Constant-Rate Coding for Multiparty Interactive Communication Is Impossible

MARK BRAVERMAN, Princeton University
KLIM EFREMENKO, Ben-Gurion University
RAN GELLES, Bar-Ilan University
BERNHARD HAEUPLER, Carnegie Mellon University

We study coding schemes for multiparty interactive communication over synchronous networks that suffer from stochastic noise, where each bit is independently flipped with probability ε. We analyze the minimal overhead that must be added by the coding scheme to succeed in performing the computation despite the noise.

Our main result is a lower bound on the communication of any noise-resilient protocol over a synchronous star network with n parties (where all parties communicate in every round). Specifically, we show a task that can be solved by communicating T bits over the noise-free network, but for which any protocol with success probability of $1 - o(1)$ must communicate at least $\Omega(T \frac{\log n}{\log \log n})$ bits when the channels are noisy. By a 1994 result of Rajagopalan and Schulman, the slowdown we prove is the highest one can obtain on any topology, up to a $\log \log n$ factor.

We complete our lower bound with a matching coding scheme that achieves the same overhead; thus, the capacity of (synchronous) star networks is $\Theta(\log \log n / \log n)$. Our bounds prove that, despite several previous coding schemes with rate $\Omega(1)$ for certain topologies, no coding scheme with constant rate $\Omega(1)$ exists for arbitrary n-party noisy networks.

CCS Concepts: • **Mathematics of computing** → **Coding theory**; • **Theory of computation** → **Communication complexity**; *Interactive computation*;

Additional Key Words and Phrases: Multiparty interactive communication, coding theory, star network, communication complexity, lower bounds, random noise

ACM Reference format:
Mark Braverman, Klim Efremenko, Ran Gelles, and Bernhard Haeupler. 2017. Constant-Rate Coding for Multiparty Interactive Communication Is Impossible. *J. ACM* 65, 1, Article 4 (December 2017), 41 pages.
https://doi.org/10.1145/3050218

This article is supported in part by an NSF CAREER award (CCF-1149888), NSF CCF-1525342 and DMS-1128155, a Packard Fellowship in Science and Engineering, the Simons Collaboration on Algorithms and Geometry, the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement No. 257575, ISF Grant No. 1078/17, NSF Grants No. CCF-1527110 and No. CCF-1618280.
Authors' addresses: M. Braverman, Department of Computer Science, Princeton University, 35 Olden Street, Princeton, NJ 08540, USA; email: mbraverm@cs.princeton.edu; K. Efremenko, Department of Computer Science, Ben-Gurion University of the Negev, POB 653, Beer-Sheva 84105, ISRAEL; email: klimefrem@gmail.com; R. Gelles (corresponding author), Faculty of Engineering, Bar-Ilan University, Ramat-Gan 52900, ISRAEL; email: ran.gelles@biu.ac.il; B. Haeupler, Computer Science Department, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA 15213, USA; email: haeupler@cs.cmu.edu.

https://doi.org/10.1145/3050218

1 INTRODUCTION

Assume a network of n remote parties who perform a distributed computation of some function of their private inputs, while their communication may suffer from stochastic noise. The task of *coding for interactive communication* asks for coding schemes that allow the parties to correctly compute the needed function while limiting the overhead incurred by the coding. For the two-party case, $n = 2$, Schulman, in a pioneering line of results [37–39], showed how to convert any protocol that takes T rounds when the communication is noiseless into a resilient protocol that succeeds with high probability and takes $O(T)$ rounds when the communication channel is a *binary symmetric channel* (BSC),[1] that is, when the communication may suffer from random noise.

For the general case of n parties, Rajagopalan and Schulman [36] showed a coding scheme that succeeds with high probability and takes $O(T \log n)$ rounds in the worst case. Here, a "round" means simultaneous communication of a single bit over each one of the channels. More precisely, the communication of the coding scheme in Reference [36] depends on the specific way the parties are connected to each other. Specifically, the scheme takes $O(T \log(d + 1))$ rounds, where d is the maximal number of neighbors a party may have. Thus, for certain topologies like a line or a cycle, the slowdown is constant $O(1)$, however, in the worst case, i.e., when the topology is a complete graph, the scheme has a slowdown of $O(\log n)$.

The work of Alon et al. [2] shows how to improve the $O(\log n)$ slowdown when the network's topology is a complete graph. Specifically, they provide a coding scheme with high probability of success and slowdown of $O(1)$ for a rich family of "highly connected" topologies, including the complete graph. Therefore, a constant-slowdown coding scheme is achievable either when the degree is constant [36], or when the connectivity is high [2], i.e., when many disjoint paths connect every two parties.

The main outstanding open question left by these works is whether a constant-slowdown coding scheme can be obtained for *all* topologies. We answer this question in the negative and show a lower bound on the slowdown of any coding scheme with high probability of success, over a star network:

Theorem 1.1 (Main, Lower Bound). *Assume n parties connected as a star, and let $\varepsilon < 1/2$ be given. There exists an n-party protocol χ that takes T rounds assuming noiseless channels, such that any coding scheme that simulates χ with probability above $1/5$ when each channel is a BSC_ε, takes $\Omega(T \frac{\log n}{\log \log n})$ rounds.*

By making "simulating χ" (i.e., computing the transcript generated by χ, assuming noiseless channels) the interactive task to be performed, Theorem 1.1 implies the $\Omega(\frac{\log n}{\log \log n})$ slowdown in interactive coding. Note that the coding of Reference [36] implies a slowdown of $O(\log n)$ in this case, hence, our result is tight up to an $O(\log \log n)$ term.

We complement our lower bound with a matching upper bound and show that coding with a slowdown of $O(\frac{\log n}{\log \log n})$ is achievable and therefore tight for interactive coding over a star topology.

Theorem 1.2 (Upper Bound). *Assume n parties connected as a star, and let $\varepsilon < 1/2$. For any n-party protocol χ that takes T rounds assuming noiseless channels, there exists a coding scheme that simulates χ assuming each channel is a BSC_ε, takes $N = O(T \frac{\log n}{\log \log n})$ rounds, and succeeds with probability $1 - 2^{-\Omega(N)}$.*

[1] The BSC channel, parametrized by a probability ε, flips each bit independently with probability ε, and leaves the bit unflipped with probability $1 - \varepsilon$.

The upper bound follows quite straightforwardly from an observation by Alon et al. [2], showing that as long as one round of the noiseless χ can be simulated with high probability, then the entire protocol χ can be simulated with high probability by employing the techniques of Reference [36]. Over a star, it is quite simple to simulate $\log\log n$ rounds of an arbitrary noiseless χ using only $O(\log n)$ noisy rounds, with high probability. Thus, we can apply the technique of References [2, 36] on segments of $\log\log n$ rounds of χ, and achieve the stated coding scheme. We prove Theorem 1.2 in Section 4.

We devote Section 5 to prove the more involved lower bound of Theorem 1.1. Below we give a rather intuitive overview of our lower bound result (focusing on the binary case) and the techniques we use.

1.1 Lower Bound: Overview and Techniques

To achieve our lower bound of $\Omega(\frac{\log n}{\log\log n})$ on the slowdown, we consider protocols for the *pointer jumping task* of depth T, between n parties (also called *clients*) and the center of the star (also called the *server*). In the pointer jumping task, each client gets as an input a binary tree of depth T, where each edge is labeled with a single bit. The server's input is a 2^n-ary tree of depth T where each edge is labeled with an n-bit string. Solving the pointer jumping task is equivalent to performing the following protocol: all parties begin from the root of their trees. At each round, simultaneously for $1 \leq i \leq n$, the i-th client receives a bit b_i from the center and descends in his tree to the b_i-th child of its current node. The client then sends back to the server the label of the edge through which it traversed. The server receives, at each round, the string $B = b_1 \cdots b_n$ from the clients and descends to the B-th child of its current node. If the edge going to that node is labeled with the n-bit string $b'_1 \cdots b'_n$, then the server sends b'_i to the i-th client. The above process repeats until the parties reach the T-th level in their respective tree. At the end, each party outputs the leaf it has reached (equivalently, it outputs the "path" it traversed). Note that the T-level pointer jumping task can be solved using $2T$ rounds of alternating noiseless communication. By *alternating* we mean here that the server speaks on, say, odd rounds, while the clients speak on even rounds. Also note that the pointer jumping task is complete for interactive communication, i.e., any interactive protocol for $n + 1$ parties connected as a star can be represented as a specific input-instance of the above pointer jumping task. See Section 3 for further details about the multiparty pointer jumping task.

Next, we assume the channels are noisy. In fact, we can weaken the noise model and assume that the noise erases bits rather than flipping them, that is, we consider the binary erasure channel, BEC_ε; see Definition 2.1. Note that since the considered noise model is *weaker*, our lower bound becomes *stronger*.

Consider any protocol that solves the pointer jumping task of depth T, assuming the channels are $\mathsf{BEC}_{1/3}$. We divide the protocol into segments of length $0.1\log n$ rounds each and show that at each such segment the protocol "advances" by at most $O(\log\log n)$ levels in the underlying pointer jumping task, in expectation. Very roughly, the reason for this slow progress follows from the observation that during each segment of $0.1\log n$ rounds, with high probability there exists a set of $\sqrt{n}$ clients whose communication was completely erased. It follows that the server is missing knowledge on $\sqrt{n}$ parties and thus cannot infer its next node with high probability. On average, the server sends a very small amount of information on the labels descending from its current node that belong to the "correct" path. As a result, the clients practically receive no meaningful information on the next level(s) of the server. This, in turn, limits the amount of information they can send on *their* "correct" paths to $O(\log\log n)$ bits in expectation, thus limiting the maximal advancement in the underlying pointer jumping task. For instance, if some client who does not know the correct path in his input pointer-jumping tree communicates to the server all the labels descending from its current node, say in a breadth-first manner, the information sent during $0.1 \log n$ rounds can contain, at most, $O(\log \log n)$ levels of this client's *correct* path.

Not surprisingly, the technical execution of the above strategy requires tools for careful and accurate bookkeeping of the information the parties have learned at any given time of the (noisy) execution. The basic definition of information a party has about a random variable X sampled from a space Ω_X we employ is $I(X) \overset{\text{def}}{=} \log |\Omega_X| - H(X)$, where $H(X)$ is Shannon's entropy of X given the party's current knowledge. Note that if *a priori* X is uniformly distributed, $I(X)$ is exactly the mutual information between what the party knows and X. However, this information notion behaves more nicely under conditioning (i.e., when changing what the party knows about X as the protocol progresses), and seems generally easier to work with. Indeed, this notion was previously used when bounding the information in pointer jumping tasks [29, 33, 35].

A central notion in our analysis is the *cutoff* round of the protocol, which relates to the deepest level of the underlying pointer jumping task that the parties can infer from the communication they have received so far. Very roughly, if the cutoff is k, then parties have small information on labels below level k in the underlying tree of the party (or parties) connected to them. More precisely, for any (partial) transcript π the parties observe, we define $\text{cutoff}(\pi)$ to be the minimal round $1 \leq k \leq T$ for which the parties have a small amount of information about labels in the underlying pointer jumping task that lies in the subtree rooted at the end of the correct path of depth k, conditioned on the transcript π and on the correct path up to level k (see Definition 5.2 for the exact formulation).

The core of our analysis shows that, given a certain cutoff, $\text{cutoff}(\pi) = \ell$, and assuming the parties communicate the next $0.1 \log n$ rounds of the protocol (denote the observed transcript in this new part as Π^{new}), then in expectation over the possible inputs, noise, and randomness of the protocol, the cutoff does not increase by more than $O(\log \log n)$; that is,

$$\mathbb{E}[\text{cutoff}(\pi, \Pi^{new}) \mid \text{cutoff}(\pi) = \ell] \leq \ell + O(\log \log n).$$

This implies that, unless the protocol runs for $\Omega(T \frac{\log n}{\log \log n})$ rounds, then the cutoff at the end of the protocol is substantially smaller than T, with high probability. Using Fano's inequality, this, in turn, implies that the protocol cannot output the correct path (beyond the cutoff round) with high probability.

Bounding the information revealed by the parties at each step is the deepest technical contribution of this paper, and is done in methods which are close in spirit to a technique by Kol and Raz [29] for obtaining lower bounds in the two-party case.[2] We bound separately the information that the server reveals and the information the clients reveal in each segment of $0.1 \log n$ rounds (conditioned on a given cutoff level, i.e., on the transcript of the protocol so far and on the correct path up to the cutoff level).

Very informally, we show that the information revealed during a single chunk on labels below a continuation of the correct path (i.e., the information captured by the "new" cutoff), can be bounded by the product of (i) the probability to guess the continuation of the correct path (between the current and the new cutoff levels) and (ii) the information that the transcript so far contains on all the labels (either on the correct path or not) that lie below the new cutoff level. Indeed, if a party wants to give information about the labels of its correct path, but that party doesn't know the correct path, it can't do much more than guess the path and send information about that guess; alternatively, it can give information on labels in all possible paths, where the amount of information of each label corresponds to the probability of this label to be part of the correct path.

[2] In fact, it is an interesting question whether our techniques can be used to simplify the analysis in Reference [29].

We bound each one of the above terms separately. For the first part (i), we bound the guessing probability of a continuation of the correct path as a function of the information the observed transcript contains on the labels below the current cutoff *in the tree of the other parties*. For instance, guessing the correct path in the server's tree depends on the amount of information the transcript gives on labels along the correct path in the clients' trees, at the same levels (because these labels exactly determine the path the server should take in his tree). The definition of the cutoff and the fact that these levels lie below the cutoff level, give a bound the amount of information we have on these labels, which can be translated to a bound on the probability of guessing the corresponding path. Fano's inequality is not strong enough for our needs (i.e., sub-exponential guessing probability from sub-exponentially small information), and we devise a tighter bound via a careful analysis of the positive and negative parts of the Kullback–Leibler divergence; see Lemma 2.15. This (entropy vs. min-entropy) relation may be of independent interest.

To bound the second part (ii), we observe that the information on labels below the current cutoff is bounded *in expectation* using the definition of the cutoff, up to possibly additional $0.1n \log n$ bits that were communicated during the new segment of $0.1 \log n$ rounds.

The fact that the bound of part (ii) works only in expectation is a major hurdle, because it prevents us from bounding the above product directly (these two multiplicands are dependent!). We detour around this issue by narrowing down the probability space by conditioning on additional information that makes the two multiplicands independent. However, conditioning on additional information potentially increases the information we wish to bound, then it is essential to carefully limit the amount of additional information we condition on, so that the bound remains meaningful. Giving more details (yet still very intuitively speaking), we condition on all the labels that lie between the old and new cutoff levels, of either the server's input *or* the clients' input, according to the specific information we are currently bounding. This conditioning takes out the dependency caused by the interaction (since the labels of one side are fixed up to some given level) and makes the labels below the new cutoff independent of labels above it; specifically, the correct path between the current and the new cutoff (which is involved in the first multiplicand) is conditionally independent of the labels below the new cutoff (which are involved in the second one). This independence allows us to bound the expectation of the above product by bounding each term separately as described above.

1.2 Related Work

As mentioned above, coding for interactive communication in the presence of random noise was initiated by Schulman for the two-party case [37–39]. The coding scheme of Schulman [38, 39] achieves slowdown of $O(1)$ and exponentially high success probability; however, it is not computationally efficient and can take exponential time in the worst case. Gelles, Moitra, and Sahai [20, 21] showed how to obtain an efficient coding scheme while maintaining a constant slowdown and exponentially high success probability. Braverman [4] gave another efficient coding scheme, yet with a slightly reduced success probability. Other related work in the two-party setting considers the case of adversarial noise rather than random noise, in various settings [1, 3, 5, 7, 8, 10, 12, 14, 18, 22–24]; see [16] for a survey.

In the two-party setting, the minimal possible slowdown over a BSC_ε as a function of the noise parameter ε, was initially considered by Kol and Raz [29], who showed a lower bound of $1 + \Omega(\sqrt{\varepsilon \log 1/\varepsilon})$ on the slowdown. Later, Haeupler [26] showed that the order in which the parties are speaking affects the slowdown, and if the parties are assumed to be alternating, a slowdown of $1 + O(\sqrt{\varepsilon})$ is achievable. When the noise is adversarial rather than random, the slowdown increases to $1 + O(\sqrt{\varepsilon \log \log 1/\varepsilon})$ [26]. The slowdown in other types of channels, such as the binary erasure channel BEC_ε or channels with noiseless feedback, was considered by Gelles and Haeupler [17], who showed efficient coding schemes with an optimal slowdown of $1 + \Theta(\varepsilon \log 1/\varepsilon)$ over these channels.

As for the multiparty case, the work of Rajagopalan and Schulman [36] was the first to give a coding scheme for the case of random noise over arbitrary topology, with a slowdown of $O(\log(d + 1))$ for d, the maximal degree of the connectivity graph. As in the two-party case, that scheme is not efficient, but can be made efficient using References [20, 21]. Alon, Braverman, Efremenko, Gelles, and Haeupler [2] considered coding schemes over d-regular graphs with mixing time[3] m, and obtained a slowdown of $O(m^3 \log m)$. This implies a coding scheme with a constant slowdown $O(1)$ whenever the mixing time is constant, $m = O(1)$, e.g., over complete graphs.

For the case of adversarial noise in the multiparty setting, Jain, Kalai, and Lewko [28] showed an asynchronous coding scheme for star topologies with slowdown $O(1)$ for up to $O(1/n)$-fraction of noise. A communication-balanced version of that scheme was given by Lewko and Vitercik [31]. Hoza and Schulman [27] showed a coding scheme in the synchronous model that works for any topology, tolerates $O(1/n)$-fraction of noise, and demonstrates a slowdown of $O(\frac{m}{n} \log n)$ where m here is the number of edges in the given connectivity graph.

Finally, we mention the work of Gallager [15]. This article assumes a different setting than the above works, namely, the case where parties are all connected via a noisy broadcast channel (the noisy blackboard model [13]). Gallager showed that a slowdown of $O(\log \log n)$ is achievable for the task where each party begins with a bit and needs to output the input bits of all other parties. Goyal, Kindler, and Saks [25] showed that this slowdown is tight by providing a matching slowdown of $\Omega(\log \log n)$ for the same task in the noisy broadcast model. It is not clear whether there is a direct connection between results in these two models—there does not seem to be a way to translate results in either direction.

In a subsequent work, Gelles and Kalai [19] revisit the question of slowdown in the multiparty case, and relax the assumption taken in References [2, 36] as well as in this article—that at each round all the parties must send a bit to each one of their neighbors. Using the machinery we develop in this article, Gelles and Kalai show a communication slowdown of $\Omega(\log n)$ for interactive coding over *cycle graphs* if parties are not required to communicate at each and every round. This is somewhat surprising in light of the *constant* upper bound on the slowdown implied by Reference [36] for graphs with a constant degree, such as the cycle whose degree is $d = 2$.

2 PRELIMINARIES

For $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, 2, \ldots, n\}$. The log() function is taken to base 2. We denote the natural logarithm by ln(). If X is a random variable with distribution P_X, we write $\mathbb{E}_{x\sim X}[f(x)]$ (or simply $\mathbb{E}_x[f(x)]$) to denote the expectation over the distribution of X, namely, $\mathbb{E}_{x\sim X}[f(x)] \overset{\text{def}}{=} \sum_x P_X(x) f(x)$. For an event $\mathcal{E}$, we let $\mathbb{E}_{x\sim X|\mathcal{E}}[f(x)] \overset{\text{def}}{=} \sum_x P_{X|\mathcal{E}}(x) f(x)$ be the expectation over the conditional distribution $P_{X|\mathcal{E}}$.

2.1 Coding Over Noisy Networks

Given an undirected graph $G = (V, E)$ we assume a network with $|V|$ parties, where $u, v \in V$ share a communication channel if $(u, v) \in E$. In the case of a noisy network, each such link is assumed to be a BSC_ε or a BEC_ε.

[3] Intuitively speaking, the mixing time of a graph is the minimal number of steps a random walk needs to end up at every node with approximately equal probability.

Definition 2.1 (Channels [11]). For $\varepsilon \in [0,1]$ we define the binary symmetric channel $\mathrm{BSC}_\varepsilon : \{0,1\} \to \{0,1\}$ in which the input bit is flipped with probability ε, and remains the same with probability $1-\varepsilon$. The binary erasure channel $\mathrm{BEC}_\varepsilon : \{0,1\} \to \{0,1,\perp\}$ turns each input bit into an erasure mark $\perp$ with probability ε, or otherwise keeps the bit intact. When a channel is accessed multiple times, each instance is independent.

A *round* of communication in the network means the simultaneous transmission of $2|E|$ messages: for any $(u,v) \in E$, u sends a bit to v and receives a bit from v. A protocol for an n-party function $f(x_1,\ldots,x_n) = (y_1,\ldots,y_n)$ is a distributed algorithm between n parties $\{p_1,\ldots,p_n\}$, where each p_i begins the protocol with an input x_i, and after N rounds of communication outputs y_i. The communication complexity of a protocol, $\mathrm{CC}()$, is the number of bits sent throughout the protocol. Note that given any network G, the round complexity of a protocol and its communication complexity differ by a factor of $2|E|$.

Assume χ is a protocol over a *noiseless* network G. We say that a protocol χ' simulates χ over a channel C with rate R if, when χ' is run with inputs $(x_1,\ldots,x_n)$ over the network G where each communication channel is C, each party outputs with high probability its respective transcript in the execution of χ on the inputs $(x_1,\ldots,x_n)$ and it holds that $\mathrm{CC}(\chi)/\mathrm{CC}(\chi') = R$. Note that given a transcript of χ, the parties can compute the output values $\chi(x_1,\ldots,x_n)$. We also use the term *slowdown* to denote the inverse of the rate, R^{-1}, that is, the (multiplicative) increase in the communication due to the coding.

2.2 Information, Entropy, and Min-Entropy

Throughout, we will use U_Ω to denote a random variable uniformly distributed over the finite and discrete domain Ω. In particular, U_n denotes a random variable uniformly distributed over $\{0,1\}^n$.

Definition 2.2 (Information). Let X be a random variable over a finite discrete domain Ω. The *information* of X is given by

$$I(X) \overset{\text{def}}{=} \log|\Omega| - H(X),$$

where $H(X)$ is the Shannon entropy of X, $H(X) = \sum_{x\in\Omega} \Pr(X=x)\log(1/\Pr(X=x))$.

Given a random variable Y, the *conditional information* of X given Y is

$$\begin{aligned} I(X \mid Y) &\overset{\text{def}}{=} \log|\Omega| - H(X \mid Y) \\ &= \mathbb{E}_y I(X \mid Y = y). \end{aligned}$$

Also note that $I(X) = D(X\|U_\Omega)$ where $D(\|)$ is the Kullback-Leibler divergence (Definition 2.7).

Lemma 2.3 (Superadditivity of Information). *Let $X_1,\ldots,X_n$ be n random variables. Then,*

$$\sum_{i=1}^{n} I(X_i) \le I(X_1,\ldots,X_n).$$

The equality is satisfied when $X_1,\ldots,X_n$ are mutually independent.

Proof. Using the subadditivity of the entropy function [11], we get

$$\sum_{i=1}^{n} I(X_i) = \sum_i (\log|\Omega_i| - H(X_i)) \le \log\left(\prod_i |\Omega_i|\right) - H(X_1,\ldots,X_n) = I(X_1,\ldots,X_n). \qquad \square$$

Lemma 2.4. *Let X, Y be random variables over the finite discrete domains Ω_X and Ω_Y, respectively. Then,*

(1) $I(X \mid Y) = I(X) + I(X;Y)$
(2) $I(X \mid Y) \leq I(X) + \log|\Omega_Y|$
(3) $I(X \mid Y) \leq I(X,Y)$,

where $I(X;Y) = H(X) + H(Y) - H(X,Y)$ *is the mutual information between* X *and* Y *(not to be confused with* $I(X,Y) = \log|\Omega_X| + \log|\Omega_Y| - H(X,Y)$*).*

Proof. We prove the three claims by order,

(1)
$$\begin{aligned} I(X \mid Y) &= \log|\Omega_X| - H(X \mid Y) \\ &= \log|\Omega_X| - H(X) + H(Y) - H(Y \mid X) \\ &= I(X) + I(X;Y). \end{aligned}$$

(2) Follows from (1) and the fact that $I(X;Y) \leq \log|\Omega_Y|$.

(3)
$$\begin{aligned} I(X,Y) &= \log|\Omega_X| + \log|\Omega_Y| - H(X,Y) \\ &\geq \log|\Omega_X| + H(Y) - (H(Y) + H(X \mid Y)) \\ &= I(X \mid Y). \end{aligned}$$ □

Definition 2.5 (Min-entropy). Let X be a random variable over a discrete domain Ω. The min-entropy of X is given by

$$H_\infty(X) = \log(1/p_{\max}(X)).$$

$p_{\max}(X)$ is the probability of the most probable value of X, i.e., $p_{\max}(X) \overset{\text{def}}{=} \max_{x\in\Omega} \Pr(X = x)$. At times, $p_{\max}$ is called the *guessing probability* of X.

We relate information (or, entropy) with the guessing probability (or min-entropy) via the next Lemma, which is a special case of Fano's inequality (see, e.g., Reference [11]).

Lemma 2.6. *Let* X *be a random variable over a discrete finite domain* Ω. *It holds that*

$$I(X) \geq p_{\max}(X)\log(|\Omega|) - h(p_{\max}(X)),$$

where $h(x) = -x\log x - (1-x)\log(1-x)$ *is the binary entropy.*

Proof. The lemma is an immediate corollary of the following version of Fano's inequality,

$$H(X) \leq \log|\Omega|(1 - 2^{-H_\infty(X)}) + h(2^{-H_\infty(X)}). \tag{1}$$

Let us prove Equation (1). Assume without loss of generality that $\Omega = \{1, \ldots, n\}$. Let $p_i = \Pr(X = i)$, and again assume without loss of generality that for any $i < j$, it holds that $p_i \geq p_j$. Thus, $p_{\max}(X) = p_1$. If $p_1 = 1$, the claim is trivial. Otherwise,

$$H(X) = p_1 \log\frac{1}{p_1} + \sum_{i=2}^{n} p_i \log\frac{1}{p_i}. \tag{2}$$

Define Y to have the same distribution of X conditioned on $X \neq 1$, i.e., $\Pr(Y = 1) = 0$ and $\Pr(Y = i) = p_i/(1 - p_1)$ for $i \in \{2, \ldots, n\}$. Note that

$$H(Y) = \sum_{i=2}^{n} \frac{p_i}{1-p_1}\log\frac{1-p_1}{p_i} = \left(\sum_{i=2}^{n} \frac{p_i}{1-p_1}\log\frac{1}{p_i}\right) - \log\frac{1}{1-p_1}.$$

Going back to Equation (2), we have

$$\begin{aligned} H(X) &= p_1\log\frac{1}{p_1} + (1-p_1)H(Y) + (1-p_1)\log\frac{1}{1-p_1} \\ &\leq h(p_1) + (1-p_1)\log|\Omega|, \end{aligned}$$

which holds because $H(Y) \leq \log(|\Omega| - 1) < \log|\Omega|$. Then Equation (1) and the lemma follow by substituting $p_1 = p_{\max}(X) = 2^{-H_\infty(X)}$. □

We note here that similar claims to the above lemmas hold when we additionally condition on some event $\mathcal{E}$; indeed, one can apply these lemmas on the random variable $(X \mid \mathcal{E})$.

Another key tool we use is the Kullback-Leibler divergence.

Definition 2.7 (KL Divergence [30]). Let X, Y be random variables over a discrete domain Ω. The KL divergence of X and Y is

$$\mathrm{D}(X\|Y) \overset{\text{def}}{=} \sum_{\omega\in\Omega} \Pr(X=\omega)\log\left(\frac{\Pr(X=\omega)}{\Pr(Y=\omega)}\right).$$

Define $\Omega^+ = \{\omega \in \Omega : \Pr(X=\omega) > \Pr(Y=\omega)\}$ and $\Omega^- = \Omega\backslash\Omega^+$. We can split the KL divergence into its positive and negative parts,

$$D(X\|Y) = \mathrm{D}^+(X\|Y) - \mathrm{D}^-(X\|Y),$$

where $\mathrm{D}^+(X\|Y) = \sum_{\omega\in\Omega^+}\Pr(X=\omega)\log(\frac{\Pr(X=\omega)}{\Pr(Y=\omega)})$ and $\mathrm{D}^-(X\|Y) = -\sum_{\omega\in\Omega^-}\Pr(X=\omega)$ $\log(\frac{\Pr(X=\omega)}{\Pr(Y=\omega)})$.

Lemma 2.8. *Let X, Y be random variables over a discrete domain Ω. Then, for every $\Omega' \subseteq \Omega$ it holds that*

$$\sum_{\omega\in\Omega'} \Pr(X=\omega)\log\left(\frac{\Pr(X=\omega)}{\Pr(Y=\omega)}\right) \leq \mathrm{D}^+(X\|Y).$$

Proof. Immediate from the definition of $\mathrm{D}^+(\cdot\|\cdot)$. □

Lemma 2.9 (Pinsker Inequality [34]). *Let X, Y be random variables over a discrete domain Ω, then*

$$\|X - Y\|^2 \leq 2\ln(2)\cdot \mathrm{D}(X\|Y),$$

where $\|X - Y\| = \sum_{\omega\in\Omega}|\Pr(X=\omega) - \Pr(Y=\omega)|$.

We now upper bound the negative part of the KL divergence. Note that one can easily show that $\mathrm{D}^-(X\|Y) \leq 1$, but we will need a better upper bound that applies when $\mathrm{D}^-(X\|Y) \ll 1$.

Lemma 2.10.

$$\mathrm{D}^-(X\|Y) \leq \sqrt{\frac{2}{\ln(2)}\mathrm{D}(X\|Y)}.$$

Proof. For every $\omega \in \Omega$, let $p_\omega \overset{\text{def}}{=} \Pr(X=\omega)$ and $q_\omega \overset{\text{def}}{=} \Pr(Y=\omega)$. We can relate any negative term of the divergence with a difference of probabilities via the following claim:

Claim 2.11. *For $p_\omega \leq q_\omega$ it holds that* $\ln(2)p_\omega\log\frac{q_\omega}{p_\omega} \leq q_\omega - p_\omega$.

Proof. Note that the equality holds for $p_\omega = q_\omega$. If we take the derivative with respect to q_ω, the LHS is $\frac{p_\omega}{q_\omega}$ and the RHS is 1. Since $\frac{p_\omega}{q_\omega} \leq 1$ when $p_\omega \leq q_\omega$, the claim holds. □

Note that by definition, $\mathrm{D}^-(X\|Y) = \sum_{\omega:\, p_\omega\leq q_\omega} p_\omega\log\frac{q_\omega}{p_\omega}$. From the claim above it holds that $\mathrm{D}^-(X\|Y) \leq \frac{1}{\ln(2)}\|X - Y\|$. The lemma then follows from Pinsker's inequality (Lemma 2.9). □

2.3 Technical Lemmas

We now prove several technical lemmas which we will use for our lower bound proof. Their operational meaning will be explained in Section 5.

Lemma 2.12. *Let $Z, D, X_1, \ldots, X_n$ be random variables. Let $f : Z \to [n]$ be some function. Suppose that, conditioned on $D = d$, Z and $(X_1, \ldots, X_n)$ are independent. Denote the guessing probability $p_{\max}(f(Z) \mid D = d) = 2^{-H_\infty(f(Z) \mid D=d)}$, then*

$$\mathbb{E}_{z\sim Z|D=d} I(X_{f(Z)} \mid D = d, Z = z) \leq p_{\max}(f(Z) \mid D = d) \cdot I(X_1, \ldots, X_n \mid D = d).$$

Proof.

$$\begin{aligned}
\mathbb{E}_{z\sim Z|D=d} I(X_{f(Z)} \mid D = d, Z = z) &= \sum_{z} \Pr(Z = z \mid D = d) I(X_{f(z)} \mid D = d, Z = z) \\
&= \sum_{i=1}^{n} \left(\sum_{z:f(z)=i} \Pr(Z = z \mid D = d) \right) I(X_i \mid D = d) \\
&= \sum_{i=1}^{n} \Pr(f(Z) = i \mid D = d) I(X_i \mid D = d) \\
&\leq \sum_{i=1}^{n} p_{\max}(f(Z) \mid D = d) \cdot I(X_i \mid D = d) \\
&\leq p_{\max}(f(Z) \mid D = d) \cdot I(X_1, \ldots, X_n | D = d).
\end{aligned}$$

The second line follows since Z and $(X_1, \ldots, X_n)$ are independent conditioned on $D = d$, by grouping together terms with the same $f(Z)$ value. The last inequality follows from the super-additivity of information (Lemma 2.3). □

Lemma 2.13. *Let $X_1, \ldots, X_n \geq 0$ and $Y_1, \ldots, Y_n \geq 0$ be random variables, with expectations $\mu_i = \mathbb{E}[X_i]$ and $\xi_i = \mathbb{E}[Y_i]$, and assume that $\sum_{i=1}^{n} \mu_i \leq C_1$ and $\sum_{i=1}^{n} \xi_i \leq C_2$, for some constants C_1, C_2. Set $M(t_1, t_2) = \operatorname{argmin}_i\{(X_i < t_1) \wedge (Y_i < t_2)\}$ to be the minimal index i for which both $X_i < t_1$ and $Y_i < t_2$. Then,*

$$\mathbb{E}[M(t_1, t_2)] \leq 1 + \frac{C_1}{t_1} + \frac{C_2}{t_2}.$$

Proof.

$$\begin{aligned}
\mathbb{E}[M(t_1, t_2)] &= \sum_{i=1}^{n} \Pr[M(t_1, t_2) \geq i] \\
&\leq 1 + \sum_{i=1}^{n} \Pr[M(t_1, t_2) > i] \\
&= 1 + \sum_{i=1}^{n} \Pr[(X_1 \geq t_1 \vee Y_1 \geq t_2) \wedge \cdots \wedge (X_i \geq t_1 \vee Y_i \geq t_2)]
\end{aligned}$$

$$
\begin{aligned}
&\leq 1 + \sum_{i=1}^{n} \Pr[X_i \geq t_1 \vee Y_i \geq t_2] \\
&\leq 1 + \sum_{i=1}^{n} (\Pr[X_i \geq t_1] + \Pr[Y_i \geq t_2]) \\
&\leq 1 + \sum_{i=1}^{n} \left(\frac{\mu_i}{t_1} + \frac{\xi_i}{t_2} \right) \\
&\leq 1 + \frac{C_1}{t_1} + \frac{C_2}{t_2},
\end{aligned}
$$

where the penultimate inequality is due Markov's inequality. □

Lemma 2.14. *Let T be a set of binary random variables, ordered as a tree of depth n. For any fixed path P of depth $i \leq n$ starting from the root of T, let $T[P]$ be the set of variables along that path, and let $p_{\max}(T[P]) = 2^{-H_\infty(T[P])}$ be the maximal probability that some assignment to $T[P]$ can obtain. For any $i \leq n$ define*

$$p_{\max}(i) = \max_{P \text{ s.t. } |P|=i} \{p_{\max}(T[P])\} .$$

Then for any $t \geq 6$ it holds that

$$\sum_{i=t}^{n} p_{\max}(i) < 2I(T) + 4\sqrt{I(T)} + 20 \cdot 2^{-t/3}.$$

This lemma is an immediate corollary of the following stronger Lemma 2.15, that proves a similar claim when considering any subset S of n random variables of arbitrary size $|\Sigma|$. In particular, for the special case of Lemma 2.14, the random variables are binary $|\Sigma| = 2$, and the subset S contains variables along a single path in T (note that the parameter n in the above lemma corresponds to $|S|$ of Lemma 2.15).

Lemma 2.15. *Let $B = (B_1, \ldots, B_n)$ be a sequence of n discrete random variables, where $B_i \in \Sigma$. For any $S \subseteq [n]$ we let $B(S) \overset{\text{def}}{=} \{B_i \mid i \in S\}$ be the variables indexed by S. Let $p_{\max}(B(S)) = 2^{-H_\infty(B(S))}$ be the maximal probability that $B(S)$ can attain. For $1 \leq i \leq n$, let*

$$p_{\max}(i) \overset{\text{def}}{=} \max_{|S|=i} p_{\max}(B(S)).$$

Then it holds that for any $t \geq \frac{2e}{\log |\Sigma|}$,

$$\sum_{i=t}^{n} p_{\max}(i) < 2I(B) + 4\sqrt{I(B)} + 20 \cdot |\Sigma|^{-t/3}.$$

Proof. Let Σ be a fixed finite set. For any given string $a \in \Sigma^n$ we let $\nu_a \overset{\text{def}}{=} \Pr[B = a]$ the probability that B attains the value a.

For any $1 \leq i \leq n$, consider $p_{\max}(i)$, and fix $S_i \subset [n]$ of size $|S_i| = i$ and $\beta_i = b_1 b_2 \cdots b_i \in \Sigma^i$ to be the specific values for which $\Pr[B(S_i) = \beta_i] = p_{\max}(i)$, i.e., the certain subset of size i of variables in B and their assignments that are maximal; we know that at least one such subset and assignment exists by the definition of $p_{\max}(i)$.

Define $V_i \overset{\text{def}}{=} \{a \in \Sigma^n \mid a(S_i) = \beta_i\}$ to be all the strings a of length n over Σ whose restriction to S_i equals β_i. Define

$$W_i \overset{\text{def}}{=} V_i \setminus \left(\bigcup_{j>i} V_j \right)$$

to be the set of all the strings $a \in \Sigma^n$ such $a(S_i) = \beta_i$, but for any $j > i$, $a(S_j) \neq \beta_j$. Let $w_i \overset{\text{def}}{=} \Pr(W_i) = \sum_{a \in W_i} \nu_a$. Note that the sets $W_1, W_2, \ldots, W_n$ are disjoint by definition, and that $V_i \subseteq \cup_{j=i}^n W_j$. It is easy to verify that $p_{\max}(i) \leq \sum_{j=i}^n w_i$. Then, we get

$$\sum_{i=t}^{n} p_{\max}(i) \leq \sum_{i=t}^{n} \sum_{j=i}^{n} w_j = \sum_{j=t}^{n} (j - t + 1) \cdot w_j \leq \sum_{j=t}^{n} j \cdot w_j. \tag{3}$$

Next, we upper bound the term $\sum_{j=t}^n j w_j$. Fix a specific j and consider the sum $\sum_{a \in W_j} \nu_a \log(|\Sigma|^n \nu_a)$. This sum can be bounded by

$$\sum_{a \in W_j} \nu_a \log(|\Sigma|^n \nu_a) \geq \left(\sum_{a \in W_j} \nu_a \right) \log \left(|\Sigma|^n \frac{\sum_{a \in W_j} \nu_a}{|W_j|} \right) = w_j \log \left(\frac{|\Sigma|^n}{|W_j|} w_j \right) \geq w_j \log(|\Sigma|^j w_j),$$

where the first inequality follows from the convexity of the function[4] $x \log(cx)$, and the last inequality holds since $|W_j| \leq |V_j| \leq |\Sigma|^{n-j}$. Therefore,

$$\sum_{j=t}^{n} \sum_{a \in W_j} \nu_a \log\left(|\Sigma|^n \nu_a\right) \geq \sum_{j=t}^{n} w_j \log\left(|\Sigma|^j w_j\right) = \log|\Sigma| \sum_{j=t}^{n} j w_j + \sum_{j=t}^{n} w_j \log w_j. \tag{4}$$

Claim 2.16. *For any n, t, Σ such that $|\Sigma|^{-t/2} \leq e^{-1}$, it holds that*

$$\sum_{j=t}^{n} w_j \log(1/w_j) \leq 10 \log|\Sigma| \cdot |\Sigma|^{-t/3} + \frac{\log|\Sigma|}{2} \sum_{j=t}^{n} j w_j.$$

Proof. For any given j, split the sum to indices where $w_j < |\Sigma|^{-j/2}$ and indices where $w_j \geq |\Sigma|^{-j/2}$:

$$\sum_{w_j \geq |\Sigma|^{-j/2}} w_j \log(1/w_j) \leq w_j \cdot \log(|\Sigma|^{j/2}) \leq \frac{j \log(|\Sigma|)}{2} w_j$$

and, assuming $|\Sigma|^{-j/2} < e^{-1}$, the function $x \log(1/x)$ is increasing on $(0, e^{-1})$, thus

$$\sum_{w_j \leq |\Sigma|^{-j/2}} w_j \log(1/w_j) \leq |\Sigma|^{-j/2} \log(|\Sigma|^{j/2}) \leq \frac{j \log|\Sigma|}{2|\Sigma|^{j/2}}.$$

Combining the above, we get

$$\begin{aligned}\sum_{j=t}^{n} w_j \log(1/w_j) &\leq \frac{\log|\Sigma|}{2} \sum_{j=t}^{n} j w_j + \frac{\log|\Sigma|}{2} \sum_{j=t}^{n} \frac{j}{|\Sigma|^{j/2}} \\ &\leq \frac{\log|\Sigma|}{2} \sum_{j=t}^{n} j w_j + 10 \log|\Sigma| \cdot |\Sigma|^{-t/3},\end{aligned}$$

where the last inequality is a crude bound that follows from the fact that $|\Sigma| \geq 2$, and that the infinite sum $\sum_{j=t}^{\infty} \frac{j}{x^j}$ for any $x > 1$ converges to $\frac{(t(\sqrt{x}-1)+1)x^{1/2-t/2}}{(\sqrt{x}-1)^2}$. □

[4] Recall that any convex function f satisfies $\frac{f(x_1)+\cdots+f(x_n)}{n} \geq f(\frac{x_1+\cdots+x_n}{n})$.

Continuing with Equation (4), it follows from Claim 2.16 that

$$\begin{aligned}\sum_{j=t}^{n}\sum_{a\in W_j} v_a \log\left(|\Sigma|^n v_a\right) &\geq \log|\Sigma| \sum_{j=t}^{n} jw_j + \sum_{j=t}^{n} w_j \log w_j\\ &\geq \log|\Sigma| \sum_{j=t}^{n} jw_j - \left(10\log|\Sigma|\cdot|\Sigma|^{-t/3} + \frac{\log|\Sigma|}{2}\sum_{j=t}^{n} jw_j\right)\\ &\geq \frac{\log|\Sigma|}{2}\sum_{j=t}^{n} jw_j - 10\log|\Sigma|\cdot|\Sigma|^{-t/3}.\end{aligned}$$

Rearranging the above we conclude that

$$\sum_{j=t}^{n} jw_j \leq \frac{2}{\log|\Sigma|}\sum_{j=t}^{n}\sum_{a\in W_j} v_a \log(|\Sigma|^n v_a) + 20\cdot|\Sigma|^{-t/3}. \tag{5}$$

Denote by U_Σ the random variable sampled uniformly from Σ, and let $U_\Sigma^{\otimes n}$ be n independent instances of U_Σ. Note that $\mathrm{D}(B\|U_\Sigma^{\otimes n}) = I(B)$ by definition. Furthermore, since the W_j are disjoint, we have by Lemma 2.8 (and Definition 2.7) that

$$\begin{aligned}\sum_{j=t}^{n}\sum_{a\in W_j} v_a \log(|\Sigma|^n v_a) &\leq \mathrm{D}^+(B\|U_\Sigma^{\otimes n})\\ &= \mathrm{D}(B\|U_\Sigma^{\otimes n}) + \mathrm{D}^-(B\|U_\Sigma^{\otimes n}).\end{aligned}$$

Substituting the above into Equation (5), and noting that Lemma 2.10 implies that $\mathrm{D}^-(B\|U_\Sigma^{\otimes n}) \leq \sqrt{\frac{2}{\ln 2}I(B)}$, we get

$$\sum_{j=t}^{n} jw_j \leq 2I(B) + \frac{1}{\log|\Sigma|}\sqrt{\frac{8}{\ln 2}I(B)} + 20\cdot|\Sigma|^{-t/3}.$$

The above and Equation (3) complete the proof. □

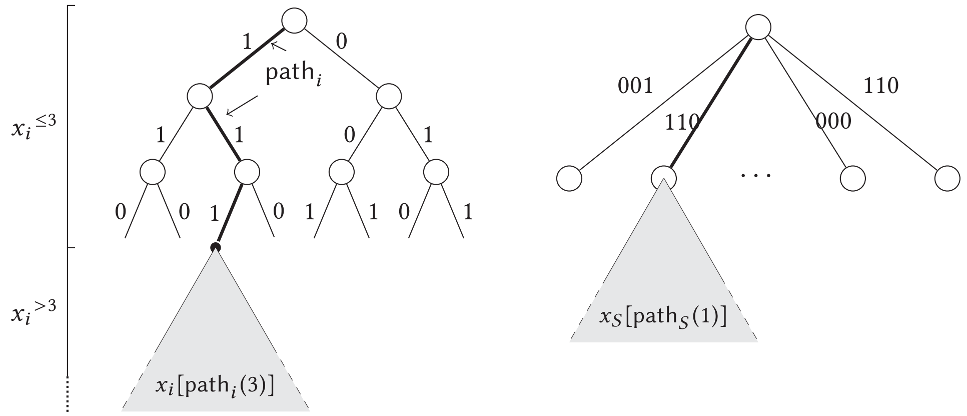
3 MULTIPARTY INTERACTIVE COMMUNICATION OVER NOISY NETWORKS

In the following, we assume a network of $n+1$ parties that consists of a server p_S and n clients $p_1,\ldots,p_n$. The network consists of a communication channel (p_i, p_S) for every $i\in[n]$, that is, the topology is a star.

3.1 The Pointer Jumping Task

We assume the parties want to compute a generalized *pointer jumping task*. Formally, the pointer jumping task of depth T over star networks is the following. Each client p_i holds a binary tree x_i of depth T where each edge is labeled by a bit b. The server holds a 2^n-ary tree x_S of depth T where each edge of the tree is labeled with an n-bit string from $\{0,1\}^n$.

The server starts from the root of x_S. At each round, the server receives from the clients n bits which it interprets as an index $i\in[2^n]$. The server then transmits back to the clients the label on the i-th edge descending from his current node (one bit per client). The node at the end of this edge becomes the server's new node. Similarly, each client receives at each round a bit b from the server, and sends back the label of the edge indexed by b descending from its current node. For the first round, we can assume that the clients take the left child of the root of x_i and transmit to the server the label of that edge. The above is repeated until both the server and the clients have



(a) A possible input x_i of some client p_i; $\text{path}_i(3)$ is marked with bold edges.

(b) A possible input x_S of the server

Fig. 1. An illustration of the inputs, the "correct" path (marked with bold lines) and the sub-input conditioned on a partial correct path.

reached depth T in their trees. The parties then output the path from the root to their current node (i.e., to a leaf at depth T).

We denote this "correct" output of party p_i by path_i. The entire output is denoted $\text{path} = (\text{path}_S, \text{path}_1, \ldots, \text{path}_n)$. For a certain party $i \in [n] \cup \{S\}$ and a level $1 \leq k \leq T$, we let $\text{path}_i(k)$ be the first k edges of path_i.

We use the following notations throughout. Given any tree $\mathcal{T}$ of depth N, we denote its first k levels by $\mathcal{T}^{\leq k}$ and its $N - k$ last levels by $\mathcal{T}^{>k}$. Given a path $z = (e_1, e_2, \ldots)$, we denote by $\mathcal{T}[z]$ the subtree of $\mathcal{T}$ rooted at the end of the path that begins at the root of $\mathcal{T}$ and follows the edge-sequence z. [For instance, many times z will be the correct path so far (e.g., until some round ℓ) in the input tree x_i; then we will care about the subtrees $x_i[\text{path}_i(\ell)]$, effectively obtaining a new instance of the pointer jumping task, with a smaller depth.] We let $x = (x_S, x_1, \ldots, x_n)$ be the entire input and also use the short notation $x = (x_S, x_{[n]})$ for the server's and clients' parts, respectively. The above notation composes in a straightforward way, e.g., $x^{\leq k}$, ${x_{[n]}}^{\leq k}$, and ${x_S}^{\leq k}$ denote the appropriate set of partial trees in x, $x_{[n]}$, and x_S, respectively, and $x[\text{path}(\ell)]$ denotes the set of subtrees $x_i[\text{path}_i(\ell)]$, for $i \in [n] \cup \{S\}$. We will sometimes be negligent and write $x_i[\text{path}(\ell)]$ for $x_i[\text{path}_i(\ell)]$. See Figure 1 for an illustration of some of the notations.

The above pointer jumping task is complete for the case of a star network. That is, any noiseless protocol over a star network can be described as a pointer jumping task by setting the inputs $(x_S, x_1, \ldots, x_n)$ appropriately. For our purpose, we will have the inputs distributed randomly. That is, for every client, the label on each edge is distributed uniformly in $\{0, 1\}$ independently of all other edges; for the server, the labels are independent and uniform over $\{0, 1\}^n$. We denote the random variable describing the input of p_i by X_i. The correct path also becomes a random variable that we denote PATH_i, and which is a function of the inputs. The same holds for the subtree of a certain input, given the certain path of some depth ℓ and so on.

Lastly, we denote by π an observed transcript (possibly noisy) of the protocol. That is, π is the string *received* by the parties (in some natural order); note that no single party observes the entire transcript π, but each party observes some part of it. The corresponding random variable is denoted Π. At times, π will denote a partial transcript, that is, the communication observed by the parties up to some round k of the protocol.

3.2 Independence of Inputs Conditioned on the Transcript

An important property that will be needed for our lower bound is the fact that the inputs of the users are independent, *even when conditioned on the transcript so far.* This implies that only party p_i is capable of sending useful information about its input x_i, regardless of the transcript so far (and, therefore, if the communication of p_i is noisy, the information is lost; it is impossible that a different party p_j compensates for this loss). This claim is well known in the folklore, and we now prove it formally.

Lemma 3.1. *Conditioned on the observed transcript* Π, *the random variables* $X_S, X_1, \ldots, X_n$ *are mutually independent.*

Proof. The proof goes by induction on the length of Π. The base case where $|\Pi| = 0$ is trivial from the definition of the inputs $X_S, X_1, \ldots, X_n$.

Assume the claim holds for some transcript $\Pi = \pi$ of length $\ell - 1$, and consider the next bit Π_ℓ, sent without loss of generality by p_i, where $i \in \{S\} \cup [n]$. This bit (in case it was not changed by the channel) depends only on X_i and the previous communication Π, that is, $\Pi_\ell = f(\Pi, X_i)$. To simplify notations, denote by $X_{\neq i} = (X_S, X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$ all the variables except X_i. We have

$$\begin{aligned}
&\Pr(X_1 = x_1, \ldots, X_S = x_S \mid \Pi = \pi, \Pi_\ell = b) \\
&= \frac{\Pr(X_1 = x_1, \ldots, X_S = x_S, \Pi_\ell = b \mid \Pi = \pi)}{\Pr(\Pi_\ell = b \mid \Pi = \pi)} && \text{by definition} \\
&= \frac{\Pr(X_{\neq i} = x_{\neq i} \mid \Pi = \pi)\Pr(X_i = x_i, \Pi_\ell = b \mid \Pi = \pi)}{\Pr(\Pi_\ell = b \mid \Pi = \pi)} && \text{by induction, since } X_i, f(X_i, \Pi) \perp X_{\neq i} \mid \Pi \\
&= \left(\prod_{j \neq i} \Pr(X_j = x_j \mid \Pi = \pi)\right) \frac{\Pr(X_i = x_i, \Pi_\ell = b \mid \Pi = \pi)}{\Pr(\Pi_\ell = b \mid \Pi = \pi)} \\
&= \prod_{j \neq i} \Pr(X_j = x_j \mid \Pi = \pi, \Pi_\ell = b) \times \Pr(X_i = x_i \mid \Pi = \pi, \Pi_\ell = b),
\end{aligned}$$

where the last transition follows since X_i and $X_{\neq i}$ are independent given Π, thus conditioning on a function of either X_i or Π does not change the probability.

Finally, note that if b was changed by the channel, $b' = b \oplus E$, the claim still holds since the noise E is independent of all the other variables (i.e., we can condition on E and reduce to the case above). If the bit b was erased (in the case of a BEC) then the claim trivially holds. □

As a corollary to the above, note that, conditioned on any piece of information that the parties can communicate as part of their transcripts, the variables $X_S, X_1, \ldots, X_n$ remain independent. Specifically, the above holds if we condition on the correct path (up to some level), or on some levels of the inputs—we can assume a protocol in which the parties simply communicate that information (so it is a part of Π), and apply the above lemma.

Corollary 3.2. *The random variables* $X_S, X_1, \ldots, X_n$ *are independent, conditioned on the observed transcript* $\Pi = \pi$, *the correct path* PATH = path *(up to some level), and parts of the inputs.* □

4 UPPER BOUND

Showing an upper bound of $O(\log n / \log\log n)$ on the slowdown for multiparty interactive communication on star networks is rather straightforward. Essentially, all that we need to show is that, for every $\log n$ rounds of communication, the parties can advance $\Theta(\log\log n)$ levels in the underlying pointer jumping task.

Theorem 4.1. *For any $\varepsilon < 1/2$ and any $T > 0$, there exists a coding scheme for the pointer jumping task of depth T over a star network with $n+1$ parties, that takes $O_\varepsilon(T\frac{\log n}{\log\log n})$ rounds and succeeds with high probability if each communication channel is a BSC_ε.*

Proof. First, let us recall the existence of good error correction codes.

Lemma 4.2 (Shannon Coding Theorem [40]). *For any discrete memoryless channel* CH *with capacity C and any k, there exists a code* $\mathsf{ECC} : \{0,1\}^k \to \{0,1\}^n$ *and* $\mathsf{ECC}^{-1} : \{0,1\}^n \to \{0,1\}^k$ *with $n = O(\frac{1}{C}k)$ such that for any $m \in \{0,1\}^k$ it holds that*

$$\Pr\left[\mathsf{ECC}^{-1}(\mathsf{CH}(\mathsf{ECC}(m))) \neq m\right] < 2^{-\Omega(n)}.$$

The coding scheme is as follows. Assume that the parties have already correctly solved the pointer jumping task until a certain depth $\gamma \geq 0$. Each client encodes the next $\log\log n$ levels of his input (this is a subtree of size $\log n$, rooted at the current position) using a good Shannon error correcting code given by Lemma 4.2. The encoded message is of length $O(\log n)$, and we are guaranteed that the server can correctly decode the entire subtree with probability $1 - n^{-c}$, for some constant $c > 1$ to our choice. Using a union bound, the server gets all the subtrees of the clients with high probability $1 - n^{-c+1}$. Next, the server computes the correct path (of length $\log\log n$) that corresponds to each party, and sends an encoding of this path to the corresponding party. The process then repeats from the new depth $\gamma + \log\log n$. The entire scheme therefore takes $\frac{T}{\log\log n} \cdot O(\log n)$ rounds and succeeds with probability $1 - \frac{T}{\log\log n} \cdot n^{-\Omega(1)}$.

However, T may be very large with respect to n. To further improve the probability of success and prove Theorem 1.2, we use a theorem by Rajagopalan and Schulman (see Reference [2, Section 3]).

Theorem 4.3 [2, 36]. *For any T round protocol over any n-party network G with maximal degree d, there exists a coding scheme Π, that takes $O(T)$ rounds and succeeds with probability $1 - n$ $(2(d+1)p)^{\Omega(T)}$ given that any symbol transmitted in the network is correctly received with probability $1 - p$.*

In the scheme we describe above, any $\log\log n$ symbols are correctly decoded with probability $1 - p$, where we can choose p to be small enough, e.g., by taking $p = O(n^{-2})$. In this case, Theorem 4.3 guarantees a coding scheme for the pointer jumping task with the same slowdown of $O(\log n/\log\log n)$ as above, which succeeds with probability $1 - n^{-\Omega(T/\log\log n)}$, that is, $1 - 2^{-\Omega(T\log n/\log\log n)}$. □

5 LOWER BOUND

In this section, we prove our main theorem of a lower bound of $\Omega(\frac{\log n}{\log\log n})$ on the slowdown of coding for interactive communication over star networks. Toward the lower bound, we can assume the noisy channel is actually a BEC_ε rather than a BSC_ε. This only makes the noise model weaker, and renders the lower bound stronger. In the following, we assume the channel erasure probability is $\varepsilon = 1/3$. The specific value of $\varepsilon < 1$ only affects the constants involved and does not affect the validity of our result. Fixing its value will allow an easier exposition of the result.

Our main theorem is the following,

Theorem 5.1. *There exists a constant c such that for large enough n, any protocol that solves the pointer jumping task of depth T (for some $T > \log\log n$) over star networks with $n+1$ parties in less than $c \cdot T\frac{\log n}{\log\log n}$ rounds, assuming each communication channel is a $\mathsf{BEC}_{1/3}$, has a success probability at most $1/5$.*

We begin by defining the cutoff of the protocol: an information-based measure of progress that is related to the advancement in the underlying pointer jumping task.[5]

Definition 5.2 (Cutoff). For any transcript π, and any input $x = (x_s, x_1, \ldots, x_n)$, the *cutoff of the protocol* $\mathsf{cutoff}(\pi, x)$ is the minimal number k, such that both the equations below are satisfied:

$$I(X_S[\mathsf{path}_S(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)) \le 2^{-0.1\sqrt{n}}, \text{and} \tag{6}$$

$$\sum_{i=1}^{n} I(X_i[\mathsf{path}_i(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)) \le 0.01n. \tag{7}$$

If no such k exists we set $\mathsf{cutoff} = T$.

The operational meaning of the cutoff is that if k is the cutoff level, then the parties know very little information on the correct paths beyond the first k edges in that path. Recall that $\mathsf{path}(k)$ is fully determined by the first k levels of the input, $x^{\le k}$. Therefore, if $\mathsf{cutoff}(\pi, x) = k$, then for any x' such that $x'^{\le k} = x^{\le k}$, it holds that $\mathsf{cutoff}(\pi, x') = k$. Furthermore, the cutoff is only a function of the path up to level k, that is, if $\mathsf{cutoff}(\pi, x) = k$ then for any input x' that induces the same $\mathsf{path}(k)$ as x, it holds that $\mathsf{cutoff}(\pi, x') = k$; When the path is fixed (but we do not care about the specific input), we will usually abuse notations and write $\mathsf{cutoff}(\pi, \mathsf{path}(k)) = k$.

Our analysis will actually bound the cutoff in two steps. Very roughly, at the first step, we will bound the information of the server given a "new" chunk of communication π^{new}, yet bound the clients' information without this new chunk. At the second step we condition on π^{new} both for the server and the clients. For the first step described above, we define the following "server cutoff":

Definition 5.3 (Server Cutoff). Given any transcript π, and any input $x = (x_S, x_1, \ldots, x_n)$, and given any continuation of the transcript π^{new}, we define the *server's cutoff level* $\mathsf{cutoff}_S(\pi, \pi^{new}, x)$ as the minimal number k for which

$$I(X_S[\mathsf{path}_S(k)] \mid \underline{\Pi = \pi \circ \pi^{new}}, \mathsf{PATH}(k) = \mathsf{path}(k)) \le 2^{-0.2\sqrt{n}}, \text{and} \tag{8}$$

$$\sum_{i=1}^{n} I(X_i[\mathsf{path}_i(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)) \le 0.01n. \tag{9}$$

If no such k exists we set $\mathsf{cutoff}_S = T$.

The following proposition shows that in order for a protocol to output the correct value with high probability, the cutoff (given the complete transcript) must be $\approx T$. Hence, protocols that succeed with high probability must produce transcripts whose cutoff is large in expectation.

Proposition 5.4. *Fix a protocol that solves the pointer jumping task of depth T over a star network with $n+1$ parties, that succeeds with probability at least $1/5$ on average, i.e., a protocol for which* $\Pr_{X,\Pi}(\textit{correct output}) \ge 1/5$. *Then,*

$$\mathbb{E}_{X,\Pi}[\mathsf{cutoff}(\Pi, X)] \ge \left(\frac{1}{5} - \frac{2}{n}\right) T.$$

Proof. Recall that the event $\mathsf{cutoff}(\pi, x) = k$ depends only on π and $\mathsf{path}(k)$ and is independent of $x^{>k}$. We show that if $\mathsf{cutoff}(\pi, \mathsf{path}(k)) = k$ for some $k < T$, then the protocol gives the correct output with only small probability of $2/n$. This will bound the probability of the event

[5] The reader is encouraged to recall the definition of the pointer jumping task and our notations such as $\mathsf{path}(k)$, $X_i[\mathsf{path}_i(k)]$, and the like, stated in Section 3.1.

$\mathsf{cutoff}(\Pi, X) < T$ by $1/5 - 2/n$, and will prove that in expectation (over all inputs and possible transcripts), the cutoff is at least $T/5 - 2T/n$.

Claim 5.5. *Given π and $k < T$ and $\mathsf{path}(k)$ such that $\mathsf{cutoff}(\pi, \mathsf{path}(k)) = k$,*

$$\Pr[\text{correct output} \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)] < \frac{2}{n}.$$

Proof. Let L be the n-bit label of $\mathsf{PATH}_S(k+1)$. Note that this label is included in the subtree $X_S[\mathsf{path}(k)]$. If $\mathsf{cutoff}(\pi, \mathsf{path}(k)) = k$, then by the cutoff's definition,

$$I(X_S[\mathsf{path}_S(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)) \le 2^{-0.1\sqrt{n}},$$

and by Lemma 2.6 it holds that

$$2^{-H_\infty(L \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k))} \le \frac{1 + 2^{-0.1\sqrt{n}}}{|L|} \le \frac{2}{n}.$$

Then, the probability that the protocol is correct is at least the probability that the clients (here treated as a single party) output the correct label L:

$$\begin{aligned}\Pr[\text{correct output} \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)] &\le 2^{-H_\infty(L \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k), X_{[n]})} \\ &= 2^{-H_\infty(L \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k))} \\ &\le \frac{2}{n},\end{aligned}$$

where the equality holds since the input of the server is independent of the input of the users conditioned on π and $\mathsf{path}(k)$. This is implied by Lemma 3.1 (as also stated by Corollary 3.2): consider a protocol that, after completing the pointer jumping task, communicates the correct path during its last T rounds. That is, $\mathsf{path}(k)$ is simply part of the transcript of this protocol. Now Lemma 3.1 suggests that, because the inputs are independent when conditioned on that transcript, and because the path is simply the suffix of the transcript, then the inputs are independent conditioned on both the correct path and the prefix of the transcript (that doesn't contain the path). □

The above holds for any $k < T$ and any $\pi, \mathsf{path}(k)$ for which $\mathsf{cutoff}(\pi, \mathsf{path}(k)) = k$. Therefore, conditioned on the event that $\mathsf{cutoff}(\Pi, X) < T$ the protocol outputs the correct value with probability at most $2/n$, that is, $\Pr_{X,\Pi}[\text{correct output} \mid \mathsf{cutoff}(\Pi, X) < T] \le 2/n$. Since the protocol is correct with probability $1/5$ on average over the inputs and randomness of the protocol (and the noise), the claim follows. Indeed,

$$\begin{aligned}\frac{1}{5} &\le \Pr_{X,\Pi}[\text{correct output}] \\ &= \Pr[\mathsf{cutoff}(\Pi, X) < T] \Pr[\text{correct output} \mid \mathsf{cutoff}(\Pi, X) < T] \\ &\quad + \Pr[\mathsf{cutoff}(\Pi, X) = T] \Pr[\text{correct output} \mid \mathsf{cutoff}(\Pi, X) = T] \\ &\le \Pr[\mathsf{cutoff}(\Pi, X) < T] \cdot 2/n + \Pr[\mathsf{cutoff}(\Pi, X) = T] \cdot 1,\end{aligned}$$

ergo,

$$\Pr[\mathsf{cutoff}(\Pi, X) = T] \ge \frac{1}{5} - \frac{2}{n}$$

and

$$\mathbb{E}_{X,\Pi}[\mathsf{cutoff}(\Pi, X)] \ge T\left(\frac{1}{5} - \frac{2}{n}\right),$$

as claimed. □

To prove the main theorem, we show that during every $0.1 \log n$ rounds of communication, the cutoff level increases by at most $O(\log\log n)$, in expectation. Formally, Theorem 5.6. *Given a protocol for the pointer jumping task, let π be the transcript of the protocol observed up to some round, and let Π^{new} be a random variable describing the observed transcript over the next $0.1\log n$ rounds. Then, for any $\ell \le T$, and for any $x^{\le \ell}$ it holds that*

$$\mathbb{E}\left[\mathsf{cutoff}(\pi \circ \Pi^{new}, X) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \mathsf{cutoff}(\pi, X) = \ell\right] \le \ell + O(\log\log n).$$

Note that the expectation is over the inputs, the noise, and the protocol's randomness.

With the above propositions, the proof of Theorem 5.1 is immediate: if a protocol outputs the correct answer with probability at least 1/5, it must be that the expected cutoff level at the end of the protocol is $> T/5 - o(T)$, but this would take $O(T\frac{\log n}{\log\log n})$ rounds of communication, in expectation. Formally,

Proof (Theorem 5.1). Using Theorem 5.6, for any protocol for the pointer jumping task there exists a (small enough) constant $c > 0$ such that after running $cT\frac{\log n}{\log\log n}$ rounds of the protocol, the expected cutoff for the observed transcript is small, $\mathbb{E}_{X,\Pi}[\mathsf{cutoff}(\Pi, X)] < T/10$. Therefore, it cannot be that the protocol correctly solves the T-depth pointer jumping with probability above 1/5 as this will contradict Proposition 5.4. □

We now turn to prove the key technical Theorem 5.6. Intuitively speaking, the main idea is the following. We cut the protocol into chunks of length $0.1\log n$ rounds and treat each one separately, showing that the cutoff level cannot increase during any chunk by more than $O(\log\log n)$ in expectation. We can assume that at the beginning of each chunk, all the parties are given the information about the correct path up to the depth matching the current cutoff level, and reduce this case (in some sense[6]) to a new instance of the pointer jumping task starting at that depth.

During the $0.1\log n$ rounds of the next chunk, with probability at least $1 - 2^{-\sqrt{n}}$, there exists a subset Q of $\sqrt{n}$ parties about which the server does not have much information (beyond the cutoff point) whose communication was *completely erased* by the channel throughout this chunk. We can assume that, other than this set of parties Q, the communication is noiseless. In this case, it is quite intuitive that the cutoff level cannot increase by too much: the server is missing any relevant information about the inputs of parties in Q beyond the cutoff level, thus the information that it sends during that chunk is practically meaningless, and the server's cutoff level remains more or less the same. Additionally, since the server did not communicate a lot of meaningful information about his input, the clients do not know how to proceed and cannot send too much relevant information; thus, their cutoff level does not increase too much as well. On the other hand, in the rare case where no subset Q exists (i.e., the communication in this chunk is practically noiseless), the cutoff may tremendously increase; however, since this event is so rare, it will add only $O(1)$ to the accumulated cutoff level throughout the entire protocol, in expectation.

Proof (Theorem 5.6). We begin by showing that with high probability, there exists a subset of size $\sqrt{n}$ of the clients, for which the server knows very little information beyond the cutoff level, and yet in the next $0.1n\log n$ rounds their communication was completely erased by the channel.

Definition 5.7. Given a transcript π and an input x so that $\mathsf{cutoff}(\pi, x) = k$. For $i \in [n]$, we say that a client p_i is *critical* if

$$I\left(X_i[\mathsf{PATH}_i(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)\right) \le 0.02.$$

[6] The main difference is that previous communication may have leaked some information on this new instance, and we need to account for this information as well.

Lemma 5.8. *Let π be the transcript so far and consider the next $0.1 \log n$ rounds of communication. Denote by $E_{silence}$ the event that there exists a subset Q of parties of size at least $\sqrt{n}$, such that all the parties in Q are critical and all the bits sent by parties in Q were erased by the channel. Then,*

$$\Pr[E_{silence}] > 1 - 2^{-\sqrt{n}}.$$

Proof. There are at least $n/2$ *critical* parties, or otherwise,

$$\sum_i I\left(X_i[\mathsf{PATH}_i(k)] \mid \Pi = \pi, \mathsf{PATH}(k) = \mathsf{path}(k)\right) \geq \frac{n}{2} \cdot 0.02 \geq 0.01n,$$

and k cannot be the cutoff round, by Definition 5.2. Moreover, note that the probability that all the $0.1 \log n$ transmissions of a specific party p_i are erased (or even the $0.2 \log n$ bits sent and received by this party), is $\frac{1}{3}^{0.1 \log n} \geq n^{-0.4}$. Let Q be the set of all critical parties whose entire communication was erased by the channel. By Chernoff bound and assuming large enough n we have

$$\Pr\left[|Q| < \sqrt{n}\right] < \exp\left(-\frac{n^{0.6}}{4}\right).$$

Here we use the fact that $\varepsilon = 1/3$; however, it is clear that for any other constant ε, we can reduce the length of a chunk to be $c \log n$ such that, say, $\varepsilon^{c \log n} \geq n^{-0.4}$ and all the other proofs below remain valid, maybe up to adjusting the constants as needed. □

For any $\ell \leq T$, any fixing $\mathsf{path}(\ell)$, and any transcript π, denote by $E_{(\pi, \mathsf{path}(\ell), \ell)}$, the event that $(\Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \mathsf{cutoff}(\pi, X) = \ell)$. Recall that whether the cutoff is ℓ depends only on π and the first ℓ levels the correct path, therefore $E_{(\pi, \mathsf{path}(\ell), \ell)}$ is either empty or equal to $(\Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell))$. For any continuation π_S^{new} of bits sent by the server in the new chunk define $E^S_{(\pi, \pi_S^{new}, \mathsf{path}(\ell), \ell)}$ the event $(\Pi = \pi, \Pi_S^{new} = \pi_S^{new}, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \mathsf{cutoff}_S (\pi, \Pi_S^{new}, X) = \ell)$.

The proof of the theorem will follow from the next three propositions:

Proposition 5.9. *For any $\ell \leq T$, any $\mathsf{path}(\ell)$ and any transcript π*

$$\mathbb{E}[\mathsf{cutoff}_S(\pi, \Pi^{new}, X) \mid E_{(\pi, \mathsf{path}(\ell), \ell)}, E_{silence}] \leq \ell + 40.$$

Proposition 5.10. *Split the observed new transcript $\Pi^{new} = (\Pi_S^{new}, \Pi_{[n]}^{new})$ to the parts corresponding to information sent by the server and by the clients, respectively. For any $\ell' \leq \ell \leq T$, any fixing $\mathsf{path}(\ell)$, any transcript π, and any (server's) new transcript π_S^{new},*

$$\mathbb{E}[\mathsf{cutoff}(\pi \circ \Pi^{new}, X) \mid E_{(\pi, \mathsf{path}(\ell'), \ell')}, E^S_{(\pi, \pi_S^{new}, \mathsf{path}(\ell), \ell)}, E_{silence}] \leq \ell + 5 \log \log n.$$

Proposition 5.11. *For any $\ell \leq T$, any fixing $\mathsf{path}(\ell)$ and any transcript π*

$$\mathbb{E}[\mathsf{cutoff}(\pi \circ \Pi^{new}, X) \mid E_{(\pi, \mathsf{path}(\ell), \ell)}, \overline{E_{silence}}] \leq \ell + O(n \log n \log \log n).$$

The above three propositions prove the theorem: When the good event $E_{silence}$ doesn't happen, the cutoff increases by at most $O(n \log n \log \log n)$ (Proposition 5.11), but this happens with probability at most $\Pr[\overline{E_{silence}}] < 2^{-\sqrt{n}}$ (Lemma 5.8), thus the expected contribution to the increase of the cutoff by such chunks is bounded by a negligible amount of $O(n \log n \log \log n) \cdot 2^{-\sqrt{n}}$. Otherwise, assuming the previous cutoff was ℓ, then with the information of Π^{new}, the server's cutoff level, according to Proposition 5.9, is in expectation at most $\ell^S \leq \ell + 40$. Finally, given that the server's cutoff is ℓ^S, Proposition 5.10 guarantees that the new cutoff (i.e., when considering Π^{new} for both the server and the clients), is in expectation at most $\ell^S + 5 \log \log n = \ell + O(\log \log n)$. □

In the following three subsections, we prove the above three propositions in turn.

5.1 Bounding the Server's Cutoff: Proof of Proposition 5.9

To prove Proposition 5.9, we need to find the minimal round k that satisfies Equations (8) and (9), and show that this round is in expectation at most $\ell + 40$, provided that the old cutoff level is ℓ, and that $E_{silence}$ occurs. We begin in Section 5.1.1 by bounding the information on X_S revealed by the transcript so far, as a function of k, towards satisfying Equation (8). In Section 5.1.2, we bound the information on the X_i's as a function of k, towards satisfying Equation (9). Finally, in Section 5.1.3, we use the two bounds on the information to derive a bound the new server's cutoff k.

5.1.1 Bounding the Information in Equation (8). Recall the setting: the protocol has run for some rounds, producing the transcript $\Pi = \pi$ so that the cutoff until that point is ℓ. In other words, we are given $(\pi, \mathsf{path}(\ell)) \in E_{(\pi, \mathsf{path}(\ell), \ell)}$.

Now we run the protocol for another $0.1 \log n$ rounds and obtain a new transcript Π^{new}, describing the bits observed in those new $0.1 \log n$ rounds, up to erasures. We condition on the event $E_{silence}$ that guarantees that there is a set of $\sqrt{n}$ critical clients whose communication (in Π^{new}) was completely erased. Next, we reveal to all parties the correct path of depth ℓ [i.e., we condition on $\mathsf{PATH}(\ell) = \mathsf{path}(\ell)$], and we wish to find the expected new cutoff induced by $\pi \circ \Pi^{new}$.

Let us first set some notations that will be used throughout the first part of the proof. Let $Z(k) = \mathsf{PATH}_S(k + \ell)$ be the correct path of length k in X_S *below the cutoff level.*[7] Given specific transcripts π, π^{new}, a specific path $\mathsf{path}(\ell)$ and specific fixing $x_S[\mathsf{path}_S(\ell)]^{\leq k}$ of the k first levels of the input of the server in the subtree induced by $\mathsf{path}_S(\ell)$, we define the short-handed events

$$\mathcal{E} \stackrel{\text{def}}{=} (\Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)), \text{ and}$$
$$\mathcal{E}^+ \stackrel{\text{def}}{=} (\mathcal{E}, X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}).$$

The information measure in Equation (8) conditions exactly on $\mathcal{E}$. However, we will need to condition on even a smaller event space (i.e., on $\mathcal{E}^+$) to utilize independence between several variables. To this end, we use the following claim, that proves an independence between the correct path (between levels ℓ and $\ell + k$) and the server's input at depths below $\ell + k$, when conditioning on $\mathcal{E}^+$. This will be instrumental when using Lemma 2.12 to bound the information measure related with the cutoff.

Claim 5.12. *Conditioned on the event*

$$\mathcal{E}^+ = \left(X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)\right),$$

the variables $\mathsf{PATH}(k + \ell)$ *and* $X_S[Z(0)]^{>k}$ *are independent.*

Proof. Once we condition on $X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}$ and $\mathsf{PATH}_S(\ell) = \mathsf{path}_S(\ell)$, then $\mathsf{PATH}(k + \ell)$ becomes a function only of ${X_{[n]}}^{>\ell} \cap {X_{[n]}}^{\leq k+\ell}$, and these are all independent of $X_S^{>\ell}$, when conditioned on the transcript and on the other parts of $\mathcal{E}^+$ (which can be included as part of the transcript), via Corollary 3.2. □

[7] Since we condition on $\mathsf{PATH}(\ell) = \mathsf{path}(\ell)$, the remaining unfixed random variables are only the suffix of length k.

We now get to the core of the proof. For any $k > 0$, define the random functions

$$S^*(k \mid \pi^{new}, \mathsf{path}(k+\ell)) \overset{\text{def}}{=}$$
$$I(X_S[\mathsf{path}_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \mathsf{path}(k+\ell)),$$
$$S(k \mid \pi^{new}, x_S[\mathsf{path}_S(\ell)]^{\leq k}) \overset{\text{def}}{=}$$
$$\mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid \mathcal{E}^+}$$
$$I(X_S[\rho_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \rho, X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}_S(\ell)]^{\leq k}).$$

To clarify the above notation, note that $\rho \sim \mathsf{PATH}(k+\ell)$ is a variable of the expectation going over all respective paths of length $k+\ell$ (for all parties), and we can write $\rho = (\rho_1, \ldots, \rho_n, \rho_S)$ according to its parts.

The random variables $S^*(k)$ precisely describe the measure we need to bound for Equation (8); however, we will actually bound the measure $S(k)$, which is similar to $S^*(k)$ up to conditioning on $\mathcal{E}^+$ rather than on $\mathcal{E}$. The measure $S(k)$ upper bounds $S^*(k)$ in expectation (via Claim 5.13 below), thus it suffices to bound $S(k)$ to obtain a bound on $S^*(k)$ and satisfy Equation (8). We take this detour because we cannot bound $S^*(k)$ directly; however, bounding $S(k)$ is possible once we take advantage of the independence between PATH and X_S in non-overlapping depths of the trees, as stated by Lemma 5.12.

Claim 5.13. *Given any $\pi, \pi^{new}, \mathsf{path}(\ell)$ and any k,*

$$\mathbb{E}_{\mathsf{path}(k+\ell) \mid \mathcal{E}, E_{silence}} S^*(k \mid \pi^{new}, \mathsf{path}(k+\ell)) \leq \mathbb{E}_{x_S[\mathsf{path}_S(\ell)]^{\leq k} \mid \mathcal{E}, E_{silence}} S(k \mid \pi^{new}, x_S[\mathsf{path}_S(\ell)]^{\leq k}).$$

Proof. First, note that $\mathcal{E}$ determines whether $E_{silence}$ occurs or not (indeed: π^{new} determines which bits are erased, and $\pi, \mathsf{path}(\ell)$ determine the set of critical parties), therefore it suffices to condition on $\mathcal{E}$ alone. Starting with the definition of $S(k)$,

$$\mathbb{E}_{x_S[\mathsf{path}_S(\ell)]^{\leq k} \mid \mathcal{E}} S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k})$$
$$= \mathbb{E}_{x_S[\mathsf{path}_S(\ell)]^{\leq k} \mid \mathcal{E}}$$
$$\mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid x_S[\mathsf{path}_S(\ell)]^{\leq k}, \mathcal{E}}$$
$$I(X_S[\rho_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \rho, X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}_S(\ell)]^{\leq k}),$$

exchanging the order of expectations, and using Definition 2.2,

$$= \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid \mathcal{E}}$$
$$\mathbb{E}_{x_S[\mathsf{path}_S(\ell)]^{\leq k} \mid \rho, \mathcal{E}}$$
$$I(X_S[\rho_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \rho, X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}_S(\ell)]^{\leq k})$$
$$= \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid \mathcal{E}} I(X_S[\rho_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \rho, X_S[Z(0)]^{\leq k}),$$

conditioning on $X_S[Z(0)]^{\leq k}$ can only increase the information (Lemma 2.4), thus,

$$\geq \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid \mathcal{E}} I(X_S[\rho_S(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \rho)$$
$$= \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid \mathcal{E}} S^*(k \mid \pi^{new}, \rho).$$ □

Now we can bound the measure $S(k)$. We show that the expected sum of this quantity, for $k \geq 30$, is exponentially small. This will be used in Section 5.1.3 to show that the first k^* for which $S^*(k^*) < 2^{-0.2\sqrt{n}}$ as required for Equation (8), is bounded in expectation by 40.

Lemma 5.14. *Given any* $(\pi, \mathsf{path}(\ell)) \in E_{(\pi,\mathsf{path}(\ell),\ell)}$,

$$\sum_{k=30}^{T-\ell} \mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \left[S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k}) \right] \le n \log n \cdot 2^{-0.5\sqrt{n}}.$$

Proof. The outline of the proof is as follows. First we use Lemma 2.12 to bound $S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k})$ as the product of the probability to guess the correct path between layers ℓ and $\ell + k$, and the information on the subtrees rooted in level $\ell + k$. We then bound each part independently to obtain the stated claim.

Let the $\{X_i\}$ of Lemma 2.12 be all the subtrees of X_S rooted at the end of a path of depth $k + \ell$, whose prefix is $\mathsf{path}_S(\ell)$. Note that those subtrees and (the last k edges in each of) $\mathsf{PATH}(k + \ell)$ are independent conditioned on $\mathcal{E}^+$, due to claim 5.12 above. Also note that the union of all these subtrees is contained in $X_S[Z(0)]^{>k}$. It follows that (Lemma 2.12)

$$S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k}) \le p_{\max}(Z(k) \mid \mathcal{E}^+) \times I(X_S[Z(0)]^{>k} \mid \mathcal{E}^+). \tag{10}$$

First, we bound the second term. We show that the expected amount of information we gain in the new chunk of communication on the input of the server (below the cutoff level ℓ) is bounded by the $\approx 0.2n \log n$ bits that were communicated in the new chunk.

Claim 5.15. *Given any* $(\pi, \mathsf{path}(\ell)) \in E_{(\pi,\mathsf{path}(\ell),\ell)}$, *for any k it holds that*

$$\mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \left[I\left(X_S[Z(0)]^{>k} \,\middle|\, \mathcal{E}^+ \right) \right] \le n \log n.$$

Proof. Note that $X_S[Z(0)] = (X_S[Z(0)]^{\le k}, X_S[Z(0)]^{>k})$. The claim follows using Lemma 2.4(3),

$$\begin{aligned}
&\mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\le k} \mid \pi, \mathsf{path}(\ell), E_{silence}} I\left(X_S[Z(0)]^{>k} \,\middle|\, X_S[Z(0)]^{\le k} = x_S[\mathsf{path}(\ell)]^{\le k}, \mathcal{E} \right) \\
&= \mathbb{E}_{\pi^{new} \mid \pi, \mathsf{path}(\ell), E_{silence}} I\left(X_S[Z(0)]^{>k} \,\middle|\, X_S[Z(0)]^{\le k}, \mathcal{E} \right) \\
&\le \mathbb{E}_{\pi^{new} \mid \pi, \mathsf{path}(\ell), E_{silence}} I\left(X_S[Z(0)] \mid \mathcal{E} \right),
\end{aligned}$$

where the equality comes from performing the expectation over $x_S[\mathsf{path}(\ell)]^{\le k}$, and the transition is via Lemma 2.4(3), and recalling that $X_S[Z(0)] = (X_S[Z(0)]^{\le k}, X_S[Z(0)]^{>k})$. Substituting $\mathcal{E}$ back for better clarity, via Definition 2.2 we get

$$\begin{aligned}
&= \mathbb{E}_{\pi^{new} \mid \pi, \mathsf{path}(\ell), E_{silence}} I\left(X_S[Z(0)] \mid \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \Pi = \pi, \Pi^{new} = \pi^{new} \right) \\
&= I\left(X_S[Z(0)] \,\middle|\, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \Pi = \pi, \widetilde{\Pi}^{new} \right) \\
&\le I\left(X_S[Z(0)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell) \right) + \log |\Omega_{\widetilde{\Pi}^{new}}| \\
&\le 2^{-0.1\sqrt{n}} + 0.2n \log n \\
&\le n \log n,
\end{aligned}$$

where $\widetilde{\Pi}^{new}$ is distributed like Π^{new} conditioned on $(E_{silence}, \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell))$. The penultimate transition holds since $\mathsf{cutoff}(\pi, \mathsf{path}(\ell)) = \ell$, thus without $\widetilde{\Pi}^{new}$ the information is bounded by $2^{-0.1\sqrt{n}} \le 1$. Furthermore, Π^{new} contains only $0.2n \log n$ bits, some of which may be erased, but this gives no extra information on X_S [in fact, half of these bits are sent by the clients and those are (conditionally) independent of X_S and give no further information, but we can count them as well]. Therefore, conditioning on $\widetilde{\Pi}^{new}$ can increase the information by at most $0.2n \log n$ in expectation due to Lemma 2.4(2). □

Since Claim 5.15 bounds the second part of Equation (10) only in expectation, we cannot bound directly the expectation of the product without showing that these two parts are independent.

To this end, we bound the first term directly (not in expectation), and show that the bound is independent of the expectation variables.

Bounding $p_{\max}(Z(k) \mid \mathcal{E}^+)$ is based on the technical Lemma 2.14. We use the fact that the correct path $Z(k)$ in the server's tree is determined by the labels on the correct paths in the clients' trees. Since the amount of information on these labels (beyond the cutoff point) is small, Lemma 2.14 asserts that the probability to guess $Z(k)$ is also small.

First, note that in the derivation below we consider only π^{new} for which $E_{silence}$ occurs; other transcripts never appear in the expectation of the lemma's statement. Also recall we are guaranteed that $(\pi, \mathsf{path}(\ell)) \in E_{(\pi, \mathsf{path}(\ell), \ell)}$. For any specific k, we can think of $Z(k)$ as composed of n binary variables where each represents the path induced by a different client, $Z(k) \overset{\text{def}}{=} (Z_1(k), \ldots, Z_n(k))$. Let $a_1(k), a_2(k), \ldots, a_n(k)$ be n paths of length k that attain the maximal probability, that is, paths that satisfy

$$\Pr[Z_1(k) = a_1(k), Z_2(k) = a_2(k), \ldots, Z_n(k) = a_n(k) \mid \mathcal{E}^+] = p_{\max}(Z(k) \mid \mathcal{E}^+). \tag{11}$$

Note that $Z_1(k), \ldots, Z_n(k)$ and $X_S[Z(0)]^{\leq k}$ induce paths $P_1(k), \ldots, P_n(k)$ on $X_1, \ldots, X_n$, respectively. Each P_i starts at the end of $\mathsf{path}_i(\ell)$ and is of length k. That path is uniquely determined by the i-th bit of the labels along $Z(k)$ in $X_S[Z(0)]$. Then, Equation (11) equals

$$p_{\max}(Z(k) \mid \mathcal{E}^+) = \Pr[label(P_1(k)) = a_1(k), \ldots, label(P_n(k)) = a_n(k) \mid \mathcal{E}^+].$$

Via Corollary 3.2, the labels of P_i are independent of labels of P_j for $j \neq i$, conditioned on $\mathcal{E}^+$ (because these labels are just part of the variables X_i), and the above equals

$$\begin{aligned} p_{\max}(Z(k) \mid \mathcal{E}^+) &= \prod_{i \in [n]} \Pr[label(P_i(k)) = a_i(k) \mid \mathcal{E}^+] \\ &\leq \prod_{i \in Q} \Pr[label(P_i(k)) = a_i(k) \mid \mathcal{E}^+] \\ &\leq \prod_{i \in Q} \max_{P'_i(k)} \Pr[label(P'_i(k)) = a_i(k) \mid \mathcal{E}^+], \end{aligned}$$

where Q is the set of all critical clients (for $\mathsf{cutoff}(\pi, \mathsf{path}(\ell)) = \ell$) *whose communication was completely erased*, and $P'_i(k)$ is any path of length $k + \ell$ in X_i, whose prefix is $\mathsf{path}(\ell)$. That is, instead of looking at a *specific* path P_i, we are looking at all the possible paths, and take the one that maximizes the probability.

Since the communication of any party $i \in Q$ is fully erased in π^{new}, the probability of $label(P_i(k))$ is independent of π^{new}.[8] Also note that once we consider the path $P_i(k)$ that maximizes the probability (out of all possible paths), then the specific path we take no longer matters. Then, the above probability is just the probability that some label pattern occurs in X_i (between levels ℓ and $k + \ell$), and this probability is (conditionally) independent of X_S by Corollary 3.2. Continuing with the above, explicitly writing the elements of $\mathcal{E}^+$ and removing the conditioning on $X_S[Z(0)]^{\leq k}$ (which are just parts of X_S) and the conditioning on π^{new} as explained above, we obtain

$$p_{\max}(Z(k) \mid \mathcal{E}^+) \leq \prod_{i \in Q} \max_{P'_i(k)} \Pr[label(P'_i(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)]. \tag{12}$$

[8] We can assume that both the incoming and outgoing communication of p_i is erased. However, in fact, a stronger claim holds even if we only assume the outgoing communication is erased. The incoming bits are sent by the server and, conditioned on π, are independent of X_i; see also Claim 5.20.

We observe that we can use the bound in Equation (12) not only for a specific k, but even for their sum for $k \geq 30$. This observation will be useful shortly. Formally,

$$\begin{aligned}
&\sum_{k=30}^{T-\ell} \prod_{i\in Q} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}_i(\ell)] \\
&\leq \prod_{i\in Q} \sum_{k=30}^{T-\ell} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}_i(\ell)],
\end{aligned}$$

since exchanging the order of summation and product just adds positive terms. Then we can use Lemma 2.14 to bound the summation,

$$\leq \prod_{i\in Q} \left(2I_i + 4\sqrt{I_i} + 20 \cdot 2^{-30/3}\right),$$

where here $I_i = I(X_i[\mathsf{path}_i(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell))$. Since each party $i \in Q$ is critical we know by Definition 5.7 that $\forall i \in Q, I_i \leq 0.02$, and since $|Q| \geq \sqrt{n}$ when $E_{silence}$ occurs (Lemma 5.8), we conclude that

$$\begin{aligned}
&\sum_{k=30}^{T-\ell} \prod_{i\in Q} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}_i(\ell)] \\
&\leq \prod_{i\in Q} \left(2 \cdot 0.02 + 4\sqrt{0.02} + 20 \cdot 2^{-30/3}\right) \\
&\leq 2^{-0.5\sqrt{n}}. \qquad (13)
\end{aligned}$$

Putting all the ingredients together, we now bound the expectation of $\sum_{k\geq 30} S(k \mid \pi^{new}, x_S[\mathsf{path}_S(\ell)]^{\leq k})$ over all the possible new transcripts and fixings of $x_S[\mathsf{path}_S(\ell)]^{\leq k}$ that occur with positive probability conditioned on $E_{silence}$ and $(\pi, \mathsf{path}(\ell)) \in E_{(\pi,\mathsf{path}(\ell),\ell)}$, and complete the proof of this lemma. Starting with Equation (10),

$$\begin{aligned}
&\sum_{k=30}^{T-\ell} \mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \left[S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k})\right] \\
&\leq \sum_{k=30}^{T-\ell} \mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \left[p_{\max}(Z(k) \mid \mathcal{E}^+) \times I(X_S[Z(0)]^{>k} \mid \mathcal{E}^+)\right],
\end{aligned}$$

now we can bound $p_{\max}(Z(k) \mid \mathcal{E}^+)$ using Equation (12) [note that the expectation is only on transcripts and inputs in $E_{silence}, E_{(\pi,\mathsf{path}(\ell),\ell)}$ as assumed in the derivation of Equation (12)]:

$$\begin{aligned}
\leq \sum_{k=30}^{T-\ell} \mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \Bigg[&\prod_{i\in Q} \max_{P_i'} \Pr[label(P_i') = a_i \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \\
&\times I(X_S[Z(0)]^{>k} \mid \mathcal{E}^+) \Bigg].
\end{aligned}$$

Now, the first term of the product is constant with respect to the expectation,

$$\leq \sum_{k=30}^{T-\ell} \prod_{i \in Q} \max_{P'_i} \Pr[label(P'_i) = a_i \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)]$$
$$\times \mathbb{E}_{\pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k} \mid \pi, \mathsf{path}(\ell), E_{silence}} \left[I(X_S[Z(0)]^{>k} \mid \mathcal{E}^+) \right]$$
$$\leq 2^{-0.5\sqrt{n}} \times n \log n,$$

where the last step is due to Equation (13) and Claim 5.15. □

5.1.2 Bounding the Information in Equation (9). Similarly to the information about the server's X_S, we need to bound the information about the clients' X_i's to satisfy Equation (9), but note that here we only consider π and not π^{new} (thus, there is no need to condition on $E_{silence}$). Still, the information measure in Equation (9) may have increased due to the fact we condition on $\mathsf{path}(k+\ell)$ instead of $\mathsf{path}(\ell)$. We now show that this cannot lead to increasing the server's cutoff level by more than a constant.

We will abuse notations in this second part and redefine $Z_1(k), \ldots, Z_n(k)$ to be the correct paths in $X_1, \ldots, X_n$ of length $k+\ell$, that is, we let $Z_i(k) = \mathsf{PATH}_i(k+\ell)$. Given any $(\pi, \mathsf{path}(\ell)) \in E_{(\pi, \mathsf{path}(\ell), \ell)}$ we define

$$C_i^*(k \mid \mathsf{path}(k+\ell)) \stackrel{\text{def}}{=} I\left(X_i[\mathsf{path}_i(k+\ell)] \,\middle|\, \Pi = \pi, \mathsf{PATH}(k+\ell) = \mathsf{path}(k+\ell)\right),$$
$$C^*(k \mid \mathsf{path}(k+\ell)) \stackrel{\text{def}}{=} \sum_{i=1}^{n} C_i^*(k \mid \mathsf{path}(k+\ell)),$$

which is indeed the measure we need to bound to satisfy Equation (9). As above, we will bound $C^*(k)$ via $C(k)$. Redefine the event $\mathcal{E}$ as

$$\mathcal{E} \stackrel{\text{def}}{=} (\Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)),$$

and let

$$C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\leq k}) \stackrel{\text{def}}{=} \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E}}$$
$$I(X_i[\rho_i] \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathsf{PATH}(k+\ell) = \rho, \mathcal{E}),$$
$$C(k \mid x_{[n]}[\mathsf{path}(\ell)]^{\leq k}) \stackrel{\text{def}}{=} \sum_{i=1}^{n} C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\leq k}).$$

Indeed, the quantity $C(k)$ gives an upper bound on $C^*(k)$, in expectation on the fixing of the k levels of the clients beyond the cutoff level. Formally,

Claim 5.16. *Given any $\pi, \mathsf{path}(\ell)$, and for any k, and any $i \in [n]$,*

$$\mathbb{E}_{\mathsf{path}(k+\ell) \mid \mathcal{E}} C_i^*(k \mid \mathsf{path}(k+\ell)) \leq \mathbb{E}_{x_i[\mathsf{path}_i(\ell)]^{\leq k} \mid \mathcal{E}} C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\leq k}).$$

Proof. The proof is very similar to the proof of Claim 5.13.

$$\begin{aligned}
&\mathbb{E}_{x_i[\mathsf{path}(\ell)]^{\le k}|\mathcal{E}} C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\le k})\\
&= \mathbb{E}_{x_i[\mathsf{path}(\ell)]^{\le k}|\mathcal{E}}\\
&\qquad \mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|x_i[\mathsf{path}_i(\ell)]^{\le k},\mathcal{E}}\\
&\qquad\quad I(X_i[\rho_i] \mid X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathsf{PATH}(k+\ell)=\rho,\mathcal{E})\\
&= \mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|\mathcal{E}}\\
&\qquad \mathbb{E}_{x_i[\mathsf{path}(\ell)]^{\le k}|\rho,\mathcal{E}}\\
&\qquad\quad I(X_i[\rho_i] \mid X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathsf{PATH}(k+\ell)=\rho,\mathcal{E})\\
&= \mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|\mathcal{E}} I(X_i[\rho_i] \mid X_i[\mathsf{path}_i(\ell)]^{\le k}, \mathsf{PATH}(k+\ell)=\rho,\mathcal{E}).
\end{aligned}$$

Using Lemma 2.4(1) we get

$$\begin{aligned}
&\ge \mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|\mathcal{E}} I(X_i[\rho_i] \mid \mathsf{PATH}(k+\ell)=\rho,\mathcal{E})\\
&= \mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|\mathcal{E}} C_i^*(k \mid \rho).
\end{aligned}$$

□

Next, we bound the sum of expectations of $C(k)$ for $k \ge 10$.

Lemma 5.17. *Given any* $(\pi, \mathsf{path}(\ell)) \in E_{(\pi,\mathsf{path}(\ell),\ell)}$,

$$\sum_{k=10}^{T-\ell} \mathbb{E}_{x_{[n]}[\mathsf{path}(\ell)]^{\le k}|\mathcal{E}} \left[C(k \mid x_{[n]}[\mathsf{path}(\ell)]^{\le k}) \right] < 0.08n.$$

Proof. The proof follows the same steps of Lemma 5.14, but the scenario here is somewhat simpler. We use Lemma 2.12 on each C_i: again the variables $\{X_i\}$ of Lemma 2.12 are set to be all various subtrees $X_i[Z_i(k)]$ obtained by all the possible different $Z_i(k)$ that are consistent with $\mathcal{E}$. Again note that, similar to the reasoning in Claim 5.12, the path $Z_i(k)$ is independent of the labels in the subtrees of X_i rooted at the end of a path of length $k+\ell$ with prefix $\mathsf{path}_i(\ell)$, conditioned on $\mathcal{E}$ and on $X_i[\mathsf{path}_i(\ell)]^{\le k}$; this independence is required for applying Lemma 2.12. Also note that the union of all these subtrees is exactly $X_i[Z(0)]^{>k}$. Lemma 2.12 then implies that

$$\begin{aligned}
C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\le k}) \le p_{\max}&\left(Z_i(k) \,\middle|\, X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathcal{E}\right)\\
&\times I\left(X_i[\mathsf{path}_i(\ell)]^{>k} \,\middle|\, X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathcal{E}\right). \quad (14)
\end{aligned}$$

We begin with bounding the term $p_{\max}(Z_i(k) \mid X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathcal{E})$. For any specific k, assume a path $\vec{a}_i(k)$ of length k that maximizes this probability,

$$\Pr[Z_i(k) = \vec{a}_i(k) \mid X_i[\mathsf{path}_i(\ell)]^{\le k} = x_i[\mathsf{path}_i(\ell)]^{\le k}, \mathcal{E}].$$

Once fixing $\vec{a}_i(k)$, it is implied that there exists a path $P(k)$ of length k in X_S (starting from level ℓ, as a continuation of $\mathsf{path}_S(\ell)$ which is fixed given $\mathsf{path}(\ell)$),[9] whose labels, restricted to the i-th

[9] We note that the paths in X_S and the corresponding labels in $X_{[n]}$ are shifted by one level in depth, which is due to the alternating nature of the protocol and our arbitrary decision to let the clients start (assuming all of them take the left son of their root node). To ease the readability of the proof, we will neglect this edge issue and omit the ±1 shift in the indices.

bit, is exactly $\vec{a}_i(k)$. The probability to have a path with such labels is bounded by

$$\begin{aligned}&\leq \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E}]\\ &= \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}],\end{aligned}$$

where the last step follows from Corollary 3.2 that guarantees us the independence of the labels (of ${X_S}^{>\ell}$) from all the other inputs ${X_i}^{>\ell}$, even when conditioning on the transcript so far π, and on $\mathcal{E}$.

We can then bound the sum of the guessing probability of $Z_i(k)$ for $k \geq 10$:

$$\begin{aligned}&\sum_{k=10}^{T-\ell} p_{\max}(Z_i(k) \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E})\\ &\leq \sum_{k=10}^{T-\ell} \max_P \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}]\\ &\leq 2I + 4\sqrt{I} + 20 \cdot 2^{-10/3}\\ &\leq 8,\end{aligned} \tag{15}$$

where the penultimate transition is via Lemma 2.14 by letting T of the lemma be all the labels of $X_S[\mathsf{path}_S(\ell)]^{>\ell}$, setting $I = I(X_S[\mathsf{path}_S(\ell)]^{>\ell} \mid \mathcal{E})$, and recalling that ℓ is the cutoff level (given $(\pi, \mathsf{path}(\ell)) \in E_{(\pi,\mathsf{path}(\ell),\ell)}$), which in turn implies by its definition that $I \leq 2^{-0.1\sqrt{n}} \leq 1$.

Now that the first term of Equation (14) is bounded by a fixed number, bounding the expectation of the $C(k)$ reduces to bounding the expectation of the second term in Equation (14).

$$\begin{aligned}&\sum_{k=10}^{T-\ell} \mathbb{E}_{x[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}}\left[C(k \mid x[\mathsf{path}(\ell)]^{\leq k})\right]\\ &= \sum_{k=10}^{T-\ell} \mathbb{E}_{x[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}}\left[\sum_{i=1}^{n} C_i(k \mid x_i[\mathsf{path}_i(\ell)]^{\leq k})\right]\\ &\leq \sum_{k=10}^{T-\ell} \mathbb{E}_{x[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}}\Bigg[\sum_{i=1}^{n} p_{\max}\left(Z_i(k) \,\middle|\, X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E}\right)\\ &\qquad\qquad \times I\left(X_i[\mathsf{path}_i(\ell)]^{>k} \,\middle|\, X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E}\right)\Bigg]\\ &\leq \sum_{k=10}^{T-\ell} \mathbb{E}_{x[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}}\Bigg[\sum_{i=1}^{n} \max_P \Pr[label_i(P) = \vec{a}_i(k) \mid \mathcal{E}]\\ &\qquad\qquad \times I(X_i[\mathsf{path}_i(\ell)]^{>k} \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E})\Bigg]\\ &\leq \sum_{k=10}^{T-\ell} \sum_{i=1}^{n} \max_P \Pr[label_i(P) = \vec{a}_i(k) \mid \mathcal{E}]\\ &\qquad\qquad \times \mathbb{E}_{x[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}}\left[I(X_i[\mathsf{path}_i(\ell)]^{>k} \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E})\right].\end{aligned}$$

Now, by the definition of information, and using Lemma 2.4(3),

$$= \sum_{k=10}^{T-\ell} \sum_{i=1}^{n} \max_P \Pr[label_i(P) = \vec{a}_i(k) \mid \mathcal{E}] \times I(X_i[\mathrm{path}_i(\ell)]^{>k} \mid X_i[\mathrm{path}_i(\ell)]^{\leq k}, \mathcal{E})$$
$$\leq \sum_{i=1}^{n} \sum_{k=10}^{T-\ell} \max_P \Pr[label_i(P) = \vec{a}_i(k) \mid \mathcal{E}] \times I(X_i[\mathrm{path}_i(\ell)] \mid \mathcal{E}).$$

Using Equation (15),

$$\leq 8 \sum_{i=1}^{n} I(X_i[\mathrm{path}_i(\ell)] \mid \mathcal{E}),$$

recall that ℓ is the cutoff level, i.e., that $(\pi, \mathrm{path}(\ell)) \in E_{(\pi, \mathrm{path}(\ell), \ell)}$,

$$\leq 8 \cdot 0.01n$$
$$\leq 0.08n. \qquad \square$$

5.1.3 Completing the Proof of Proposition 5.9. With the above bounds on the information revealed as a function of the increase k in the new server's cutoff level, we use Lemma 2.13 to bound the expected increase in cutoff_S.

Proof (Proposition 5.9). Given $(\pi, \mathrm{path}(\ell)) \in E_{(\pi, \mathrm{path}(\ell), \ell)}$ consider the following two series of non-negative random variables:

$$\left\{ \tilde{S}(k) \stackrel{\text{def}}{=} \mathbb{E}_{\pi^{new}, \mathrm{path}(k+30+\ell) \mid \pi, \mathrm{path}(\ell), E_{silence}} [S^*(k+30 \mid \pi^{new}, \mathrm{path}(k+30+\ell))] \right\}_{k \geq 0}, \text{ and}$$
$$\left\{ \tilde{C}(k) \stackrel{\text{def}}{=} \mathbb{E}_{\mathrm{path}(k+30+\ell) \mid \pi, \mathrm{path}(\ell), E_{silence}} \left[\sum_{i=1}^{n} C_i^*(k+30 \mid \mathrm{path}(k+30+\ell)) \right] \right\}_{k \geq 0}.$$

We note again that $C(k)$ (as defined in Section 5.1.2) does not assume the event $E_{silence}$, while the above $\tilde{C}(k)$ does. This has no effect on the bounds derived in Section 5.1.2, as this event is completely independent of $C(k)$: the information in $C()$ is conditioned only on π and not on π^{new}, while the event $E_{silence}$ relates only to π^{new} and is independent of any previous communication.

Lemma 5.14 and Claim 5.13 tell us that $\sum_k \tilde{S}(k) \leq n \log n \cdot 2^{-0.5\sqrt{n}}$, and similarly Lemma 5.17 and Claim 5.16 certify that $\sum_k \tilde{C}(k) \leq 0.08n$. Therefore, from Lemma 2.13 it follows that the expectation of the minimal k^* for which $\tilde{S}(k^*) < 2^{-0.2\sqrt{n}}$ as well as $\tilde{C}(k^*) < 0.01n$ is bounded by

$$\mathbb{E}[k^*] \leq 1 + \frac{n \log n \cdot 2^{-0.5\sqrt{n}}}{2^{-0.2\sqrt{n}}} + \frac{0.08n}{0.01n} \leq 10.$$

We recall that the server's cutoff is the minimal round k in which *both* the information described by $S^*(k)$ is below $2^{-0.2\sqrt{n}}$ and $C^*(k)$ is below $0.01n$. From the above, it is then immediate that, given any $(\pi, \mathrm{path}(\ell)) \in E_{(\pi, \mathrm{path}(\ell), \ell)}$, we can bound the expected increase in the server's cutoff by

$$\begin{aligned} \mathbb{E}\left[\mathrm{cutoff}_S(\pi, \Pi^{new}, X) \mid \pi, \mathrm{path}(\ell), E_{silence}\right] &= \mathbb{E}_{\pi^{new}, x \mid \pi, \mathrm{path}(\ell), E_{silence}} \left[\mathrm{cutoff}_S(\pi, \pi^{new}, x)\right] \\ &\leq \ell + 30 + 10 \\ &= \ell + 40, \end{aligned}$$

thus,

$$\mathbb{E}\left[\mathrm{cutoff}_S(\pi, \Pi^{new}, X) \mid E_{(\pi, \mathrm{path}(\ell), \ell)}, E_{silence}\right] \leq \ell + 40,$$

as claimed. $\square$

5.2 Bounding the Cutoff: Proof of Proposition 5.10

Next, we show that, given that the server's cutoff did not increase by much after observing π^{new}, the protocol's cutoff (when considering π^{new} for both the server *and* the clients) cannot increase by more than $O(\log\log n)$ beyond the server's cutoff.

Proof (Proposition 5.10). Let us first recall the setting. We are given $\ell' \leq \ell \leq T$, and π, $\mathsf{path}(\ell)$, π_S^{new}, so that the following holds. The cutoff assuming the old transcript is ℓ', that is, $(\pi, \mathsf{path}(\ell')) \in E_{(\pi,\mathsf{path}(\ell'),\ell')}$. The server's cutoff given π, π_S^{new} is ℓ, that is, $(\pi, \pi_S^{new}, \mathsf{path}(\ell)) \in E^S_{(\pi,\pi_S^{new},\mathsf{path}(\ell),\ell)}$. Additionally, we assume that the event $E_{silence}$ occurs in the new segment of communication, i.e., we only care about $\pi_{[n]}^{new}$ that have positive probability given $E_{silence}$ and the fixed transcript and path given above. We want to show that the new cutoff (i.e., given the new transcript), is at most $\ell + O(\log\log n)$ in expectation over the inputs and $\pi_{[n]}^{new}$.

The proof resembles the proof of Proposition 5.9: we bound the information on the respective subtrees of X_S and $X_{[n]}$ using Lemma 2.12 and Lemma 2.14, and then bound the expected depth of the new subtrees whose information is below the threshold [i.e., satisfying Equations (6)–(7)] via Lemma 2.13.

Recall we can split $\pi^{new} = (\pi_S^{new}, \pi_{[n]}^{new})$ into the parts sent by the server and the clients, respectively. Throughout the proof we will be using the short notations

$$\mathcal{E} = (\Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)),$$
$$\mathcal{E}^S = (\Pi = \pi, \Pi_S^{new} = \pi_S^{new}, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)).$$

For $i \in [n]$ define $Z_i(k) = \mathsf{PATH}_i(k+\ell)$. Given any $\pi, \pi_S^{new}, \mathsf{path}(\ell)$ we define the random functions

$$C_i^*(k \mid \pi_{[n]}^{new}, \mathsf{path}(k+\ell)) \overset{\text{def}}{=} I\left(X_i[\mathsf{path}_i(k+\ell)] \mid \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \mathsf{path}(k+\ell)\right)$$

and,

$$C_i(k \mid \pi_{[n]}^{new}, x_i[\mathsf{path}_i(\ell)]^{\leq k}) \overset{\text{def}}{=} \mathbb{E}_{\rho \sim \mathsf{PATH}(k+\ell) \mid x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathcal{E}} I(X_i[\rho_i] \mid X_i[\mathsf{path}_i(\ell)]^{\leq k} = x_i[\mathsf{path}_i(\ell)]^{\leq k}, \mathsf{PATH}(k+\ell) = \rho, \mathcal{E}),$$

$$C(k \mid \pi_{[n]}^{new}, x_{[n]}[\mathsf{path}(\ell)]^{\leq k}) \overset{\text{def}}{=} \sum_{i=1}^{n} C_i(k \mid \pi_{[n]}^{new}, x_i[\mathsf{path}_i(\ell)]^{\leq k}).$$

A reminder that $\sum_i C_i^*(k)$ is indeed the quantity we wish to bound [to satisfy Equation (7)], and that for any $\pi_{[n]}^{new}$, the measure $C(k)$ upper bounds $\sum_i C_i^*(k)$ in expectation, via Claim 5.16 (note that Claim 5.16 can be used as is, by considering the entire transcript $\pi \circ \pi^{new}$ as the transcript we condition on, in that claim).

Lemma 5.18. *Given any* $(\pi, \pi_S^{new}, \mathsf{path}(\ell)) \in E^S_{(\pi,\pi_S^{new},\mathsf{path}(\ell),\ell)}$,

$$\sum_{k=3\log\log n}^{T-\ell} \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\mathsf{path}(\ell)]^{\leq k} \mid \mathcal{E}^S}\left[C\left(k \mid \pi_{[n]}^{new}, x_{[n]}[\mathsf{path}(\ell)]^{\leq k}\right)\right] < 21n.$$

Proof. The first part of the proof follows the same reasoning and notational conventions used in the proof of Proposition 5.9 (or specifically, Lemma 5.17), and we don't repeat here the detailed arguments leading to the following derivation:

$$\begin{aligned}
&\sum_{k=3\log\log n}^{T-\ell} \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k} | \mathcal{E}^S} \left[C(k \mid \pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k}) \right] \\
&= \sum_{k=3\log\log n}^{T-\ell} \sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k} | \mathcal{E}^S} \left[C_i(k \mid \pi_{[n]}^{new}, x_i[\text{path}_i(\ell)]^{\le k}) \right] \\
&\le \sum_{k=3\log\log n}^{T-\ell} \sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k} | \mathcal{E}^S} \Big[p_{\max}(Z_i(k) \mid X_i[\text{path}_i(\ell)]^{\le k} = x_i[\text{path}_i(\ell)]^{\le k}, \mathcal{E}) \\
&\qquad\qquad \times I(X_i[\text{path}_i(\ell)]^{> k} \mid X_i[\text{path}_i(\ell)]^{\le k} = x_i[\text{path}_i(\ell)]^{\le k}, \mathcal{E}) \Big] \\
&\le \sum_{k=3\log\log n}^{T-\ell} \sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k} | \mathcal{E}^S} \Bigg[\max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \\
&\qquad\qquad \times I(X_i[\text{path}_i(\ell)]^{> k} \mid X_i[\text{path}_i(\ell)]^{\le k} = x_i[\text{path}_i(\ell)]^{\le k}, \mathcal{E}) \Bigg] \\
&\le \sum_{i=1}^{n} \sum_{k=3\log\log n}^{T-\ell} \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \\
&\qquad \times \mathbb{E}_{\pi_{[n]}^{new}, x_{[n]}[\text{path}(\ell)]^{\le k} | \mathcal{E}^S} \left[I(X_i[\text{path}_i(\ell)]^{> k} \,\middle|\, X_i[\text{path}_i(\ell)]^{\le k} = x_i[\text{path}_i(\ell)]^{\le k}, \mathcal{E}) \right] \\
&= \sum_{i=1}^{n} \sum_{k=3\log\log n}^{T-\ell} \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \\
&\qquad\qquad \times \mathbb{E}_{\pi_{[n]}^{new} | \mathcal{E}^S} \left[I(X_i[\text{path}_i(\ell)]^{> k} \mid X_i[\text{path}_i(\ell)]^{\le k}, \mathcal{E}) \right],
\end{aligned}$$

which by Lemma 2.4(3) gives

$$= \sum_{i=1}^{n} \sum_{k=3\log\log n}^{T-\ell} \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \times \mathbb{E}_{\pi_{[n]}^{new} | \mathcal{E}^S} \left[I(X_i[\text{path}_i(\ell)] \mid \mathcal{E}) \right]. \tag{16}$$

We now bound the two multiplicands of Equation (16) separately.

Claim 5.19. *For any $i \in [n]$,*

$$\sum_{k=3\log\log n}^{T-\ell} \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \le 6 \cdot 2^{-0.1\sqrt{n}} + \frac{20}{\log n}.$$

Proof. Recall that $P(k)$ describes a path of length k in $X_S[\text{path}(\ell)]$. The maximal probability guess of the labels of $P(k)$ (restricted to the i-th bit) for $k \ge 3\log\log n$ is given by Lemma 2.14, setting the variable T of the lemma as $T = X_S[\text{path}(\ell)]$ (restricted to the i-th bit in each label), and using the fact that $\text{cutoff}_S(\pi, \pi^{new}, \text{path}(\ell)) = \ell$, so that $I(T) \le I(X_S[\text{path}(\ell)] \mid \mathcal{E}) \le 2^{-0.2\sqrt{n}}$.

Thus,

$$\sum_{k=3\log\log n}^{T-\ell} \max_{P(k)} \Pr[label_i(P(k)) = \vec{a}_i(k) \mid \mathcal{E}] \leq 2I(T) + 4\sqrt{I(T)} + 20 \cdot 2^{-\log\log n}$$
$$\leq 6 \cdot 2^{-0.1\sqrt{n}} + \frac{20}{\log n}. \qquad \square$$

Before we bound the second multiplicand of Equation (16), we prove the following technical claim.

Claim 5.20. *Let $\Pi = \pi$ be an observed transcript up to some point, and let Π^{new} be a continuation of Π. Write $\Pi^{new} = (\Pi_S^{new}, \Pi_{[n]}^{new})$ splitting the observed transcript to the corresponding indices sent by the server and by the clients, respectively. Then, $X_{[n]}$ is independent of Π_S^{new} conditioned on $(\Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})$.*

Proof. First, we assume there are no erasures in Π_S^{new}. Consider the string Π_S^{new}: each bit in it is a function of X_S and the communication the server sees, that is, $\Pi_S^{new} = f(X_S, \Pi_{[n]}, \Pi_{[n]}^{new})$. It is clear that if we fix and condition on a specific $(\Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})$, then $\Pi_S^{new} = g(X_S)$ where the function g is determined solely by $\pi, \pi_{[n]}^{new}$.

$$\begin{aligned}
&\Pr(X_{[n]} = x_{[n]}, \Pi_S^{new} = \pi_S^{new} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})\\
&= \Pr(X_{[n]} = x_{[n]}, f(X_S, \Pi_{[n]}, \Pi_{[n]}^{new}) = \pi_S^{new} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})\\
&= \Pr(X_{[n]} = x_{[n]}, g(X_S) = \pi_S^{new} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})
\end{aligned}$$

by Lemma 3.1, X_S and $X_{[n]}$ are independent, conditioned on any (partial) transcript,

$$\begin{aligned}
&= \Pr(X_{[n]} = x_{[n]} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new}) \Pr(g(X_S) = \pi_S^{new} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new})\\
&= \Pr(X_{[n]} = x_{[n]} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new}) \Pr(\Pi_S^{new} = \pi_S^{new} \mid \Pi = \pi, \Pi_{[n]}^{new} = \pi_{[n]}^{new}),
\end{aligned}$$

which completes the proof. The same holds if bits from Π_S^{new} are flipped or erased, since the noise is independent of all the other variables. $\square$

Claim 5.21.

$$\sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new} \mid \mathcal{E}^S} \left[I(X_i[\mathsf{path}_i(\ell)] \mid \mathcal{E}) \right] \leq n \log n.$$

Proof. Writing $\mathcal{E}$ explicitly in the claim's statement, we have

$$\sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new} \mid \mathcal{E}^S} I(X_i[\mathsf{path}_i(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \Pi_S^{new} = \pi_S^{new}, \Pi_{[n]}^{new} = \pi_{[n]}^{new}).$$

For any $i \in [n]$, Claim 5.20 suggests that the event $\Pi_S^{new} = \pi_S^{new}$ is independent of $X_{[n]}$ conditioned on the transcript so far (and the paths, and so on). Therefore, conditioning on $\Pi_S^{new} = \pi_S^{new}$ does not change the (conditional) distribution of $X_{[n]}$, and we can remove the conditioning on $\Pi_S^{new} = \pi_S^{new}$ without affecting the information,

$$= \sum_{i=1}^{n} \mathbb{E}_{\pi_{[n]}^{new} \mid \mathcal{E}^S} I(X_i[\mathsf{path}_i(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \Pi_{[n]}^{new} = \pi_{[n]}^{new}),$$

by linearity of expectation and superadditivity of information (Lemma 2.3),

$$\begin{aligned}
&\leq \mathbb{E}_{\pi_{[n]}^{new} \mid \mathcal{E}^S} I(X_{[n]}[\mathsf{path}(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \Pi_{[n]}^{new} = \pi_{[n]}^{new})\\
&= I(X_{[n]}[\mathsf{path}(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell), \widetilde{\Pi}_{[n]}^{new}),
\end{aligned}$$

where $\widetilde{\Pi}^{new}_{[n]}$ is distributed according to $\Pi^{new}_{[n]}$ conditioned on $\mathcal{E}^S$. Recall that $\Pi^{new}_{[n]}$ contains up to $0.1n\log n$ bits (some may be erased); similarly, $\widetilde{\Pi}^{new}_{[n]}$ also contains at most $0.1n\log n$ bits of information. Using Lemma 2.4(2),

$$\leq I(X_{[n]}[\mathsf{path}(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)) + 0.1n\log n.$$

Now note that, conditioned on $(\Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell))$, the variables $X_1, \ldots, X_n$ are mutually independent by Lemma 3.1, thus the superadditivity (Lemma 2.3) in this case satisfies an equality,

$$= 0.1n\log n + \sum_{i=1}^{n} I(X_i[\mathsf{path}_i(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)).$$

Finally, since ℓ is the server's cutoff given the transcript π (and π^{new}_S), we get

$$\leq 0.1n\log n + 0.01n$$
$$\leq n\log n.$$
□

Substituting the bounds in Claim 5.19 and Claim 5.21 back into Equation (16) completes the proof of Lemma 5.18. □

Now that we have bounded the information on the clients' trees, we need to bound the information on the server's tree as well [to satisfy Equation (6)]. This repeats the same methods we have seen above, but in a slightly relaxed setting: the server is currently at the cutoff level, and the communication $\pi^{new}_{[n]}$ doesn't give any new information on X_S.

We denote by $Z(k) = \mathsf{PATH}_S(k+\ell)$ the correct path of length k in X_S, below the server's cutoff level. Given any $\pi, \pi^{new}_S, \mathsf{path}(\ell)$ define

$$S^*\left(k \;\middle|\; \pi^{new}_{[n]}, \mathsf{path}(k+\ell)\right) \overset{\text{def}}{=}$$
$$I\Big(X_S[\mathsf{path}_S(k+\ell)] \;\Big|\; \Pi = \pi, \Pi^{new} = \pi^{new}, \mathsf{PATH}(k+\ell) = \mathsf{path}(k+\ell)\Big),$$
$$S\left(k \;\middle|\; \pi^{new}_{[n]}, x_S[\mathsf{path}(\ell)]^{\leq k}\right) \overset{\text{def}}{=}$$
$$\mathbb{E}_{\rho\sim\mathsf{PATH}(k+\ell)|x_S[\mathsf{path}(\ell)]^{\leq k},\mathcal{E}} I\left(X_S[\rho_S(k+\ell)] \;\middle|\; X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \mathsf{PATH}(k+\ell) = \rho, \mathcal{E}\right).$$

Note that an immediate corollary of the derivation in Claim 5.13 is the following:

Corollary 5.22. *Given any π, $\mathsf{path}(\ell)$, and $\pi^{new} = (\pi^{new}_S, \pi^{new}_{[n]})$, it holds that*

$$\mathbb{E}_{\mathsf{path}(k+\ell)|\mathcal{E}} S^*\left(k \;\middle|\; \pi^{new}_{[n]}, \mathsf{path}(k+\ell)\right) \leq \mathbb{E}_{x_S[\mathsf{path}(\ell)]^{\leq k}|\mathcal{E}} S\left(k \;\middle|\; \pi^{new}_{[n]}, x_S[\mathsf{path}(\ell)]^{\leq k}\right).$$

We can now continue to bound the sum of expectations of the quantities $S^*(k)$.

Lemma 5.23. *Given any $(\pi, \pi^{new}_S, \mathsf{path}(\ell)) \in E_{(\pi,\pi^{new}_S,\mathsf{path}(\ell),\ell)}$, and any $\pi^{new}_{[n]}$ assuming $E_{silence}$,*

$$\sum_{k=10}^{T-\ell} \mathbb{E}_{\mathsf{path}(k+\ell)|\mathcal{E}}\left[S^*\left(k \;\middle|\; \pi^{new}_{[n]}, \mathsf{path}(k+\ell)\right)\right] \leq n\cdot 2^{-0.2\sqrt{n}}.$$

Proof. The proof follows at large the arguments of Lemma 5.14, and we repeat here the minimal required details.

Lemma 2.12 asserts that

$$S(k \mid \pi^{new}, x_S[\mathsf{path}(\ell)]^{\leq k}) \leq p_{\max}(Z(k) \mid X_S[\mathsf{path}(\ell)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \mathcal{E})$$
$$\times I(X_S[Z(0)]^{>k} \mid X_S[\mathsf{path}(\ell)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \mathcal{E}), \quad (17)$$

where, again, the $\{X_i\}$ of Lemma 2.12 are all the subtrees of X_S rooted at the end of a path of depth $k+\ell$, whose prefix is $\text{path}_S(\ell)$. We note that those subtrees and (the last k edges in each of) $\text{PATH}(k+\ell)$ are independent conditioned on $\mathcal{E}$, due to claim 5.12, and that the union of all these subtrees is contained within $X_S[Z(0)]^{>k}$.

Starting with the term in the Lemma's statement, we use Corollary 5.22 and Equation (17) to get

$$\begin{aligned}
&\sum_{k=10}^{T-\ell} \mathbb{E}_{\text{path}(k+\ell)|\mathcal{E}} \left[S^*(k \mid \pi^{new}_{[n]}, \text{path}(k+\ell)) \right] \\
&\leq \sum_{k=10}^{T-\ell} \mathbb{E}_{x_S[\text{path}(\ell)]^{\leq k}|\mathcal{E}} \left[S(k \mid \pi^{new}_{[n]}, x_S[\text{path}(\ell)]^{\leq k}) \right] \\
&\leq \sum_{k=10}^{T-\ell} \mathbb{E}_{x_S[\text{path}(\ell)]^{\leq k}|\mathcal{E}} \Big[p_{\max}(Z(k) \mid X_S[\text{path}(\ell)]^{\leq k} = x_S[\text{path}(\ell)]^{\leq k}, \mathcal{E}) \\
&\qquad\qquad \times I(X_S[Z(0)]^{>k} \mid X_S[\text{path}(\ell)]^{\leq k} = x_S[\text{path}(\ell)]^{\leq k}, \mathcal{E}) \Big].
\end{aligned} \tag{18}$$

To ease the readability, in the following let us use the shorthand notation

$$\mathcal{E}^+ = \left(X_S[\text{path}(\ell)]^{\leq k} = x_S[\text{path}(\ell)]^{\leq k}, \mathcal{E} \right).$$

Using a similar reasoning to the derivation of Equation (12), we now bound $p_{\max}(Z(k) \mid \mathcal{E}^+)$ as a function of the information we have on labels below the cutoff. Again we think of $Z(k)$ as composed of n binary variables that each depends on a different user, $Z(k) \stackrel{\text{def}}{=} (Z_1(k), \ldots, Z_n(k))$, and let $a_1(k), a_2(k), \ldots, a_n(k)$ be n paths of length k that attain the maximal probability. Recall that $Z_1(k), \ldots, Z_n(k)$ and $X_S[Z(0)]^{\leq k}$ induce paths $P_1(k), \ldots, P_n(k)$ on $X_1, \ldots, X_n$, respectively. Each P_i starts at the end of $\text{path}_i(\ell)$ and is of length k. That path is uniquely determined by the i-th bit of the labels along $Z(k)$ in $X_S[Z(0)]$. Then, we can write

$$p_{\max}(Z(k) \mid \mathcal{E}^+) = \Pr[\mathit{label}(P_1(k)) = a_1(k), \ldots, \mathit{label}(P_n(k)) = a_n(k) \mid \mathcal{E}^+].$$

Via Corollary 3.2, the labels of P_i are independent of labels of P_j for $j \neq i$, conditioned on $\mathcal{E}^+$ (because these labels are just part of the variables X_i), and the above equals

$$\begin{aligned}
p_{\max}(Z(k) \mid \mathcal{E}^+) &= \prod_{i \in [n]} \Pr[\mathit{label}(P_i(k)) = a_i(k) \mid \mathcal{E}^+] \\
&\leq \prod_{i \in Q} \Pr[\mathit{label}(P_i(k)) = a_i(k) \mid \mathcal{E}^+] \\
&\leq \prod_{i \in Q} \max_{P'_i(k)} \Pr[\mathit{label}(P'_i(k)) = a_i(k) \mid \mathcal{E}^+],
\end{aligned}$$

where Q here is the set of all clients that were completely erased in the new part. Since $E_{silence}$ occurs, we know that Q is non-empty, so we can choose a specific party $i \in Q$, assume all other probabilities are 1, and get

$$p_{\max}(Z(k) \mid \mathcal{E}^+) \leq \max_{P'_i(k)} \Pr[\mathit{label}(P'_i(k)) = a_i(k) \mid \mathcal{E}^+].$$

We write $\mathcal{E}^+$ explicitly, and remind that for any party in Q (and specifically for our chosen party i) the part π^{new}_i is completely erased and thus independent of the probability of seeing a specific label in the input. Furthermore, as explained earlier, once we go over all the possible paths P'_i, the probability of $\mathit{label}(P'_i(k))$ is merely the probability to see some labels in X_i in those specific levels, and those are independent of X_S. Also, recall that, given the transcript π and the fact that party i was completely erased, π_S^{new} is a function of only $X_{\neq i}$, which is again (conditionally) independent of the probability to see certain labels in X_i (Corollary 3.2). We get,

$$p_{\max}(Z(k) \mid \mathcal{E}^+) \leq \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)].$$

Continuing with Equation (18),

$$\begin{aligned}
&\leq \sum_{k=10}^{T-\ell} \mathbb{E}_{x_S[\mathsf{path}(\ell)]^{\leq k} \mid \mathcal{E}} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \\
&\qquad\qquad \times I(X_S[Z(0)]^{>k} \mid X_S[\mathsf{path}(\ell)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \mathcal{E}) \\
&= \sum_{k=10}^{T-\ell} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \\
&\qquad \times \mathbb{E}_{x_S[\mathsf{path}(\ell)]^{\leq k} \mid \mathcal{E}} I(X_S[Z(0)]^{>k} \mid X_S[Z(0)]^{\leq k} = x_S[\mathsf{path}(\ell)]^{\leq k}, \mathcal{E}) \\
&= \sum_{k=10}^{T-\ell} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \\
&\qquad \times I(X_S[Z(0)]^{>k} \mid X_S[Z(0)]^{\leq k}, \mathcal{E})
\end{aligned}$$

with Lemma 2.4(3), and recalling that ℓ is the server's cutoff,

$$\begin{aligned}
&\leq \sum_{k=10}^{T-\ell} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \times I(X_S[Z(0)] \mid \mathcal{E}) \\
&\leq \sum_{k=10}^{T-\ell} \max_{P_i'(k)} \Pr[label(P_i'(k)) = a_i(k) \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell)] \times 2^{-0.2\sqrt{n}},
\end{aligned}$$

which, by Lemma 2.14, is bounded by

$$\leq \left(2I_i + 4\sqrt{I_i} + 20 \cdot 2^{-10/3}\right) 2^{-0.2\sqrt{n}}$$

with $I_i \leq I(X_i[\mathsf{path}_i(\ell)] \mid \Pi = \pi, \mathsf{PATH}(\ell) = \mathsf{path}(\ell))$. We know that ℓ is the server cutoff, which implies $\sum_{j \in [n]} I_j < 0.01n$, and, thus, for any party and specifically for our chosen party i, we have $I_i < 0.01n$. Then, for large enough n,

$$\begin{aligned}
&\leq \left(2 \cdot 0.01n + 4\sqrt{0.01n} + 20 \cdot 2^{-10/3}\right) \cdot 2^{-0.2\sqrt{n}} \\
&\leq n \cdot 2^{-0.2\sqrt{n}}.
\end{aligned}$$

□

Finally, we can bound the expected increase of the cutof via Lemma 2.13. Similar to Proposition 5.9, given $(\pi, \pi_S^{new}, \mathsf{path}(\ell)) \in E^S_{(\pi, \pi_S^{new}, \mathsf{path}(\ell), \ell)}$ consider the following two series of non-negative random variables:

$$\left\{\tilde{S}(k) \overset{\text{def}}{=} \mathbb{E}_{\pi_{[n]}^{new}, \mathsf{path}(k+\ell) \mid \pi, \pi_S^{new}, \mathsf{path}(\ell), E_{silence}} \left[S^*(k + 3\log\log n \mid \pi_{[n]}^{new}, \mathsf{path}(k + 3\log\log n + \ell))\right]\right\}_{k \geq 0}$$

and

$$\left\{\tilde{C}(k) \stackrel{\text{def}}{=} \mathbb{E}_{\pi_{[n]}^{new},\text{path}(k+3\log\log n+\ell)|\pi,\pi_S^{new},\text{path}(\ell)}\left[\sum_{i=1}^{n} C_i^*(k+3\log\log n \mid \pi_{[n]}^{new},\text{path}(k+\ell))\right]\right\}_{k\geq 0}.$$

Lemma 5.23 shows that $\sum_k \tilde{S}(k) \leq n\cdot 2^{-0.2\sqrt{n}}$ (it is bounded for any transcript $\pi_{[n]}^{new}$ for which $E_{silence}$ occurs, and thus also in expectation over these transcripts). Lemma 5.18 (along with Claim 5.16) proves that $\sum_k \tilde{C}(k) \leq 21n$. With these bounds, Lemma 2.13 then guarantees that the expectation of the minimal round k^* for which $\tilde{S}(k^*) < 2^{-0.1\sqrt{n}}$ as well as $\tilde{C}(k^*) \leq 0.01n$ is bounded by

$$\mathbb{E}[k^*] \leq 1 + \frac{n\cdot 2^{-0.2\sqrt{n}}}{2^{-0.1\sqrt{n}}} + \frac{21n}{0.01n} \leq 2500.$$

We conclude that, for large enough n,

$$\begin{aligned}
&\mathbb{E}\left[\text{cutoff}(\pi,\Pi^{new},X)\,\middle|\, E^S_{(\pi,\pi_S^{new},\text{path}(\ell),\ell)}, E_{silence}\right]\\
&= \mathbb{E}_{\pi_{[n]}^{new},x|E^S_{(\pi,\pi_S^{new},\text{path}(\ell),\ell)},E_{silence}}\left[\text{cutoff}(\pi,\pi^{new},x)\right]\\
&\leq \ell + 3\log\log n + 2500\\
&\leq \ell + 4\log\log n. \qquad \square
\end{aligned}$$

5.3 Bounding the Cutoff When $\overline{E_{silence}}$ Occurs: Proof of Proposition 5.11

In this section, we take care of the rare event $\overline{E_{silence}}$ where there is no subset of size $\sqrt{n}$ critical players whose communication is completely erased. We claim that during $0.1\log n$ rounds in which $E_{silence}$ did not happen, the cutoff level cannot increase by more than $O(n\log n\log\log n)$.

The idea of this proof is to show that a transmission of a single bit (whether erased or not), could be simulated by τ segments (each of $0.1\log n$ rounds), where we assume that in each one of the segments $E_{silence}$ occurs. That is, given a fixed (noisy) transcript B of $0.1\log n$ rounds in which $E_{silence}$ did not happen, we perform the following thought-experiment in which we are given access to a special channel through which the parties perform $0.1\log n$ rounds of alternating communication, and it is guaranteed that the erasure pattern induced by the channel satisfies $E_{silence}$.

Using multiple utilization of the above special-channel, the parties simulate the transcript $B = B_1\cdots B_{0.1n\log n}$, bit by bit: the first bit, B_1, is simulated by letting the party that sends B_1 in the original protocol input B_1 again and again to the special channel[10]; all the other parties input random bits to the special channel. The above process repeats until B_1 is not erased by the special-channel and is received correctly by the other side. At this point, the parties continue to simulate the second bit of B. Note that this is only a thought experiment, so we can assume an all-knowledgable oracle that tells the parties when the simulation of a given bit succeeds, when to stop, and so on.

It holds that simulating a single bit of B may take τ utilization of the special channel, where τ is a random variable whose expectation is a small constant (and in particular, finite). This means that the information that crossed the special channel during those τ segments (of $0.1\log n$ noisy rounds each) bounds the information communicated by a single bit of B. Moreover, during the first $\tau - 1$ segments, no useful information has passed across the channel. Indeed, these segments merely contain random bits erased by random noise and they have no effect on the cutoff. The last block may increase the cutof; however, we know that $E_{silence}$ occurred in this segment, and

[10] If $B_1 = \perp$, we can assume the party inputs random bits, but if this is the case, then it is clear that B_1 can be simulated using a single segment of simulation. In the following, we assume $B_1 \neq \perp$.

thus the expected increase in the cutoff in this segment is bounded by $O(\log\log n)$ as given by Proposition 5.9 and Proposition 5.10. Therefore, simulating the entire block B bit-by-bit in the above manner can increase the cutoff by at most $O(n\log n\cdot\log\log n)$, in expectation.

Proof (Proposition 5.11). Let $B = B_1\cdots B_{0.1n\log n}$ be the observed transcript Π^{new} of a block of communication with arbitrary erasure noise. For any $i\in[0.1n\log n]$, assume that B_i is communicated in the original protocol by $p_i\in S\cup[n]$.

We simulate each B_i independently via multiple segments of $0.1\log n$ noisy rounds. Assume that all parties except for p_i try to communicate random bits in the simulation, and that p_i tries to communicate B_i. Let τ_i be the minimal number of $0.1\log n$-round segments it takes until B_i is communicated unerased across the channel, conditioned that, in each such segment, the event $E_{silence}$ occurs. Denote the (noisy) transcript of these segments by $(\Pi_i^1,\ldots,\Pi_i^{\tau_i})\overset{\text{def}}{=}\Pi_i$. It is easy to verify that $E[\tau_i]<\infty$, in fact, $\mathbb{E}[\tau_i]$ is bounded by a small constant.

We begin by claiming that $\Pi_i=(\Pi_i^1,\ldots,\Pi_i^{\tau_i})$ contains the information in B_i, and other information that is independent of the inputs. Therefore, conditioned on any previous communication π', the cutoff given the simulation transcript Π_i is equal to the cutoff given B_i.

Claim 5.24. *For any π',*

$$\mathbb{E}[\mathsf{cutoff}(\pi'\circ B_i,X)] = \mathbb{E}[\mathsf{cutoff}(\pi'\circ\Pi_i^1\cdots\Pi_i^{\tau_i},X)].$$

Proof. Via a trivial reordering of indices, we can write $\Pi_i^1\cdots\Pi_i^{\tau_i}$ as $(B_i, RND, NOISE_i^1\cdots NOISE_i^{\tau_i})$, where RND is the string of random bits communicated by all the other parties and $NOISE_i^j$ is the erasure pattern observed in the j-th block of the simulation, Π_i^j; note that we assume that $E_{silence}$ occurs in all such blocks. Moreover, note that for each j, $NOISE_i^j$ have exactly the same distribution: the noise pattern depends only on the identities of critical parties, yet these are fully determined by π' and the inputs up to the cutoff assuming π'. In particular, they are independent of RND and the other noise patterns.

Furthermore, the noise pattern is conditionally independent of the inputs X, given $(\pi', \mathsf{path}(\ell'), E_{silence})$. The event $E_{silence}$ restricts the noise to one that fully corrupts at least $\sqrt{n}$ critical parties. The identity of these parties is only a function of being critical or not. Hence, additionally conditioning on the specific inputs X the parties may hold does not change the distribution.

Using the above argument, X is conditionally independent of all the information in $\Pi_i^1\cdots\Pi_i^{\tau_i}$ except for B_i. Then, for any k, $\mathsf{path}(k)'$ it holds that

$$\begin{aligned}
&I\left(X_S[\mathsf{path}'_S(k)]\,\middle|\,\Pi=\pi',\mathsf{PATH}(k)=\mathsf{path}'(k),\Pi^{new}=(\Pi_i^1,\ldots,\Pi_i^{\tau_i})\right)\\
&=I\left(X_S[\mathsf{path}'_S(k)]\,\middle|\,\Pi=\pi',\mathsf{PATH}(k)=\mathsf{path}'(k),\Pi^{new}=(B_i,RND,NOISE_i^1\cdots NOISE_i^{\tau_i})\right)\\
&=I\left(X_S[\mathsf{path}'_S(k)]\,\middle|\,\Pi=\pi',\mathsf{PATH}(k)=\mathsf{path}'(k),\Pi^{new}=B_i\right).
\end{aligned}$$

A similar argument applies for $\sum_{j\in[n]} I(X_j[\mathsf{path}'_j(k)]\mid\Pi=\pi',\mathsf{PATH}(k)=\mathsf{path}'(k),\Pi^{new}=B_i)$. The claim then follows by the definition of the cutoff. □

Next we wish to bound the cutoff increase due to B_i by bounding the cutoff increase in the simulation. Since each segment in the simulation is one in which $E_{silence}$ happened, we can bound the expected increase in the cutoff via Propositions 5.9 and 5.10.

Proposition 5.25. *Given any $(\pi',\mathsf{path}(\ell'),\ell')\in E_{(\pi',\mathsf{path}(\ell'),\ell')}$, and any B_i,*

$$\mathbb{E}[\mathsf{cutoff}(\pi'\circ\Pi_i^1\cdots\Pi_i^{\tau_i},X)]<\ell'+O(\log\log n).$$

Proof. First we note that all the segments $\Pi_i^1\cdots\Pi_i^{\tau_i-1}$ have no effect on the cutoff. Indeed, as argued above, these transcripts contain only random bits and random noise patterns, which are conditionally independent of X given $E_{(\pi',\mathrm{path}(\ell'),\ell')}$. The only block that increases the cutoff is the last one, Π^{τ_i}.

In the last block, $E_{silence}$ occurs and we can use Propositions 5.9 and 5.10 to bound the progress of the cutoff. We must also condition on the event where p_i is not blocked in this segment. However, the probability for a party not to be blocked in a particular segment in which $E_{silence}$ occurred is rather high. In particular, it is easy to verify that (assuming large enough n)

$$\Pr[p_i \text{ is } \textbf{not} \text{ completely blocked} \mid E_{silence}] > 1/2. \tag{19}$$

Indeed, recalling we assume that the underlying channel is $\mathrm{BEC}_{1/3}$, we have

$$\Pr[p_i \text{ is completely blocked} \mid E_{silence}] \leq \frac{\Pr[p_i \text{ is completely blocked}]}{\Pr[E_{silence}]} \leq \frac{\frac{1}{3}^{0.1\log n}}{1-2^{-\sqrt{n}}} \leq n^{-0.1}.$$

Denote by $\tilde{E}$ the event $(E_{silence}, E_{(\pi',\mathrm{path}(\ell'),\ell')}, p_i$ is not completely blocked in $\Pi_i^{\tau_i})$, then we have

$$\begin{aligned}
&\mathbb{E}[\mathrm{cutoff}(\pi' \circ \Pi_i, X) - \ell' | E'] \\
&= \sum_{c=1}^{\infty} \Pr[\mathrm{cutoff}(\pi' \circ \Pi_i, X) - \ell' \geq c \mid E'] \\
&\leq \sum_{c=1}^{\infty} \frac{\Pr[\mathrm{cutoff}(\pi' \circ \Pi_i, X) - \ell' \geq c \mid E_{silence}, E_{(\pi',\mathrm{path}(\ell'),\ell')}]}{\Pr[p_i \text{ is not completely blocked in } \Pi_i^{\tau_i}]} \\
&\leq \sum_{c=1}^{\infty} 2\Pr[\mathrm{cutoff}(\pi' \circ \Pi_i, X) - \ell' \geq c \mid E_{silence}, E_{(\pi',\mathrm{path}(\ell'),\ell')}] \\
&\leq 2 \cdot O(\log\log n) = O(\log\log n),
\end{aligned}$$

where the last transition follows from Propositions 5.9 and 5.10: Denote the variables of those propositions with hats, and set $\hat{\pi} = \pi' \circ \Pi_i^1 \circ \cdots \circ \Pi_i^{\tau_i - 1}$, the new transcript is the $\widehat{\Pi^{new}} = \Pi_i^{\tau_i}$, and $\hat{\ell} = \ell'$. The penultimate transition follows from Equation (19). □

Proposition 5.25 and Claim 5.24 together prove that for any $(\pi', \mathrm{path}(\ell'), \ell') \in E_{(\pi',\mathrm{path}(\ell'),\ell')}$, and any B_i,

$$\mathbb{E}[\mathrm{cutoff}(\pi' \circ B_i, X)] < \ell' + O(\log\log n).$$

That is, a single bit of communication in the original protocol increases the cutoff in expectation by at most $O(\log\log n)$, regardless of the noise that occurred in B. Using the above repeatedly bit-by-bit over the $0.1n\log n$ bits of $\Pi^{new} = B$, we get that this segment increases the cutoff by at most $O(n\log n\log\log n)$ in expectation, which completes the proof of Proposition 5.11. □

6 CONCLUSION AND OPEN QUESTIONS

In this article, we have shown a lower bound of $\Omega(\log n/\log\log n)$ on the communication of any protocol for the pointer jumping task over a star, assuming each channel is a BSC. This implies that the best interactive coding in this setting has a code rate of at most $O(\log\log n/\log n)$. In particular, the coding of Rajagopalan and Schulman [36] is optimal, up to $\log\log n$ terms. Towards this end, we introduced the “cutoff” of a protocol—a new information-theory notion via which we were able to bound how much progress a coding scheme could have made, in terms of the progress of the underlying noiseless protocol.

It is already well established that topology matters in communication [9] and in network coding [32]. Our work (along with previous results [2, 36]) suggests that the same holds also for the field of interactive communication when the noise is random. While for certain topologies (e.g., a line, a cycle, a complete graph) one can achieve a coding scheme with slowdown $O(1)$, other topologies necessitate a slowdown of $\Theta(\log n/\log\log n)$, e.g., the star topology. The main open question is to better characterize the way topology affects slowdown.

Open Question 1. *For any function $f(n) \in o(\log n)$, define the exact set of topologies for which n-party interactive coding schemes with $f(n)$ slowdown exist. In particular, characterize the set of topologies for which n-party interactive coding schemes with $O(1)$ slowdown exist.*

While Reference [36] shows that, given any topology, interactive coding with $O(\log n)$ slowdown exists, our lower bound demonstrates a necessary slowdown of only $\Omega(\log n/\log\log n)$. This gap leads to the following question:

Open Question 2. *Show a topology (if such exists) for which $\Omega(\log n)$ slowdown is necessary for n-party interactive coding.*

Currently, we do not have a candidate topology for an $\omega(\log n/\log\log n)$ slowdown, when the parties communicate bits.

6.1 Channels with a Large Alphabet Size

Another interesting question asks what happens if the parties are allowed to send symbols from an alphabet Σ whose size is super-constant, say, $|\Sigma| = \log^{\Omega(1)} n$. Interestingly, assuming such a large alphabet, one can get a stronger lower bound on the blowup of the round complexity and thus of the communication complexity. Namely, a blowup of $O(\log n)$, i.e., without the $\log\log n$ term, can be exhibited for interactive coding of the pointer jumping task over a star network.

The intuitive explanation comes from examining the protocol of the upper bound (Theorem 1.2): the protocol suggests that the clients can always encode a subtree of their input of depth $\log\log n$ using $O(\log n)$ bits. The lower bound (Theorem 5.1) suggests that this is the best they can do. However, when the parties send symbols from a larger alphabet, the depth of the subtree they can communicate by sending $O(\log n)$ symbols substantially decreases to a constant. For example, say that each symbol comes from an alphabet Σ (so the clients' input tree is $|\Sigma|$-ary tree rather than a binary tree, and every edge is labeled by a symbol from Σ). Then, encoding d levels of the tree requires $|\Sigma|^d$ symbols. Thus, assuming $|\Sigma| = \log^{\Omega(1)} n$, one can communicate only a constant number of levels when restricted to sending $O(\log n)$ symbols. This intuition implies that during $\log n$ rounds, the cutoff advances by at most $O(1)$ in expectation, which in turn implies a bound on the rate of $O(1/\log n)$.

See also Reference [19] for a subsequent work that uses the above approach to prove a lower bound of $\Omega(\log n)$ on the communication blowup over a ring network, assuming a large alphabet of size $\Theta(\log n)$.

REFERENCES

[1] Shweta Agrawal, Ran Gelles, and Amit Sahai. 2016. Adaptive protocols for interactive communication. In *Proceedings of the 2016 IEEE International Symposium on Information Theory (ISIT'16)*. 595–599. DOI: http://dx.doi.org/10.1109/ISIT.2016.7541368

[2] Noga Alon, Mark Braverman, Klim Efremenko, Ran Gelles, and Bernhard Haeupler. 2016. Reliable communication over highly connected noisy networks. In *Proceedings of the 2016 ACM Symposium on Principles of Distributed Computing (PODC'16)*. 165–173. DOI: http://dx.doi.org/10.1145/2933057.2933085

[3] Zvika Brakerski, Yael T. Kalai, and Moni Naor. 2014. Fast interactive coding against adversarial noise. *J. ACM* 61, 6, Article 35 (Dec. 2014), 30 pages. DOI: http://dx.doi.org/10.1145/2661628

[4] Mark Braverman. 2012. Towards deterministic tree code constructions. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS'12)*. ACM, 161–167. DOI: http://dx.doi.org/10.1145/2090236.2090250

[5] Mark Braverman and Klim Efremenko. 2014. List and unique coding for interactive communication in the presence of adversarial noise. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS'14)*. 236–245. DOI: http://dx.doi.org/10.1109/FOCS.2014.33

[6] Mark Braverman, Klim Efremenko, Ran Gelles, and Bernhard Haeupler. 2016. Constant-rate coding for multiparty interactive communication is impossible. In *Proceedings of the 48th Annual ACM Symposium on Theory of Computing (STOC'16)*. ACM, 999–1010. DOI: http://dx.doi.org/10.1145/2897518.2897563

[7] Mark Braverman, Ran Gelles, Jieming Mao, and Rafail Ostrovsky. 2017. Coding for interactive communication correcting insertions and deletions. *IEEE Trans. Inf. Theor.* 63, 10 (Oct 2017), 6256–6270. DOI: http://dx.doi.org/10.1109/TIT.2017.2734881

[8] M. Braverman and A. Rao. 2014. Toward coding for maximum errors in interactive communication. *IEEE Transa. Inf. Theor.* 60, 11 (Nov 2014), 7248–7255. DOI: http://dx.doi.org/10.1109/TIT.2014.2353994

[9] Arkadev Chattopadhyay, Jaikumar Radhakrishnan, and Atri Rudra. 2014. Topology matters in communication. In *Proceedings 2014 IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS'14)*. 631–640. DOI: http://dx.doi.org/10.1109/FOCS.2014.73

[10] Kai-Min Chung, Rafael Pass, and Sidharth Telang. 2013. Knowledge-preserving interactive coding. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS'13)*. 449–458. DOI: http://dx.doi.org/10.1109/FOCS.2013.55

[11] Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory* (2nd ed.). John Wiley & Sons.

[12] Klim Efremenko, Ran Gelles, and Bernhard Haeupler. 2016. Maximal noise in interactive communication over erasure channels and channels with feedback. *IEEE Trans. Inf. Theor.* 62, 8 (Aug. 2016), 4575–4588. DOI: http://dx.doi.org/10.1109/TIT.2016.2582176

[13] Abbas El Gamal. 1987. Reliable communication of highly distributed information. In *Open Problems in Communication and Computation*, Thomas M. Coverand B. Gopinath (Eds.). Springer, New York, 60–62. DOI: http://dx.doi.org/10.1007/978-1-4612-4808-8_14

[14] Matthew Franklin, Ran Gelles, Rafail Ostrovsky, and Leonard J. Schulman. 2015. Optimal coding for streaming authentication and interactive communication. *IEEE Trans. Inf. Theor.* 61, 1 (Jan 2015), 133–145. DOI: http://dx.doi.org/10.1109/TIT.2014.2367094

[15] Robert G. Gallager. 1988. Finding parity in a simple broadcast network. *IEEE Trans. Inf Theor.* 34, 2 (Mar 1988), 176–180. DOI: http://dx.doi.org/10.1109/18.2626

[16] Ran Gelles. 2015. Coding for Interactive Communication: A Survey. *Foundations and Trends® in Theoretical Computer Science* 13, 1–2 (2015), 1–157. DOI: 10.1561/0400000079

[17] Ran Gelles and Bernhard Haeupler. 2017. Capacity of interactive communication over erasure channels and channels with feedback. *SIAM J. Comput.* 46, 4 (2017), 1449–1472. DOI: http://dx.doi.org/10.1137/15M1052202

[18] Ran Gelles, Bernhard Haeupler, Gillat Kol, Noga Ron-Zewi, and Avi Wigderson. 2016. Towards optimal deterministic coding for interactive communication. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms*. 1922–1936. DOI: http://dx.doi.org/10.1137/1.9781611974331.ch135

[19] Ran Gelles and Yael T. Kalai. 2017. Constant-rate interactive coding is impossible, even in constant-degree networks. In *Proceedings of the 8th Conference on Innovations in Theoretical Computer Science (ITCS'17)*.

[20] Ran Gelles, Ankur Moitra, and Amit Sahai. 2011. Efficient and explicit coding for interactive communication. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS'11)*. 768–777. DOI: http://dx.doi.org/10.1109/FOCS.2011.51

[21] Ran Gelles, Ankur Moitra, and Amit Sahai. 2014. Efficient coding for interactive communication. *IIEEE Trans. Inf. Theor.* 60, 3 (March 2014), 1899–1913. DOI: http://dx.doi.org/10.1109/TIT.2013.2294186

[22] Ran Gelles, Amit Sahai, and Akshay Wadia. 2015. Private interactive communication across an adversarial channel. *IEEE Trans. Inf. Theor.* 61, 12 (Dec 2015), 6860–6875. DOI: http://dx.doi.org/10.1109/TIT.2015.2483323

[23] Mohsen Ghaffari and Bernhard Haeupler. 2014. Optimal error rates for interactive coding II: Efficiency and list decoding. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS'14)*. 394–403. DOI: http://dx.doi.org/10.1109/FOCS.2014.49

[24] Mohsen Ghaffari, Bernhard Haeupler, and Madhu Sudan. 2014. Optimal error rates for interactive coding I: Adaptivity and other settings. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC'14)*. 794–803. DOI: http://dx.doi.org/10.1145/2591796.2591872

[25] Navin Goyal, Guy Kindler, and Michael Saks. 2008. Lower bounds for the noisy broadcast problem. *SIAM J. Comput.* 37, 6 (2008), 1806–1841. DOI: http://dx.doi.org/10.1137/060654864

[26] Bernhard Haeupler. 2014. Interactive channel capacity revisited. In *Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS'14)*. 226–235. DOI: http://dx.doi.org/10.1109/FOCS.2014.32

[27] William M. Hoza and Leonard J. Schulman. 2016. The adversarial noise threshold for distributed protocols. In *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete Algorithms*. 240–258. DOI: http://dx.doi.org/10.1137/1.9781611974331.ch18

[28] Abhishek Jain, Yael Tauman Kalai, and Allison Lewko. 2015. Interactive coding for multiparty protocols. In *Proceedings of the 6th Conference on Innovations in Theoretical Computer Science (ITCS'15)*. 1–10. DOI: http://dx.doi.org/10.1145/2688073.2688109

[29] Gillat Kol and Ran Raz. 2013. Interactive channel capacity. In *Proceedings of the 45th annual ACM Symposium on Theory of Computing (STOC'13)*. 715–724. DOI:http://dx.doi.org/10.1145/2488608.2488699

[30] S. Kullback and R. A. Leibler. 1951. On information and sufficiency. *Ann. Math. Stat.* 22, 1 (Mar 1951), 79–86.

[31] Allison Lewko and Ellen Vitercik. 2015. Balancing communication for multi-party interactive coding (2015). arXiv:1503.06381. https://arxiv.org/abs/1503.06381.

[32] Lusa Lima, Diogo Ferreira, and Joo Barros. 2012. Topology matters in network coding. *Telecommun. Syst.* 51, 4 (2012), 247–257. DOI:http://dx.doi.org/10.1007/s11235-011-9433-4

[33] Noam Nisan and Avi Wigderson. 1993. Rounds in communication complexity revisited. *SIAM J. Comput.* 22, 1 (1993), 211–219. DOI:http://dx.doi.org/10.1137/0222016

[34] M. S. Pinsker. 1964. Информация и информационная устойчивость случайных величин и процессов. Izdat. Akad. Nauk SSSR, Moscow. Translated under the title 'Information and Information Stability of Random Variables and Processes', Holden-Day, San Francisco.

[35] Stephen J. Ponzio, Jaikumar Radhakrishnan, and S. Venkatesh. 2001. The communication complexity of pointer chasing. *J. Comput. Syst. Sci.* 62, 2 (2001), 323–355. DOI:http://dx.doi.org/10.1006/jcss.2000.1731

[36] Sridhar Rajagopalan and Leonard Schulman. 1994. A coding theorem for distributed computation. In *Proceedings of the 26th Annual ACM Symposium on Theory of Computing (STOC'94)*. 790–799. DOI:http://dx.doi.org/10.1145/195058.195462

[37] Leonard J. Schulman. 1992. Communication on noisy channels: A coding theorem for computation. In *Proceedings of the Annual IEEE Symposium on Foundations of Computer Science*. 724–733. DOI:http://dx.doi.org/10.1109/SFCS.1992.267778

[38] Leonard J. Schulman. 1993. Deterministic coding for interactive communication. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing (STOC'93)*. ACM, 747–756. DOI:http://dx.doi.org/10.1145/167088.167279

[39] Leonard J. Schulman. 1996. Coding for interactive communication. *IEEE Trans. Inf. Theor.* 42, 6 (1996), 1745–1756. DOI:http://dx.doi.org/10.1109/18.556671

[40] Claude E. Shannon. 2001. A mathematical theory of communication. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* 5, 1 (2001), 3–55. DOI:http://dx.doi.org/10.1145/584091.584093 Originally appeared in *Bell System Tech. J.* 27, 379–423 (1948), 623–656.

Received May 2016; revised October 2017; accepted October 2017