Big| \le m^{0.66}$. As $\mathcal{Z}_N(j^*)$ does not happen, due to Corollary 5.20, we can upper bound the terms corresponding

to S such that $|S| \geq m^{0.69}$ by $\frac{1}{m^{0.32}}$. We get:

$$\Pr\left(\mathcal{Y}_N^{\mathsf{p}(j)}(j) \mid \mathcal{E}\right) \leq \frac{1}{m^{0.32}} + \max_{S \subseteq [m]: |S| \leq m^{0.69}} \Pr\left(\mathsf{f}_{\mathcal{P}(p-1)}\left(\mathtt{PC}\left(F_{<\mathcal{P}(p-1)}\right)\right) \in S \mid \Pi_{\leq j^*}, N\right).$$

Using Lemma 5.9 on the second term, we get:

$$\Pr\left(\mathcal{Y}_N^{\mathsf{p}(j)}(j) \mid \mathcal{E}\right) \leq \frac{1}{m^{0.32}} + \max_{S \subseteq [m]: |S| \leq m^{0.69}} \Pr\left(\mathsf{f}_{\mathcal{P}(p-1)}\left(\mathtt{PC}\left(F_{<\mathcal{P}(p-1)}\right)\right) \in S \mid \Pi_{\leq j'}, N\right).$$

Finally, as $\mathcal{Y}_N^{\mathcal{P}(p-1)}(j')$ does not happen, we have $\mathbb{D}(\mathsf{dist}\left(\mathsf{f}_{\mathcal{P}(p-1)}\left(\mathtt{PC}\left(F_{<\mathcal{P}(p-1)}\right)\right)\right) \mid \Pi_{\leq j'}, N) \,||\, \mathcal{U}) \leq \frac{1}{m^{0.65}}$. Combining with Corollary 3.13, we get:

$$\begin{aligned}
\Pr\left(\mathcal{Y}_N^{\mathsf{p}(j)}(j) \mid \mathcal{E}\right) &\leq \frac{1}{m^{0.32}} + \max_{S \subseteq [m]: |S| \leq m^{0.69}} \frac{|S|}{m} + \sqrt{\frac{1}{2} \cdot \frac{1}{m^{0.65}}} \\
&\leq \frac{1}{m^{0.32}} + \frac{m^{0.69}}{m} + \frac{1}{m^{0.32}} \\
&\leq \frac{3}{m^{0.31}}.
\end{aligned}$$

□

The following is the main result of Section 5.2:

Lemma 5.24. *For any noise vector N for which there exists $\mathcal{P} \subseteq [n]$ and a set C such that $n \in \mathcal{P}$, $C \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}}(T)$, where $\mathcal{R} = \{j \in [T] \mid \mathsf{p}(j) \in \mathcal{P}\}$, and $N \in \mathcal{N}(C)$, we have:*

$$\Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid N) \leq \frac{1}{2} + \frac{1}{m^{0.1}}.$$

Proof. We have:

$$\begin{aligned}
\Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid N) &\leq \Pr(\mathcal{Y}_N^n(T) \mid N) + \Pr\left(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid \overline{\mathcal{Y}_N^n(T)}, N\right) \\
&\qquad \text{(Union bound)} \\
&\leq \frac{1}{m^{0.2}} + \Pr\left(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid \overline{\mathcal{Y}_N^n(T)}, N\right). \quad \text{(Lemma 5.23)}
\end{aligned}$$

It is thus enough to bound the last term by $\frac{1}{2} + \frac{1}{m^{0.2}}$. We shall in fact show bound even when conditioned on an arbitrary $\Pi_{\leq T}, F_{<n}$ such that $\mathcal{Y}_N^n(T)$ does not happen. We have:

$$\begin{aligned}
\Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid \Pi_{\leq T}, F_{<n}, N) &\leq \Pr(\mathtt{LSB}(\mathsf{f}_n(\mathtt{PC}(F_{<n}))) = \Pi \mid \Pi_{\leq T}, F_{<n}, N) \\
&\qquad \text{(As } (\Pi_{\leq T}, N) \text{ determines } \Pi(\mathsf{F})) \\
&\leq \Pr(\mathtt{LSB}(\mathsf{f}_n(\mathtt{PC}(F_{<n}))) = \Pi \mid \Pi_{\leq T}, N). \\
&\qquad \text{(Lemma 5.7 and the mutual independence of } \{\mathsf{f}_i\}_{i \in [n]} \text{ and } \mathsf{N})
\end{aligned}$$

As $\mathcal{Y}_N^n(T)$ does not happen, we have $\mathtt{PC}(F_{<n}) \notin S_N^n(\Pi_{\leq T})$ implying by Definition 5.16 that $\mathbb{D}(\mathsf{dist}(\mathsf{f}_n(\mathtt{PC}(F_{<n})) \mid \Pi_{\leq T}, N) \mid\mid \mathcal{U}) \leq \frac{1}{m^{0.65}}$. We get from Corollary 3.13 and the fact that m is even that:

$$\Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid \Pi_{\leq T}, F_{<n}, N) \leq \frac{1}{2} + \frac{1}{m^{0.2}}.$$

□

5.3 A Lower Bound in the Adversarial Model

5.3.1 Properties of Crash

The following properties of our definitions in Section 5.1 will be useful for us.

Lemma 5.25. *Let $(\mathcal{P}, \mathcal{R})$ be a nice pair. For all $r' \leq r \in \mathcal{R}$ such that $\mathtt{p}(r) \leq \mathtt{p}(r')$, we have $\mathtt{t}(r') \leq \mathtt{t}(r)$.*

Proof. Proof by contradiction. Suppose there is a counterexample and set r', r to be the counterexample with the (lexicographically) smallest value of $(\mathtt{p}(r), \mathtt{p}(r'))$. Let p, p' be such that $(\mathtt{p}(r), \mathtt{p}(r')) = (\mathcal{P}(p), \mathcal{P}(p'))$. These are well defined as $(\mathcal{P}, \mathcal{R})$ is a nice pair. As $\mathtt{t}(r') > \mathtt{t}(r)$, we must have $\mathtt{t}(r) < \infty \implies p > 1$. We first claim that:

Claim 5.26. $p = p'$.

Proof. If not, then we have $p < p'$ and $\mathtt{t}(r') > \mathtt{t}(r)$. Letting $b = B_{p'-1,r'}$, observe from Definition 5.3 that $\mathtt{Crash}(\mathcal{B}_{p'-1,r'}(b)) \subseteq \mathtt{Crash}(r')$ which implies $\mathtt{t}(r') \leq \mathtt{t}(\mathcal{B}_{p'-1,r'}(b))$. Now, either $\mathcal{B}_{p'-1,r'}(b) = 0$ which means that $\mathtt{t}(r') \leq \mathtt{t}(\mathcal{B}_{p'-1,r'}(b)) = 0$ and contradicts $\mathtt{t}(r') > \mathtt{t}(r)$, or we have $\mathtt{p}(\mathcal{B}_{p'-1,r'}(b)) = \mathcal{P}(p'-1)$ by Eq. (6) and $\mathtt{t}(r) < \mathtt{t}(r') \leq \mathtt{t}(\mathcal{B}_{p'-1,r'}(b))$ contradicts the choice of r, r'. □

As $\mathtt{t}(r) < \infty$, there is a set $C \in \mathtt{Crash}(r)$ such that $|C| = \mathtt{t}(r)$. Fix such a C and use Definition 5.3 to get that their exists $b \in [B_{p-1,r}]$ and a set $C' \in \mathtt{Crash}(\mathcal{B}_{p-1,r}(b))$ satisfying

$$C' \cup \{(\mathcal{P}(p), r'') \mid \mathcal{B}_{p-1,r}(b+1) \leq r'' < r\} \subseteq C.$$

Claim 5.27. $\mathcal{B}_{p-1,r}(b) < r'$.

Proof. If not, then we have $r' \leq \mathcal{B}_{p-1,r}(b)$ and also (from Eq. (6)) that $\mathtt{p}(\mathcal{B}_{p-1,r}(b)) = \mathcal{P}(p-1) < \mathtt{p}(r)$. By our choice choice of r, r', this must mean that $\mathtt{t}(r') \leq \mathtt{t}(\mathcal{B}_{p-1,r}(b)) \leq |C'| \leq |C| = \mathtt{t}(r)$, a contradiction. □

Conclude from Claim 5.27 that there exists $b' \in [B_{p-1,r'}]$ such that $\mathcal{B}_{p-1,r}(b) = \mathcal{B}_{p-1,r'}(b')$. From Eq. (6), note that this means either $\mathcal{B}_{p-1,r'}(b'+1) = \mathcal{B}_{p-1,r}(b+1)$ or $\mathcal{B}_{p-1,r'}(b'+1) = r'$. In either case, we can conclude that $C \in \mathtt{Crash}(r')$ implying $\mathtt{t}(r') \leq |C| = \mathtt{t}(r)$, a contradiction.

□

Lemma 5.28. *Let $(\mathcal{P}, \mathcal{R})$ be a nice pair and $p' < p \in [P]$. For all $r \in \mathcal{R} \cap \mathtt{p}^{-1}(\mathcal{P}(p))$, we have:*

1. $\mathtt{t}(r) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r}))$.
2. *If $B_{p',r} > 1$, then* $\mathtt{t}(r) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r} - 1)) + r - \mathcal{B}_{p',r}(B_{p',r})$.

Proof. We prove each part in turn:

1. Proof by contradiction. Consider a counterexample with the smallest $p - p'$. We claim that, in fact, $p - p' = 1$. Indeed, if not, define $\hat{r} = \mathcal{B}_{p-1,r}(B_{p-1,r})$ and use our choice of p, p' to conclude that $\mathtt{t}(r) \leq \mathtt{t}(\hat{r})$. If $\hat{r} \leq \mathcal{B}_{p',r}(B_{p',r})$, then Lemma 5.25 says $\mathtt{t}(r) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r}))$, a contradiction. Else, we have $\mathcal{B}_{p',r}(B_{p',r}) < \hat{r} \leq r$ implying that $\mathcal{B}_{p',\hat{r}}(B_{p',\hat{r}}) = \mathcal{B}_{p',r}(B_{p',r})$. We again use our choice of p, p' to get $\mathtt{t}(r) \leq \mathtt{t}(\hat{r}) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r}))$, a contradiction.

   Having shown that $p - p' = 1$, we use Definition 5.3 to get $\mathtt{Crash}(\mathcal{B}_{p',r}(B_{p',r})) \subseteq \mathtt{Crash}(r)$, a contradiction.

2. Proof by contradiction. Consider a counterexample with the smallest $p - p'$. We claim that, in fact, $p - p' = 1$. Indeed, if not, define $\hat{r} = \mathcal{B}_{p-1,r}(B_{p-1,r})$ and use Item 1 to get $\mathtt{t}(r) \leq \mathtt{t}(\hat{r})$. Consider the following three cases:

   - **When $\hat{r} \leq \mathcal{B}_{p',r}(B_{p',r} - 1)$:** From Lemma 5.25, we get that $\mathtt{t}(r) \leq \mathtt{t}(\hat{r}) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r} - 1))$, a contradiction.
   - **When $\mathcal{B}_{p',r}(B_{p',r} - 1) < \hat{r} \leq \mathcal{B}_{p',r}(B_{p',r})$:** Observe that, when this happens, we have $\mathcal{B}_{p',r}(B_{p',r} - 1) = \mathcal{B}_{p',\hat{r}}(B_{p',\hat{r}})$. From Item 1, we get that $\mathtt{t}(r) \leq \mathtt{t}(\hat{r}) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r} - 1))$, a contradiction.
   - **When $\mathcal{B}_{p',r}(B_{p',r}) < \hat{r}$:** Observe that, when this happens, we have $\mathcal{B}_{p',r} = \mathcal{B}_{p',\hat{r}}$. From our choice of p, p', we get that $\mathtt{t}(r) \leq \mathtt{t}(\hat{r}) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r} - 1)) + \hat{r} - \mathcal{B}_{p',r}(B_{p',r})$, a contradiction as $\hat{r} < r$.

   Having shown that $p - p' = 1$, it is easy to see from Definition 5.3 that $\mathtt{t}(r) \leq \mathtt{t}(\mathcal{B}_{p',r}(B_{p',r} - 1)) + r - \mathcal{B}_{p',r}(B_{p',r})$, a contradiction.

□

Lemma 5.29. *Let $\mathcal{P} \subseteq [n]$ be non-empty and $i^* = \max(\mathcal{P})$. Define $\mathcal{P}' = \mathcal{P} \cup ([n] \setminus [i^*])$ and:*

$$\mathcal{R} = \{r \in [T] \mid \mathtt{p}(r) \in \mathcal{P}\} \qquad \text{and} \qquad \mathcal{R}' = \{r \in [T] \mid \mathtt{p}(r) \in \mathcal{P}'\}.$$

We have:

$$\mathtt{Crash}_{\mathcal{P},\mathcal{R}}(\max(\mathcal{R})) \subseteq \mathtt{Crash}_{\mathcal{P}',\mathcal{R}'}(\max(\mathcal{R}')).$$

Proof. By our assumption that $\mathtt{p}(T + 1 - i) = n + 1 - i$ for all $i \in [n]$, we have that $\max(\mathcal{R}) = T + i^* - n$ and $\max(\mathcal{R}') = T$. The lemma follows from the following claims:

Claim 5.30. *For $r \in \{0\} \cup \mathcal{R}$, we have $\texttt{Crash}_{\mathcal{P},\mathcal{R}}(r) \subseteq \texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(r)$.*

Proof. Proof by induction. The base case $r = 0$ is straightforward from Definition 5.3. We show the result for $r > 0$ by assuming it holds for smaller values of r. As $\texttt{Crash}_{\mathcal{P},\mathcal{R}}(r)$ is empty otherwise, we can assume that $\texttt{p}(r) = \mathcal{P}(p)$ for some $p > 1$. Then, for any $C \in \texttt{Crash}_{\mathcal{P},\mathcal{R}}(r)$, there exists a value $b \in [B_{\mathcal{P},\mathcal{R},p-1,r}]$ and a set $C' \in \texttt{Crash}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b))$ satisfying

$$C' \cup \{(\mathcal{P}(p), r') \mid \mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b+1) \leq r' < r\} \subseteq C.$$

Now, note that $\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b) < r$ and we can apply the induction hypothesis on $\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b)$ to get $\texttt{Crash}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b)) \subseteq \texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b))$. Also, note by Eq. (6) and our definitions of $\mathcal{P}', \mathcal{R}'$ that $\mathcal{P}(p) = \mathcal{P}'(p)$ and $\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r} = \mathcal{B}_{\mathcal{P}',\mathcal{R}',p-1,r}$. The induction step now follows from another application of Definition 5.3. □

Claim 5.31. *For $r \geq \max(\mathcal{R})$, we have $\texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(\max(\mathcal{R})) \subseteq \texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(r)$.*

Proof. Proof by induction. The base case $r = \max(\mathcal{R})$ is straightforward. We show the result for $r > \max(\mathcal{R})$ by assuming it holds for $r - 1$. As $\texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(\max(\mathcal{R}))$ we can assume that $\texttt{p}(\max(\mathcal{R})) \neq \mathcal{P}'(1)$. It follows that $\texttt{p}(r-1) \neq \mathcal{P}'(1)$. Let $p > 1$ be such that $\texttt{p}(r-1) \neq \mathcal{P}'(p)$. Then, by Definition 5.3 and the fact that $\mathcal{B}_{\mathcal{P}',\mathcal{R}',p-1,r}(B_{\mathcal{P}',\mathcal{R}',p-1,r}) = r - 1$, we have that $\texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(r-1) \subseteq \texttt{Crash}_{\mathcal{P}',\mathcal{R}'}(r)$ and claim follows from the induction hypothesis. □

□

5.3.2 Strength of a Pair

For a subset $\mathcal{R} \subseteq [T]$ and $i \in [n]$, we define the notation $m_{\mathcal{R},i} = \max(\mathcal{R} \cap \texttt{p}^{-1}(i))$ with the convention that $m_{\mathcal{R},i} = 0$ if $\mathcal{R} \cap \texttt{p}^{-1}(i) = \emptyset$.

Definition 5.32 (Strength)**.** *Let $(\mathcal{P}, \mathcal{R})$ be a nice pair. Define the strength $\texttt{Str}_{\mathcal{P},\mathcal{R}}$ of the pair as $\texttt{Str}_{\mathcal{P},\mathcal{R}} = \min_{p \in [P]} \texttt{Str}_{\mathcal{P},\mathcal{R}}(p)$ where:*

$$\texttt{Str}_{\mathcal{P},\mathcal{R}}(p) = \texttt{t}_{\mathcal{P},\mathcal{R}}\big(m_{\mathcal{R},\mathcal{P}(p)}\big).$$

Lemma 5.33. *Let $(\mathcal{P}, \mathcal{R})$ be a nice pair such that $\mathcal{R} \neq \emptyset$. For $p \in [P]$ satisfying $\mathcal{P}(p) \leq \texttt{p}(\max(\mathcal{R}))$, we have:*

$$\texttt{t}_{\mathcal{P},\mathcal{R}}(\max(\mathcal{R})) \leq \texttt{t}_{\mathcal{P},\mathcal{R}}\big(m_{\mathcal{R},\mathcal{P}(p)}\big).$$

Proof. Proof by contradiction. Let $p \in [P]$ be the largest counterexample. We derive a contradiction by showing that $\mathcal{P}(p) = \texttt{p}(\max(\mathcal{R}))$. Indeed, this means that $m_{\mathcal{R},\mathcal{P}(p)} = m_{\mathcal{R},\texttt{p}(\max(\mathcal{R}))} = \max(\mathcal{R})$.

Claim 5.34. $\mathcal{P}(p) = \texttt{p}(\max(\mathcal{R}))$.

Proof. If not, then our choice of p implies that $\mathtt{t}\big(m_{\mathcal{R},\mathcal{P}(p)}\big) < \mathtt{t}(\max(\mathcal{R})) \leq \mathtt{t}\big(m_{\mathcal{R},\mathcal{P}(p+1)}\big)$. By the contrapositive of Lemma 5.25, we get that $m_{\mathcal{R},\mathcal{P}(p+1)} > m_{\mathcal{R},\mathcal{P}(p)}$. We derive a contradiction by showing that $\mathtt{Crash}\big(m_{\mathcal{R},\mathcal{P}(p)}\big) \subseteq \mathtt{Crash}\big(m_{\mathcal{R},\mathcal{P}(p+1)}\big)$. This is due to Definition 5.3 when applied with $r = m_{\mathcal{R},\mathcal{P}(p+1)}, b = B_{p,m_{\mathcal{R},\mathcal{P}(p+1)}}$. □

□

Lemma 5.35. *Let $(\mathcal{P},\mathcal{R})$ be a nice pair and $t \leq \mathtt{Str}_{\mathcal{P},\mathcal{R}}$. We then have $t \leq \mathtt{Str}_{\mathcal{P},\mathcal{R}'}$, where the set $\mathcal{R}'$ is defined as:*

$$\mathcal{R}' = \big\{r \in \mathcal{R} \mid \forall r' < r \in \mathcal{R} \cap \mathtt{p}^{-1}(\mathtt{p}(r)) : \mathtt{t}_{\mathcal{P},\mathcal{R}}(r') < t\big\}.$$

Proof. We can assume that $t > 0$ as the lemma is trivial otherwise. Conclude from $t \leq \mathtt{Str}_{\mathcal{P},\mathcal{R}}$ and Definition 5.32 that $t \leq \mathtt{t}_{\mathcal{P},\mathcal{R}}\big(m_{\mathcal{R},\mathcal{P}(p)}\big)$ for all $p \in [P]$. Thus, for all $p \in [P]$, there exists at least one element r' in $\mathcal{R} \cap \mathtt{p}^{-1}(\mathcal{P}(p))$ such that $\mathtt{t}_{\mathcal{P},\mathcal{R}}(r')$ is at least t. We first claim that the smallest such element is $m_{\mathcal{R}',\mathcal{P}(p)}$.

Claim 5.36. *For all $p \in [P]$, we have $m_{\mathcal{R}',\mathcal{P}(p)} = \min\{r' \in \mathcal{R} \cap \mathtt{p}^{-1}(\mathcal{P}(p)) \mid t \leq \mathtt{t}_{\mathcal{P},\mathcal{R}}(r')\}$. Moreover, for all $r \in \mathtt{p}^{-1}(\mathcal{P}(p))$, we have $r \in \mathcal{R} \wedge r \leq m_{\mathcal{R}',\mathcal{P}(p)} \iff r \in \mathcal{R}'$.*

Proof. Let $m_p^* = \min\{r' \in \mathcal{R} \cap \mathtt{p}^{-1}(\mathcal{P}(p)) \mid t \leq \mathtt{t}_{\mathcal{P},\mathcal{R}}(r')\}$ for convenience. We first show that $m_p^* \in \mathcal{R}'$. This is by definition of $\mathcal{R}'$ as $r' < m_p^* \in \mathcal{R} \cap \mathtt{p}^{-1}(\mathcal{P}(p))$ implies that $\mathtt{t}_{\mathcal{P},\mathcal{R}}(r') < t$ by definition of m_p^*. From $m_p^* \in \mathcal{R}'$, it follows that $m_p^* \leq m_{\mathcal{R}',\mathcal{P}(p)}$. We now show that $m_p^* \geq m_{\mathcal{R}',\mathcal{P}(p)}$. Indeed, if not, then as $m_p^* < m_{\mathcal{R}',\mathcal{P}(p)} \in \mathcal{R}'$, we have from the definition of $\mathcal{R}'$ that $\mathtt{t}_{\mathcal{P},\mathcal{R}}(m_p^*) < t$, a contradiction.

For the "moreover" part, the $\impliedby$ direction is trivial. For the $\implies$ direction, use the definition of $\mathcal{R}'$ and that $m_{\mathcal{R}',\mathcal{P}(p)} \in \mathcal{R}'$. □

Claim 5.37. *For all $r \in \{0\} \cup \mathcal{R}'$ and all $C \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}'}(r)$, we have that $|C| < t \implies C \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}}(r)$.*

Proof. Proof by induction on r. The base case $r = 0$ is trivial. We prove the claim for $r > 0$ by assuming it holds for smaller values of r. By Definition 5.3, if $C \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}'}(r)$, then $\mathtt{p}(r) = \mathcal{P}(p)$ for some $p > 1$ and there exists a value $b' \in [B_{\mathcal{P},\mathcal{R}',p-1,r}]$ and a set $C' \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}'}(\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b'))$ satisfying

$$C' \cup \{(\mathcal{P}(p), r') \mid \mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b'+1) \leq r' < r\} \subseteq C.$$

As $\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b') < r$, we can apply the induction hypothesis on $\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b')$ to conclude that $C' \in \mathtt{Crash}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b'))$. As $|C'| \leq |C| < t$, we have that $\mathtt{t}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b')) < t$ implying by contrapositive of Lemma 5.25 that $\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b') < m_{\mathcal{R}',\mathcal{P}(p-1)}$.

Next, note from Eq. (6) that there is a value $b \in [B_{\mathcal{P},\mathcal{R},p-1,r}]$ such that $\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b) = \mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b')$. From $\mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b') < m_{\mathcal{R}',\mathcal{P}(p-1)}$ and the "moreover" part of Claim 5.36,

we get that $\mathcal{B}_{\mathcal{P},\mathcal{R},p-1,r}(b+1) = \mathcal{B}_{\mathcal{P},\mathcal{R}',p-1,r}(b'+1)$. By Definition 5.3, this means that $C \in \texttt{Crash}_{\mathcal{P},\mathcal{R}}(r)$, as desired.

□

We now show that $t \leq \texttt{Str}_{\mathcal{P},\mathcal{R}'}$ by showing that $t \leq \texttt{t}_{\mathcal{P},\mathcal{R}'}\left(m_{\mathcal{R}',\mathcal{P}(p)}\right)$ for all $p \in [P]$. We do this by contradiction. Let p be a counterexample. As $\texttt{t}_{\mathcal{P},\mathcal{R}'}\left(m_{\mathcal{R}',\mathcal{P}(p)}\right) < t$, there exists a set $C \in \texttt{Crash}_{\mathcal{P},\mathcal{R}'}\left(m_{\mathcal{R}',\mathcal{P}(p)}\right)$ such that $|C| < t$. By Claim 5.37, we have that $C \in \texttt{Crash}_{\mathcal{P},\mathcal{R}}\left(m_{\mathcal{R}',\mathcal{P}(p)}\right)$ implying that $\texttt{t}_{\mathcal{P},\mathcal{R}}\left(m_{\mathcal{R}',\mathcal{P}(p)}\right) < t$. This contradicts Claim 5.36.

□

Lemma 5.38. *For integers $t > 0$, nice pairs $(\mathcal{P},\mathcal{R})$ satisfying $|\mathcal{P}| > 1$ and $2 \cdot \frac{|\mathcal{R}|}{|\mathcal{P}|} < t \leq \texttt{Str}_{\mathcal{P},\mathcal{R}}$, we have that:*

$$T \geq |\mathcal{P}| \cdot \frac{\texttt{Str}_{\mathcal{P},\mathcal{R}}}{10t}.$$

Proof. Proof by induction on t. The base case $t = 1$ is trivial as $|\mathcal{R}| < |\mathcal{P}|$ implies that $\texttt{Str}_{\mathcal{P},\mathcal{R}} = 0$. We show the result for $t > 1$ by assuming it holds for $t-1$. Define the set:

$$\mathcal{R}' = \left\{ r \in \mathcal{R} \mid \forall r' < r \in \mathcal{R} \cap \texttt{p}^{-1}(\texttt{p}(r)) : \texttt{t}_{\mathcal{P},\mathcal{R}}(r') < \texttt{Str}_{\mathcal{P},\mathcal{R}} \cdot \frac{t-1}{t} \right\}.$$

As $\mathcal{R}' \subseteq \mathcal{R}$, we have from Definition 5.3 that $(\mathcal{P},\mathcal{R}')$ is nice. Also, we have from Lemma 5.35 that $\texttt{Str}_{\mathcal{P},\mathcal{R}} \cdot \frac{t-1}{t} \leq \texttt{Str}_{\mathcal{P},\mathcal{R}'}$. If it holds that $2 \cdot \frac{|\mathcal{R}'|}{|\mathcal{P}|} < t-1$, we get the lemma from our induction hypothesis, so we assume otherwise and deduce:

$$2 \cdot \frac{|\mathcal{R}'|}{|\mathcal{P}|} \geq t-1 > 2 \cdot \frac{|\mathcal{R}|}{|\mathcal{P}|} - 1 \implies |\mathcal{R}'| > |\mathcal{R}| - \frac{|\mathcal{P}|}{2}.$$

Next, conclude from $\texttt{Str}_{\mathcal{P},\mathcal{R}} > 0$ that $m_{\mathcal{R},i}$ is distinct for all $i \in \mathcal{P}$. Let $\mathcal{P}' \subseteq \mathcal{P}$ be the set of $i \in \mathcal{P}$ such that $m_{\mathcal{R},i} \in \mathcal{R}'$. As $|\mathcal{R}'| > |\mathcal{R}| - \frac{|\mathcal{P}|}{2}$, we must have $|\mathcal{P}'| > |\mathcal{P}|/2$. Denote by $P' = |\mathcal{P}'|$ and let $\mathcal{P}'(1) < \mathcal{P}'(2) < \cdots < \mathcal{P}'(P')$ be the elements of $\mathcal{P}'$. We claim that:

Claim 5.39. *For all $1 < p' \leq P'$, we have:*

$$\frac{\texttt{Str}_{\mathcal{P},\mathcal{R}}}{t} < m_{\mathcal{R},\mathcal{P}'(p')} - m_{\mathcal{R},\mathcal{P}'(p'-1)}.$$

Proof. As we have $\mathcal{P}' \subseteq \mathcal{P}$, we have $q' < q \in [P]$ such that $\mathcal{P}(q') = \mathcal{P}'(p'-1)$ and $\mathcal{P}(q) = \mathcal{P}'(p')$. Let $r = m_{\mathcal{R},\mathcal{P}'(p')}$ and $r' = m_{\mathcal{R},\mathcal{P}'(p'-1)}$ for convenience. We first show that $r' < r$. If not, then as $m_{\mathcal{R},i}$ is distinct for all $i \in \mathcal{P}$, we must have $r < r'$. From Item 1 of Lemma 5.28, we get that $\texttt{t}_{\mathcal{P},\mathcal{R}}(r) \leq \texttt{t}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R},q',r}(B_{\mathcal{P},\mathcal{R},q',r}))$. Now use the fact that $\mathcal{B}_{\mathcal{P},\mathcal{R},q',r}(B_{\mathcal{P},\mathcal{R},q',r}) \leq r < r'$ and $r' \in \mathcal{R}'$ to continue as $\texttt{t}_{\mathcal{P},\mathcal{R}}(r) < \texttt{Str}_{\mathcal{P},\mathcal{R}} \cdot \frac{t-1}{t}$, a contradiction.

Having shown that $r' < r$. Observe that this implies $B_{\mathcal{P},\mathcal{R},q',r} > 1$ and $\mathcal{B}_{\mathcal{P},\mathcal{R},q',r}(B_{\mathcal{P},\mathcal{R},q',r}) = r'$. From Item 2 of Lemma 5.28, we get that

$$\texttt{Str}_{\mathcal{P},\mathcal{R}} \leq \texttt{t}_{\mathcal{P},\mathcal{R}}(r) \leq \texttt{t}_{\mathcal{P},\mathcal{R}}(\mathcal{B}_{\mathcal{P},\mathcal{R},q',r}(B_{\mathcal{P},\mathcal{R},q',r} - 1)) + r - r'.$$

Now use $\mathcal{B}_{\mathcal{P},\mathcal{R},q',r}(B_{\mathcal{P},\mathcal{R},q',r}-1) < r'$ and $r' \in \mathcal{R}'$ to get $\texttt{Str}_{\mathcal{P},\mathcal{R}} < \texttt{Str}_{\mathcal{P},\mathcal{R}} \cdot \frac{t-1}{t} + r - r'$, as claimed.

□

As $P' = |\mathcal{P}'| > |\mathcal{P}|/2 \geq 1$, we have from Claim 5.39 that:

$$T \geq m_{\mathcal{R},\mathcal{P}'(P')} \geq (P'-1) \cdot \frac{\texttt{Str}_{\mathcal{P},\mathcal{R}}}{t} \geq |\mathcal{P}| \cdot \frac{\texttt{Str}_{\mathcal{P},\mathcal{R}}}{10t}.$$

□

5.3.3 Abundance of Small Crashing Sets

We now show the main result of Section 5.3 which is that there are plenty of disjoint and small crashing sets.

Lemma 5.40. *Let $k = \sqrt{n}$. There exists nice pairs $\{\mathcal{P}_l, \mathcal{R}_l\}_{l \in \{0\} \cup [k]}$ and disjoint sets $\{C_l\}_{l \in [k]}$ such that:*

1. *For all $l \in \{0\} \cup [k]$, we have $|\mathcal{P}_l| \geq n - l \cdot \frac{\log n}{100}$ and $n \in \mathcal{P}_l$.*
2. *For all $l \in \{0\} \cup [k]$, we have $\mathcal{R}_l = \{r \in [T] \mid \texttt{p}(r) \in \mathcal{P}_l\}$ and $T \in \mathcal{R}_l$.*
3. *For all $l \in [k]$, we have $|C_l| \leq \frac{\log n}{100}$ and $C_l \in \texttt{Crash}_{\mathcal{P}_{l-1},\mathcal{R}_{l-1}}(\max(\mathcal{R}_{l-1}))$.*

Proof. It is enough to show a weaker version of the lemma that does not the conditions $n \in \mathcal{P}_l$ and $T \in \mathcal{R}_l$ as these conditions can be obtained by applying Lemma 5.29 on the weaker version. We will need the following helper claim:

Claim 5.41. *Let $(\mathcal{P}, \mathcal{R})$ be a nice pair such that $|\mathcal{P}| > \frac{n}{2}$ and $\mathcal{R} = \{r \in [T] \mid \texttt{p}(r) \in \mathcal{P}\}$. There exists a set C such that $|C| < \frac{\log n}{100}$ and $C \in \texttt{Crash}_{\mathcal{P},\mathcal{R}}(\max(\mathcal{R}))$.*

Proof. Note that $\texttt{Str}_{\mathcal{P},\mathcal{R}} < \frac{\log n}{100}$. Indeed, if not, we have by the contrapositive of Lemma 5.38 with $t = \frac{\sqrt{\log n}}{1000}$ that $T \geq \frac{n}{2} \cdot \sqrt{\log n}$, a contradiction. By Definition 5.32, this means that there is $p \in [P]$ such that $\texttt{t}_{\mathcal{P},\mathcal{R}}\left(m_{\mathcal{R},\mathcal{P}(p)}\right) < \frac{\log n}{100}$. By Lemma 5.33 (the conditions in Lemma 5.33 are satisfied due to our assumption that $\texttt{p}(T+1-i) = n+1-i$ for all $i \in [n]$), we can continue as $\texttt{t}_{\mathcal{P},\mathcal{R}}(\max(\mathcal{R})) < \frac{\log n}{100}$. The lemma now follows from definition of $\texttt{t}(\cdot)$. □

We define the sets inductively. At stage j of the induction, for $j \in \{0\} \cup [k]$, we would have defined the sets $\{\mathcal{P}_l, \mathcal{R}_l\}_{l \in \{0\} \cup [j]}$ and $\{C_l\}_{l \in [j]}$ such that Items 1 to 3 are satisfied for all $l \leq j$. For the base case $j = 0$, define $\mathcal{P}_0 = [n]$ and $\mathcal{R}_0 = [T]$ and observe that these satisfy Items 1 to 3. For the inductive step, let $j > 0$ be such that $\mathcal{P}_{j-1}, \mathcal{R}_{j-1}$ have been defined. Define C_j to be the set promised by Claim 5.41. By Claim 5.41, we have $|C_j| \leq \frac{\log n}{100}$ and $C_j \in \texttt{Crash}_{\mathcal{P}_{j-1},\mathcal{R}_{j-1}}(\max(\mathcal{R}_{j-1}))$ and Item 3 is satisfied. Next, define

$$\mathcal{P}_j = \mathcal{P}_{j-1} \setminus \{i \in \mathcal{P}_{j-1} \mid \exists j \in [T] : (i,j) \in C_j\}.$$

$$\mathcal{R}_j = \{r \in [T] \mid \mathtt{p}(r) \in \mathcal{P}_j\}.$$

Item 2 is clearly satisfied. For Item 1, note that $|\mathcal{P}_j| \geq |\mathcal{P}_{j-1}| - |C_j| \geq n - j \cdot \frac{\log n}{100}$. Finally, the sets $\{C_l\}_{l\in[k]}$ are disjoint as for all $l \in [k]$, we have that $C_l \subseteq \mathcal{P}_{l-1} \times [T]$ by Definition 5.3. By our definition of $\mathcal{P}_j$, this gives $C_l \subseteq (\mathcal{P}_{l-1} \setminus \mathcal{P}_l) \times [T]$ and it follows from $\mathcal{P}_l \subseteq \mathcal{P}_{l-1}$ that the sets $\{C_l\}_{l\in[k]}$ are disjoint. □

5.4 Finishing the Proof

We now combine Lemmas 5.24 and 5.40 to finish the proof of Theorem 5.1.

Proof of Theorem 5.1. Let $k = \sqrt{n}$ and $\{\mathcal{P}_l, \mathcal{R}_l\}_{l\in\{0\}\cup[k]}$ and $\{C_l\}_{l\in[k]}$ be those promised by Lemma 5.40. As $|C_l| \leq \frac{\log n}{100}$, we have that $\Pr(\mathsf{N} \in \mathcal{N}(C_l)) \geq \frac{1}{n^{0.01}}$. As the sets $\{C_l\}_{l\in[k]}$ are disjoint we have $\Pr(\mathcal{E}) \leq 2^{-n^{0.25}}$, where:

$$\mathcal{E} = \{N \mid \forall l \in [k] : N \notin \mathcal{N}(C_l)\}.$$

We derive:

$$\begin{aligned}
\Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F}))) &\leq \Pr(\mathcal{E}) + \Pr\big(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid \overline{\mathcal{E}}\big) && \text{(Union bound)}\\
&\leq \Pr(\mathcal{E}) + \max_{N\notin\mathcal{E}} \Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid N)\\
&\leq 2^{-n^{0.25}} + \max_{N\notin\mathcal{E}} \Pr(\Pi(\mathsf{F}) = \mathtt{LSB}(\mathtt{PC}(\mathsf{F})) \mid N) && (\text{As } \Pr(\mathcal{E}) \leq 2^{-n^{0.25}})\\
&\leq \frac{1}{2} + 2^{-n^{0.25}} + \frac{1}{m^{0.1}} && \text{(Lemma 5.24)}\\
&\leq \frac{1}{2} + \frac{1}{n}.
\end{aligned}$$

□

References

[ABE+16] 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. 4

[AGL20] Yagel Ashkenazi, Ran Gelles, and Amir Leshem. Brief announcement: Noisy beeping networks. In *Symposium on Principles of Distributed Computing (PODC)*, pages 458–460, 2020. 4

[AGS16] Shweta Agrawal, Ran Gelles, and Amit Sahai. Adaptive protocols for interactive communication. In *Information Theory (ISIT)*, pages 595–599. IEEE, 2016. 4

[BEGH16] 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. 4

[CHHZ17] 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. 4

[CK85] Imrich Chlamtac and Shay Kutten. On broadcasting in radio networks-problem analysis and protocol design. *IEEE Trans. Communications*, 33(12):1240–1246, 1985. 3

[EKPS21] 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. 3, 4

[EKS18] 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. 3, 4, 6, 32

[EKS19] Klim Efremenko, Gillat Kol, and Raghuvansh Saxena. Radio network coding requires logarithmic overhead. In *Foundations of Computer Science (FOCS)*, pages 348–369, 2019. 4

[EKS20a] Klim Efremenko, Gillat Kol, and Raghuvansh R. Saxena. Interactive error resilience beyond 2/7. In *Symposium on Theory of Computing (STOC)*, pages 565–578, 2020. 4

[EKS20b] 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. 4

[EKS21] Klim Efremenko, Gillat Kol, and Raghuvansh R Saxena. Optimal error resilience of adaptive message exchange. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1235–1247, 2021. 4

[FK00] Uriel Feige and Joe Kilian. Finding OR in a noisy broadcast network. *Information Processing Letters*, 73(1-2):69–75, 2000. 3, 4

[Gal88] Robert G. Gallager. Finding parity in a simple broadcast network. *IEEE Transactions on Information Theory*, 34(2):176–180, 1988. 1, 3, 5, 6, 9

[Gam87] 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. 1, 3

[GHM18] Ofer Grossman, Bernhard Haeupler, and Sidhanth Mohanty. Algorithms for noisy broadcast with erasures. In *Colloquium on Automata, Languages, and Programming (ICALP)*, volume 107 of *LIPIcs*, pages 153:1–153:12, 2018. 4

[GHS14] Mohsen Ghaffari, Bernhard Haeupler, and Madhu Sudan. Optimal error rates for interactive coding i: Adaptivity and other settings. In *Symposium on Theory of computing (STOC)*, pages 794–803, 2014. 4

[GKS08] Navin Goyal, Guy Kindler, and Michael Saks. Lower bounds for the noisy broadcast problem. *SIAM Journal on Computing*, 37(6):1806–1841, 2008. 1, 2, 3, 4

[Hae14] Bernhard Haeupler. Interactive channel capacity revisited. In *Foundations of Computer Science (FOCS)*, pages 226–235. IEEE, 2014. 4

[KM05] Eyal Kushilevitz and Yishay Mansour. Computation in noisy radio networks. *SIAM J. Discrete Math.*, 19(1):96–108, 2005. 3, 4

[New04] Ilan Newman. Computing in fault tolerance broadcast networks. In *Computational Complexity Conference (CCC)*, pages 113–122, 2004. 2, 3, 4

[RS94] Sridhar Rajagopalan and Leonard J. Schulman. A coding theorem for distributed computation. In *Symposium on the Theory of Computing (STOC)*, pages 790–799, 1994. 4

[Sch92] Leonard J Schulman. Communication on noisy channels: A coding theorem for computation. In *Foundations of Computer Science (FOCS)*, pages 724–733. IEEE, 1992. 4, 9

[Yao97] Andrew Chi-Chih Yao. On the complexity of communication under noise. *invited talk in the 5th ISTCS Conference*, 1997. 1, 2, 3

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