The Role of Interactivity in Local Differential Privacy

Matthew Joseph *   Jieming Mao †   Seth Neel ‡   Aaron Roth §

November 11, 2019

Abstract

We study the power of interactivity in local differential privacy. First, we focus on the difference between *fully interactive* and *sequentially interactive* protocols. Sequentially interactive protocols may query users adaptively in sequence, but they cannot return to previously queried users. The vast majority of existing lower bounds for local differential privacy apply only to sequentially interactive protocols, and before this paper it was not known whether fully interactive protocols were more powerful.

We resolve this question. First, we classify locally private protocols by their *compositionality*, the multiplicative factor $k \geq 1$ by which the sum of a protocol's single-round privacy parameters exceeds its overall privacy guarantee. We then show how to efficiently transform any fully interactive k-compositional protocol into an equivalent sequentially interactive protocol with an $O(k)$ blowup in sample complexity. Next, we show that our reduction is tight by exhibiting a family of problems such that for any k, there is a fully interactive k-compositional protocol which solves the problem, while no sequentially interactive protocol can solve the problem without at least an $\tilde{\Omega}(k)$ factor more examples.

We then turn our attention to hypothesis testing problems. We show that for a large class of compound hypothesis testing problems — which include all simple hypothesis testing problems as a special case — a simple noninteractive test is optimal among the class of all (possibly fully interactive) tests.

*University of Pennsylvania, Computer and Information Science. majos@cis.upenn.edu
†University of Pennsylvania, Warren Center. jiemingm@seas.upenn.edu
‡University of Pennsylvania, Wharton Statistics. sethneel@wharton.upenn.edu
§University of Pennsylvania, Computer and Information Science. aaroth@cis.upenn.edu

1 Introduction

In the last several years, differential privacy in the *local* model has seen wide adoption in industry, including at Google [6, 20], Apple [3], and Microsoft [15]. The choice of adopting the *local* model of differential privacy — in which privacy protections are added at each individual's device, before data aggregation — instead of the more powerful *central* model of differential privacy — in which a trusted intermediary is allowed to first aggregate data before adding privacy protections — is driven by practical concerns. Local differential privacy frees the data analyst from many of the responsibilities that come with the stewardship of private data, including liability for security breaches, and the legal responsibility to respond to subpoenas for private data, amongst others. However, the local model of differential privacy comes with its own practical difficulties. The most well known of these is the need to have access to a larger number of users than would be necessary in the central model. Another serious obstacle — the one we study in this paper — is the need for *interactivity*.

There are two reasons why interactive protocols — which query users adaptively, as a function of the answers to previous queries — pose practical difficulties. The first is that communication with user devices is slow: the communication in noninteractive protocols can be fully parallelized, but for interactive protocols, the number of rounds of interactivity becomes a running-time bottleneck. The second is that user devices can go offline or otherwise become unreachable — and so it may not be possible to return to a previously queried user and pose a new query. The first difficulty motivates the study of noninteractive protocols. The second difficulty motives the study of *sequentially interactive* protocols [17] — which may pose adaptively chosen queries — but must not pose more than one query to any user (and so in particular never need to return to a previously queried user).

It has been known since [27] that there can be an exponential gap in the sample complexity between noninteractive and interactive protocols in the local model of differential privacy, and that this gap can manifest itself even in natural problems like convex optimization [32, 33]. However, it was not known whether the full power of the local model could be realized with only *sequentially interactive* protocols. Almost all known lower bound techniques applied only to either noninteractive or sequentially interactive protocols, but there were no known fully interactive protocols that could circumvent lower bounds for sequential interactivity.

1.1 Our Results

We present two kinds of results, relating to the power of sequentially adaptive protocols and non-adaptive protocols respectively. Throughout, we consider protocols operating on datasets that are drawn i.i.d. from some unknown distribution $\mathcal{D}$, and focus on the *sample complexity* of these protocols: how many users (each corresponding to a sample from $\mathcal{D}$) are needed in order to solve some problem, defined in terms of $\mathcal{D}$.

Sequential Interactivity We classify locally private protocols in terms of their *compositionality*. Informally, a protocol is k-compositional if the privacy costs $\{\varepsilon_j^i\}_{j=1}^r$ of the local randomizers executed by any user i over the course of the protocol sum to at most $k\varepsilon$, where ε is the overall privacy cost of the protocol: $\sum_j \varepsilon_j^i \leq k\varepsilon$. When $k = 1$, we say that the protocol is compositional. Compositional protocols capture most of the algorithms studied in the published literature, and in particular, any protocol whose privacy guarantee is proven using the composition theorem for

ε-differential privacy[1].

1. **Upper Bounds**: For any (potentially fully interactive) compositional protocol M, we give a generic and efficient reduction that compiles it into a sequentially interactive protocol M', with only a constant factor blow-up in privacy guarantees and sample complexity, while preserving (exactly) the distribution on transcripts generated. This in particular implies that up to constant factors, sequentially adaptive compositional protocols are as powerful as fully adaptive compositional protocols. More generally, our reduction compiles an arbitrary k-compositional protocol M into a sequentially interactive protocol M' with the same transcript distribution, and a blowup in sample complexity of $O(k)$.

2. **Lower Bounds**: We show that our upper bound is tight by proving a separation between the power of sequentially and fully interactive protocols in the local model. In particular, we define a family of problems (Multi-Party Pointer Jumping) such that for any k, there is a fully interactive k-compositional protocol which can solve the problem given sample complexity $n = n(k)$, but such that no sequentially interactive protocol with the same privacy guarantees can solve the problem with sample complexity $\tilde{o}(k \cdot n)$. Thus, the sample complexity blowup of our reduction cannot be improved in general.

Noninteractivity We then turn our attention to the power of noninteractive protocols. We consider a large class of compound hypothesis testing problems — those such that both the null hypothesis H_0 and the alternative hypothesis H_1 are closed under mixtures. For every problem in this class, we show that the optimal locally private hypothesis test is noninteractive. We do this by demonstrating the existence of a simple hypothesis test for such problems. We then prove that this test's sample complexity is optimal even among the set of all fully interactive tests by extending information theoretical lower bound techniques developed by Braverman et al. [8] and first applied to local privacy by Joseph et al. [25] and Duchi and Rogers [16] to the fully interactive setting.

1.2 Related Work

The local model of differential privacy was introduced by Dwork et al. [19] and further formalized by Kasiviswanathan et al. [27], who also gave the first separation between noninteractive locally private protocols and interactive locally private protocols. They did so by constructing a problem, Masked Parity, that requires exponentially larger sample complexity without interaction than with interaction. Daniely and Feldman [14] later expanded this result to a different, larger class of problems. Smith et al. [32] proved a similar separation between noninteractive and interactive locally private convex optimization protocols that use neighborhood-based oracles.

Recent work by Acharya et al. [1, 2] gives a qualitatively different separation between the private-coin and public-coin models of noninteractive local privacy. Informally, the public-coin model allows for an additional "half step" of interaction over the private-coin model in the form of coordinated local randomizer choices across users. In this paper, we use the public-coin model of noninteractivity.

Duchi et al. [17] introduced the notion of sequential interactivity for local privacy. They also provided the first general techniques for proving lower bounds for sequentially interactive locally

[1]Not every protocol is 1-compositional: exceptions include RAPPOR [20] and the evolving data protocol of Joseph et al. [24].

private protocols by bounding the KL-divergence between the output distributions of ε-locally private protocols with different input distributions as a function of ε and the total variation distance between these input distributions. Bassily and Smith [5] and Bun et al. [7] later generalized this result to (ε, δ)-locally private protocols, and Duchi et al. [18] obtained an analogue of Assouad's method for proving lower bounds for sequentially interactive locally private protocols.

More recently, Duchi and Rogers [16] showed how to combine the above analogue of Assouad's method with techniques from information complexity [8, 22] to prove lower bounds for estimation problems that apply to a restricted class of *fully* interactive locally private protocols. A corollary of their lower bounds is that several known *noninteractive* algorithms are optimal minimax estimators within the class they consider. However, their results do not imply any separation between sequential and full interaction. Moreover, our reduction implies that every (arbitrarily interactive) compositional locally private algorithm can be reduced to a sequentially interactive protocol with only constant blowup in sample complexity, and as a result all known lower bounds for sequentially interactive protocols also hold for arbitrary compositional protocols.

Canonne et al. [10] study simple hypothesis testing under the centralized model of differential privacy, and Theorem 1 of Duchi et al. [17] implies a tight lower bound for sequentially interactive locally private simple hypothesis testing. We extend this lower bound to the fully interactive setting and match it with a noninteractive upper bound for a more general class of compound testing problems that includes simple hypothesis testing as a special case.

Finally, recent subsequent work [26] gives a stronger exponential sample complexity separation separation between the sequentially and fully interactive models. It does so through a general connection between communication complexity and sequentially interactive sample complexity. Applying this connection to a communication problem similar to the "multi-party pointer jumping" described in Section 4 completes the result.

2 Preliminaries

We begin with the definition of approximate differential privacy. Given data domain $\mathcal{X}$, two data sets $S, S' \in \mathcal{X}^n$ are *neighbors* (denoted $S \sim S'$) if they differ in at most one coordinate: i.e. if there exists an index i such that for all $j \neq i$, $S_j = S'_j$. A differentially private algorithm must have similar output distributions on all pairs of neighboring datasets.

Definition 2.1 ([19]). *Let $\varepsilon, \delta \geq 0$. A randomized algorithm $\mathcal{M} : \mathcal{X}^n \to \mathcal{O}$ is* (ε, δ)-differentially private *if for every pair of neighboring data sets $S \sim S' \in \mathcal{X}^n$, and every event $\Omega \subseteq \mathcal{O}$*

$$\mathbb{P}_{\mathcal{M}}\left[\mathcal{A}(S) \in \Omega\right] \leq \exp(\varepsilon)\mathbb{P}_{\mathcal{M}}\left[\mathcal{A}(S') \in \Omega\right] + \delta.$$

When $\delta = 0$, we say that $\mathcal{M}$ satisfies (pure) ε-differential privacy.

Differential privacy has two nice properties. First, it composes neatly: the composition of algorithms $\mathcal{M}_1, \ldots, \mathcal{M}_n$ that are respectively $(\varepsilon_1, \delta_1), \ldots, (\varepsilon_n, \delta_n)$-differentially private is $(\sum_i \varepsilon_i, \sum_i \delta_i)$-differentially private. For pure differential privacy, this is tight in general. Second, differential privacy is resilient to post-processing: given an (ε, δ)-differentially private $\mathcal{M}$ and any function f, $f \circ \mathcal{M}$ is still (ε, δ)-differentially private (see Appendix A.1 for details). For brevity, we often abbreviate "differential privacy" as "privacy".

As defined, the constraint of differential privacy is on the *output* of an algorithm $\mathcal{M}$, not on its internal workings. Hence, it implicitly assumes a trusted data curator, who has access to the entire raw dataset. This is sometimes referred to as differential privacy in the central model. In contrast, this paper focuses on the more restrictive *local* model [19] of differential privacy. In the local model, the private computation is an interaction between n users, each of whom hold exactly one dataset record, and is coordinated by a protocol $\mathcal{A}$. We assume throughout this paper that each user's datum is drawn i.i.d. from some unknown distribution: $x_i \sim_{iid} \mathcal{D}$[2]. Informally, at each round t of the interaction, a protocol $\mathcal{A}$ observes the transcript of interactions so far, selects a user, and assigns the user a randomizer. The user then applies the randomizer to their datum, using fresh randomness for each application, and publishes the output. In turn, the protocol observes the updated transcript, selects a new user-randomizer pair, and the process continues. We define these terms precisely below.

Definition 2.2. *An* (ε, δ)-randomizer $R\colon X \to Y$ *is an* (ε, δ)*-differentially private function taking a single data point as input.*

A simple, canonical, and useful randomizer is *randomized response* [19, 34].

Example 2.1 (Randomized Response)**.** *Given data universe* $\mathcal{X} = [k]$ *and datum* $x_i \in \mathcal{X}$*,* ε*-randomizer* $\mathsf{RR}\,(x_i, \varepsilon)$ *outputs* x_i *with probability* $\frac{e^\varepsilon}{e^\varepsilon + k - 1}$ *and otherwise outputs a uniformly random element of* $\mathcal{X} - \{x_i\}$*.*

Next, we formally define transcripts and protocols.

Definition 2.3. *A* transcript π *is a vector consisting of 5-tuples* $(i^t, R_t, \varepsilon_t, \delta_t, y_t)$ *— encoding the user chosen, randomizer assigned, randomizer privacy parameters, and randomized output produced — for each round* t*.* $\pi_{<t}$ *denotes the transcript prefix before round* t*. Letting* S_π *denote the collection of all transcripts and* S_R *the collection of all randomizers, a* protocol *is a function* $\mathcal{A}\colon S_\pi \to ([n] \times S_R \times \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0}) \cup \{\perp\}$ *mapping transcripts to users, randomizers, and randomizer privacy parameters (*$\perp$ *is a special character indicating a protocol halt).*

The transcript that results from running a locally private computation will often be post-processed to compute some useful function of the data. However, the privacy guarantee must hold even if the entire transcript is observed. Hence, in this paper we abstract away the task that the computation is intended to solve, and view the output of a locally private computation as simply the transcript it generates.

To clarify the role of interaction in these private computations – especially when analyzing reductions between computations with different kinds of interactivity – it is often useful to speak separately of protocols and *experiments*. While the protocol $\mathcal{A}$ is a function mapping transcripts to users and randomizers, the experiment is the interactive process that maps a protocol and collection of users drawn from a distribution $\mathcal{D}$ to a finished transcript. In the simplest case, $\mathsf{FollowExpt}$ (Algorithm 1), the experiment exactly follows the outputs of its protocol.

However, experiments may in general heed, modify, or ignore the outputs of their input protocol. We delineate the privacy characteristics of experiment-protocol pairs and protocols in isolation below. Here and throughout, the dataset is not viewed as an input to an experiment, but is drawn from $\mathcal{D}$ by the experiment-protocol pair. Drawing a fresh user $\sim \mathcal{D}$ corresponds to adding an additional data point, and so the sample complexity of an experiment-protocol pair is the number

[2] Roughly speaking, this corresponds to a setting in which users are "symmetric" and in which nothing differentiates them a priori. All of our results generalize to the setting in which there are different "types" of users, known to the protocol up front.

Algorithm 1

procedure $\mathsf{FollowExpt}(\mathcal{A}, \mathcal{D}, n)$
  Draw n users $\{x_i\} \sim \mathcal{D}^n$
  Initialize transcript $\pi_0 \leftarrow \emptyset$
  for $t = 1, 2, \ldots$ **do**
    if $\mathcal{A}(\pi_{<t}) = \perp$ **then**
      Output transcript $\pi_{<t}$
    else
      $(i^t, R_t, \varepsilon_t, \delta_t) \leftarrow \mathcal{A}(\pi_{<t})$
      User i^t publishes $y_t \sim R_t(x_{i^t}, \varepsilon_t, \delta_t)$
    end if
  end for
end procedure

of draws from $\mathcal{D}$ over the run of the algorithm. For the simple algorithm $\mathsf{FollowExpt}(\mathcal{A})$ defined above, the sample complexity is always n. Finally we remark that although the distribution $\mathcal{D}$ and the sample complexity n are inputs to the experiment, for brevity we typically omit them and focus on the protocol $\mathcal{A}$; e.g. writing $\mathsf{Expt}(\mathcal{A})$ rather than $\mathsf{Expt}(\mathcal{A}, \mathcal{D}, n)$.

Definition 2.4. *Experiment-protocol pair* $\mathsf{Expt}(\mathcal{A})$ *satisfies* (ε, δ)-local differential privacy *(LDP) if it is* (ε, δ)*-differentially private in its transcript outputs. A protocol* $\mathcal{A}$ *satisfies* (ε, δ)*-local differential privacy (LDP) if experiment-protocol pair* $\mathsf{FollowExpt}(\mathcal{A})$ *is* (ε, δ)*-locally differentially private.*

Experiment-protocol pairs can be, by increasing order of generality, *noninteractive*, *sequentially interactive*, and *fully interactive*.

Definition 2.5. *An experiment-protocol pair* $\mathsf{Expt}(\mathcal{A})$ *is* noninteractive *if, at each round* t*, as random variables,* $(i^t, R_t, \varepsilon_t, \delta_t) \perp\!\!\!\perp \Pi_{<t} \mid t$.

In other words, noninteractivity forces nonadaptivity, and all user-randomizer assignments are made before the experiment begins. In contrast, in sequentially interactive experiment-protocol pairs, users may be queried adaptively, but only once.

Definition 2.6. *An experiment-protocol pair* $\mathsf{Expt}(\mathcal{A})$ *is* sequentially interactive *if, at each round* t*,* $i^t \neq i^{t-1}, \ldots, i^1$.

Finally, in in fully interactive experiments, the experiment-protocol may make user-randomizer assignments adaptively, and each user may receive arbitrarily many randomizer assignments. Along the same lines, we say a protocol $\mathcal{A}$ is noninteractive (respectively sequentially and fully interactive) if $\mathsf{FollowExpt}(\mathcal{A})$ is a noninteractive (respectively sequentially and fully interactive) experiment-protocol pair. This experiment-protocol formalism will be useful in constructing the full-to-sequential reduction in Section 3; elsewhere, we typically elide the distinction and simply reason about $\mathsf{FollowExpt}(\mathcal{A})$ as "protocol $\mathcal{A}$". For any locally private protocol, we refer to the number of users n that it queries as its *sample complexity.* For fully interactive protocols, the total number of rounds — which we denote by T — may greatly exceed n. In contrast, for both non-interactive and sequentially interactive protocols, the number of rounds $T \leq n$.

At each round t of a fully interactive ε-locally private protocol, we know that $\varepsilon_t \leq \varepsilon$. For many protocols, we can say more about how the ε_t parameters relate to ε:

Definition 2.7. *Consider an ε-locally private protocol $\mathcal{A}$. Let $\{\varepsilon_t\}_{t=1}^T$ denote the minimal privacy parameters of the local randomizers R_t selected at round t considered as random variables. We say the protocol $\mathcal{A}$ is k-*compositionally private *if for all $i \in [n]$, with probability 1 over the randomness of the transcript,*

$$\sum_{t:\ i_t = i} \varepsilon_t \leq k\varepsilon.$$

If $k = 1$, a protocol is simply compositional private.

Remark. *In fact, all of our results hold without modification even under the weaker condition of* average *k-compositionality. For a protocol $\mathcal{A}$ with sample complexity n, $\mathcal{A}$ is k-compositional on average if*

$$\sum_t \varepsilon_t \leq k\varepsilon n.$$

For brevity, we often shorthand "k-compositionally private" as simply "k-compositional".

Informally, a compositionally private protocol is one in which the privacy parameters for each user "just add up." Almost every locally private protocol studied in the literature (and in particular, every protocol whose privacy analysis follows from the composition theorem for pure differential privacy) is compositionally private[3]. They are so ubiquitous that it is tempting to guess that all $(\varepsilon, 0)$-locally private protocols are compositional. However, this is false: for every k and ε, there are ε-locally private protocols that fail to be k-compositionally private. The following example shows that by taking advantage of special structure in the data domain and choice of randomizers it is possible to achieve $(\varepsilon, 0)$-local privacy, even as the sum of the round-by-round privacy parameters greatly exceeds ε.

Example 2.2 (Informal)**.** *Let the data universe $\mathcal{X}$ consist of the canonical basis vectors $e_1, \ldots, e_d \in \{0,1\}^d$, and let each $x_1, \ldots, x_n$ be an arbitrary element of $\mathcal{X}$. Consider the d round protocol where, for each round $j \in [d]$, every user i with $x_i = e_j$ outputs a sample from $\mathsf{RR}(1, \varepsilon)$, and the remaining users output a sample from $\mathsf{Ber}(0.5)$. As $\mathsf{RR}(\cdot, \varepsilon)$ is an ε-local randomizer which each user employs only once, and remaining outputs are data-independent, this protocol is ε-locally private. But the protocol fails to be k-compositionally private for $k < d/2$.*

The preceding example demonstrates that the careful choice of local randomizers based on the data universe structure can strongly violate compositional privacy. Seen another way, when multiple queries are asked of the same user, there are situations in which the correlation in privatized responses induced by being run on the same data element can lead to arbitrarily sub-compositional privacy costs. The main result of our paper is that the additional power of a fully interactive protocol, on top of sequential interactivity, is characterized by its compositionality.

3 From Full to Sequential Interactivity

We show that any $(\varepsilon, 0)$-locally private compositional protocol is "equivalent" to a sequentially interactive protocol with sample complexity that is larger by only a small constant factor. By equivalent, we mean that for any $(\varepsilon, 0)$-locally private compositional protocol, we can exhibit a

[3]This simple compositionality applies even if $\{\varepsilon_t\}_{t=1}^T$ are chosen adaptively in each round (see Theorem 3.6 in Rogers et al. [30]).

sequentially interactive $(3\varepsilon, 0)$-locally differentially private protocol with only a constant factor larger sample complexity that induces exactly the same distribution on transcripts. Thus for any task for which the original protocol was useful, the sequentially interactive protocol is just as useful[4].

More generally, we give a generic reduction under which any $(\varepsilon, 0)$-private k-compositional protocol can be compiled into a sequentially interactive protocol with an $e^{\varepsilon}k$-factor increase in sample complexity.

Our proof is constructive; given an arbitrary k-compositional $(\varepsilon, 0)$-locally differentially private protocol we show how to simulate it using a sequentially interactive protocol that induces the same joint distribution on transcripts. The "simulation" is driven by three main ideas:

1. **Bayesian Resampling**: The dataset used in a locally differentially private protocol is static once the protocol begins. However, we consider the following thought experiment: each user's datum is *resampled* from the posterior distribution on their datum, conditioned on the transcript thus far, before every round in which they are given a local randomizer. We observe that the mechanism from this thought experiment induces exactly the same joint distribution on datasets and transcripts upon completion of the mechanism. Thus, for the remainder of the argument, we can seek to simulate this "Bayesian Resampling" version of the mechanism.

2. **Private Rejection Sampling**: Because of the local differential privacy guarantee, at any step of the algorithm, the posterior on a user's datum conditioned on the private transcript generated so far must be close to their prior. Thus, it is possible to sample from this posterior distribution by first sampling from the prior, and then applying a rejection sampling step that is both a) likely to succeed, and b) differentially private. Sampling from the prior simply corresponds to querying a new user. At first glance, applying rejection sampling as needed seems to require information that the users will not have available, because they do not know the underlying data distribution $\mathcal{D}$. But an application of Bayes rule, together with a data independent rescaling can be used to re-write the required rejection probability using only quantities that each user can compute from her own data point and the transcript. A similar use of rejection sampling appears in the simulation of locally private algorithms by statistical query algorithms given by Kasiviswanathan et al. [27].

3. **Data Independent Decomposition of Local Randomizers**: The two ideas above suffice to transform a fully interactive mechanism into a sequentially interactive mechanism, with a blowup in sample complexity from n to T (because in the sequentially interactive protocol that results from rejection sampling, each user applies only one local randomizer instead of an average of T/n). However, we generalize a recent result of [4] to show that any ε_i-private local randomizer can be described as a mixture between a *data independent* distribution and an $(\varepsilon, 0)$-private local randomizer for any $\varepsilon > \varepsilon_i$, where the weight on the data independent distribution is roughly (for small constant ε) $1 - \varepsilon_i/\varepsilon$. Thus we can simulate each local

[4]Formally, for any loss function defined over a data distribution $\mathcal{D}$ and a transcript Π, when data points x_i are drawn i.i.d. from $\mathcal{D}$, the two protocols induce exactly the same distribution over transcripts, and hence the same distribution over losses. Once one restricts attention to locally private protocols with privacy parameter $\varepsilon \leq 1$ that take as input points drawn i.i.d. from a distribution $\mathcal{D}$, it is without loss of generality to measure the success or failure of a protocol with respect to the underlying distribution $\mathcal{D}$, rather than with respect to the sample. This is because such protocols are $\approx \varepsilon/\sqrt{n}$ differentially private when viewed in the central model of differential privacy (in which the input may be permuted before used in the protocol) [4, 21], and hence the distribution on transcripts would be almost unchanged even if the entire dataset was *resampled* i.i.d. from $\mathcal{D}$. [13, 28]. Thus, for such protocols, the transcript distribution is governed by the data distribution $\mathcal{D}$, but not (significantly) by the sample.

randomizer while only needing to query a new user with probability $\varepsilon_i/\varepsilon$. As a result, for any compositional mechanism, 1 user in the sequential setting suffices (in expectation) to simulate the entire transcript of a single user in the fully interactive setting. More generally, if the mechanism is k-compositional, then k users are required in expectation to carry out the simulation. The realized sample complexity concentrates sharply around its expectation.

3.1 Step 1: A Bayesian Thought Experiment

The first step of our construction is to observe that for any locally private protocol $\mathcal{A}$, $\mathsf{BayesExpt}(\mathcal{A})$ induces exactly the same distribution over transcripts as $\mathsf{FollowExpt}(\mathcal{A})$. The difference is that in $\mathsf{BayesExpt}(\mathcal{A})$, between each interaction with a given user i, their datum x_i is *resampled* from the posterior distribution on user i's data conditioned on the portion of the transcript generated thus far. We prove in Lemma 3.1 that the two experiments produce exactly the same transcript distribution. Once we establish this, our goal will be to simulate the transcript distribution induced by $\mathsf{BayesExpt}(\mathcal{A})$.

Algorithm 2

procedure $\mathsf{BayesExpt}(\mathcal{A}, \mathcal{D}, n\ )$
  Initialize transcript $\pi_0 = \emptyset$
  for $t = 1, 2, \ldots$ **do**
    if $\mathcal{A}(\pi_{<t}) = \perp$ **then**
      Output transcript $\pi_{<t}$
    else
      $(i^t, R_t, \varepsilon_t, \delta_t) \leftarrow \mathcal{A}(\pi_{<t})$
      Redraw $x_{i^t} \sim Q_{i,t}$ ▷ $Q_{i,t}$ is the posterior on x_{i^t} given $\pi_{<t}$
      User i^t publishes $y_t \sim R_t(x_{i^t})$
    end if
  end for
end procedure

Note that when i^t is selected for the first time $Q_{i,t} = \mathcal{D}$, and so the sample complexity (e.g. number of draws from $\mathcal{D}$) of $\mathsf{BayesExpt}(\mathcal{A})$ is bounded by n.

Lemma 3.1. *For any protocol $\mathcal{A}$, Let Π^f be the transcript random variable that is output by* $\mathsf{FollowExpt}(\mathcal{A})$ *and let Π^b be the transcript output by* $\mathsf{BayesExpt}(\mathcal{A})$. *Then*

$$\Pi^f \overset{d}{=} \Pi^b$$

where $\overset{d}{=}$ denotes equality of distributions.

Proof. We show this by (strong) induction on rounds in the transcript. The base case $t = 1$ is immediate: for any index i^1 selected by $\mathsf{BayesExpt}(\mathcal{A})$, the posterior distribution $Q_{i,1}$ is the same as the prior $\mathcal{D}$.

Now suppose it is true up to time $t+1$, i.e. $\Pi^f_{<t+1} \overset{d}{=} \Pi^b_{<t+1}$. Then since the joint distributions $\Pi_{<t+2}$ factor as $(i^{t+1}, R_{t+1}, \varepsilon_{t+1}, \delta_{t+1}, Y_{t+1} | \Pi_{<t+1}) \cdot \Pi_{<t+1}$, it suffices to show that the conditional distributions on $i^{t+1}, R_{t+1}, \varepsilon_{t+1}, \delta_{t+1}, Y_{t+1} | \Pi_{<t+1}$ coincide. Moreover, the conditional distribution on $i^{t+1}, R_{t+1}, \varepsilon_{t+1}, \delta_{t+1} | \Pi_{<t+1}$ is given by $\mathcal{A}(\Pi_{<t+1})$ under both algorithms, and so it remains only to

show that $Y_{t+1}|i^{t+1}, R_{t+1}, \varepsilon_{t+1}, \delta_{t+1}, \Pi_{<t+1}$ is the same distribution under both algorithms. Under $\mathsf{FollowExpt}(\mathcal{A})$,

$$Y_{t+1}|i^{t+1}, R_{t+1}, \varepsilon_{t+1}, \delta_{t+1}, \Pi_{<t+1} \sim R_{t+1}(x_{i^{t+1}}, \varepsilon_{t+1}, \delta_{t+1}|\Pi_{<t+1}) \stackrel{d}{=} R_{t+1}(u, \varepsilon_{t+1}, \delta_{t+1}),$$

where $u \stackrel{d}{=} x_{i^{t+1}}|\Pi_{<t+1} \stackrel{d}{=} Q_{i,t+1}$ by definition, and we use the fact that after conditioning on $\Pi_{<t+1}$, $x_{i^{t+1}}$ is independent of ε_{t+1} and δ_{t+1}. Redrawing $u \sim Q_{i,t+1}$ does not change the marginal distribution of $R_{t+1}(u, \varepsilon_{t+1}, \delta_{t+1})$, which is exactly the distribution under $\mathsf{BayesExpt}(\mathcal{A})$, as desired. □

3.2 Step 2: Sequential Simulation of Algorithm 2 via Rejection Sampling

We now show how to replace step 8 in Algorithm 2 by selecting a new datapoint (drawn from $\mathcal{D}$) at every round and using rejection sampling to simulate a draw from $Q_{i,t}$. The result is a sequentially interactive mechanism that preserves the transcript distribution of Algorithm 2 (and, by Lemma 3.1, of Algorithm 1), albeit one with a potentially very large increase in sample complexity (from n to T). The rejection sampling step increases the privacy cost of the protocol by at most a factor of 2.

We first review why it is non-obvious that rejection sampling can be performed in this setting. We want to sample from the target distribution $Q_{i,t}$, the posterior $x_i^t|\pi_{<t}$, using samples from the proposal distribution $\mathcal{D}$. Let p_π denote the density function of $Q_{i,t}$ and let p denote the density function of $\mathcal{D}$. In rejection sampling, we would typically sample $u \sim \mathcal{D}$, and with probability $\propto \frac{p_\pi(u)}{p(u)}$ we would accept u as a sample drawn from $Q_{i,t}$, or else redraw another u and continue.

This is not immediately possible in our setting, since the individuals (who must perform the rejection sampling computation) do not know the prior density p and hence do not know the posterior p_π. As a result, they cannot compute either the numerator or denominator of the expression for the acceptance probability. We solve this problem by using the fact that we are simulating a posterior with a prior distribution, and formulate the rejection sampling probability ratio as a quantity depending only on a user's private data point and the transcript. Users may then compute this quantity themselves.

To define our transformed rejection sampler we set up some new notation: given a user i and round t, let $\pi_{<t,i}$ denote the subset of the realized transcript up to time t that corresponds to user i's data, i.e. $\pi_{<t,i} = \{(i^{t'}, R_{t'}, \varepsilon_{t'}, \delta_{t'}, y_{t'}) : t' < t, i^{t'} = i\}$. Let $\mathbb{P}_{x_i}[\pi_{<t,i}]$ denote the conditional probability of the messages corresponding to user i given the choices of privacy parameters and randomizers up to time t:

$$\mathbb{P}[\pi_{<t,i}] = \prod_{t' : i^{t'} = i} \mathbb{P}_{R_{t'}}[R_{t'}(x_i, \varepsilon_{t'}, \delta_{t'}) = y_{t'}].$$

Using this notation, we define our rejection sampling procedure $\mathsf{RejSamp}$ in Algorithm 3.

We now prove that $\mathsf{RejSamp}$ is private and does not need to sample many users.

Lemma 3.2. *Let $Y_t \stackrel{d}{=} R_t(x')$, where $x' \sim Q_{i,t}$ and let Y'_t be defined by the rejection sampling algorithm* $\mathsf{RejSamp}$ *above. Let the sample complexity N be the total number of new users x drawn in step 4 of* $\mathsf{RejSamp}$. *Then* $\mathsf{RejSamp}$ *is $(\varepsilon + \varepsilon_t, 0)$-locally private, $Y_t \stackrel{d}{=} Y'_t$, and $\mathbb{E}[N] \leq 2e^{\varepsilon}$.*

Proof of Lemma 3.2.

Algorithm 3 Rejection Sampling

procedure $\mathsf{RejSamp}(i, \pi_{<t}, \varepsilon, \varepsilon_t, R_t(\cdot), \mathcal{D})$ ▷ Publishing $\Pi_{<t}$ is $(\varepsilon, 0)$-private
    Initialize indicator $\mathsf{accept} \leftarrow 0$
    while $\mathsf{accept} = 0$ **do**
        Draw a new user $x \sim \mathcal{D}$
        User x computes $p_x \leftarrow \frac{\mathbb{P}_x[\pi_{<t,i}]}{\max_{x^*} \mathbb{P}_{x^*}[\pi_{<t,i}]}$
        User x publishes $\mathsf{accept} \sim \mathsf{Ber}\,(p_x/2)$
        if $\mathsf{accept} = 1$ **then**
            User x outputs $Y_t' \sim R_t(x, \varepsilon_t)$
        end if
    end while
end procedure

Claim 3.3. $\mathsf{RejSamp}$ *is* $(\varepsilon + \varepsilon_t)$*-locally private.*

We first show that publishing a draw from $\mathsf{Ber}\,(p_x/2)$ is $(\varepsilon, 0)$-locally private. By assumption publishing $\pi_{<t}$, and hence publishing $\pi_{<t,i}$ (by post-processing), $(\varepsilon, 0)$-private. Hence for any $x \in \mathcal{X}$

$$\mathbb{P}\left[1 \mid x\right] = p_x/2 = \frac{\mathbb{P}_x\left[\pi_{<t,i}\right]}{2 \max_{x^*} \mathbb{P}_{x^*}\left[\pi_{<t,i}\right]} \in [1/(2e^{\varepsilon}), 1/2].$$

Therefore for any x, x', $\mathbb{P}\left[1 \mid x\right] \leq e^{\varepsilon}\mathbb{P}\left[1 \mid x'\right]$. Similarly,

$$\mathbb{P}\left[0 \mid x\right] = (1 - p_x/2) \in [1/2, (2e^{\varepsilon} - 1)/2e^{\varepsilon}]$$

and by $1 + x \leq e^x$, we get $1 - \varepsilon \leq e^{-\varepsilon}$, so $1 + \varepsilon \geq 2 - e^{-\varepsilon}$, and $e^{\varepsilon}/2 \geq (2^{\varepsilon} - 1)/(2e^{\varepsilon})$. Thus for any x, x', $\mathbb{P}\left[0 \mid x\right] \leq e^{\varepsilon}\mathbb{P}\left[0 \mid x'\right]$.

Releasing $R_t(x, \varepsilon_t)$ is ε_t-locally private, so by composition the whole process is $(\varepsilon + \varepsilon_t)$-locally private.

Claim 3.4. $Y_t \overset{d}{=} Y_t'$

It suffices to show that $x|\{\mathsf{accept} = 1\} \overset{d}{=} Q_{i,t}$. Fix any $x_0 \in \mathcal{X}$. Then by Bayes' rule

$$\begin{aligned}
\mathbb{P}\left[x = x_0 \mid \mathsf{accept} = 1\right] &= \mathbb{P}\left[\mathsf{accept} = 1 \mid x = x_0\right] \cdot \frac{\mathbb{P}\left[x = x_0\right]}{\mathbb{P}\left[\mathsf{accept} = 1\right]} \\
&= \frac{\mathbb{P}_{x_0}\left[\pi_{<t,i}\right]}{\max_{x^*} \mathbb{P}_{x^*}\left[\pi_{<t,i}\right]} \cdot \frac{\mathbb{P}\left[x = x_0\right]}{\sum_{x'} \mathbb{P}\left[x = x'\right] \frac{\mathbb{P}_{x'}[\pi_{<t,i}]}{\max_{x^*} \mathbb{P}_{x^*}[\pi_{<t,i}]}} \\
&= \frac{\mathbb{P}_{x_0}\left[\pi_{<t,i}\right] \mathbb{P}\left[x = x_0\right]}{\sum_{x'} \mathbb{P}\left[x = x'\right] \mathbb{P}_{x'}\left[\pi_{<t,i}\right]} \\
&= \frac{\mathbb{P}_{x_0}\left[\pi_{<t,i}\right] \mathbb{P}\left[x = x_0\right]}{\mathbb{P}\left[\pi_{<t,i}\right]} \\
&= \mathbb{P}\left[x = x_0 | \pi_{<t,i}\right] \overset{d}{=} Q_{i,t},
\end{aligned}$$

as desired. Finally, since $p_x/2 \geq \frac{1}{2e^{\varepsilon}}$, the expected number of samples until $\mathsf{accept} = 1$ is $\leq 2e^{\varepsilon}$. $\square$

3.3 Step 3: Data Independent Decomposition of Local Randomizers

The preceding sections enable us to simulate a fully interactive k-compositional $(\varepsilon, 0)$-locally private protocol with a sequentially interactive $(2\varepsilon, 0)$-locally private protocol. However, our solution so far may require sampling a new user for each query in the original protocol. Since a fully interactive protocol's query complexity may greatly exceed its sample complexity, this is undesirable. To address this problem, we *decompose* each local randomizer in a way that substantially reduces the number of queries that actually require samples.

Let $R : \mathcal{X} \to \mathcal{Y}$ be an ε' local randomizer, fix an arbitrary element $x_0 \in \mathcal{X}$, and let x be a private input to R. Then Lemma 5.2 in Balle et al. [4] shows that we can write $R(x)$ as a mixture $\gamma w + (1-\gamma)d_x$, where w is a data-independent distribution, d_x is a data-dependent distribution, and $\gamma \geq e^{-\varepsilon'}$. This suggests that decomposition — by answering a proportion of queries from data-independent distributions — can reduce the sample complexity of our solution. Unfortunately, the data dependent distribution need not be differentially private (in fact, it often corresponds to a point mass on the private data point), so the privacy of the overall mechanism crucially relies on not releasing *which* of the two mixture distributions the output was sampled from.

We first generalize this result, showing that for any $\varepsilon \geq \varepsilon'$, we can write $R(x)$ as $(1-\gamma)w + \gamma\tilde{R}(x)$ where $\tilde{R}$ is a 2ε-differentially private local randomizer, and $\gamma = \frac{e^{-\varepsilon'}-1}{e^{-\varepsilon}-1}$ (Lemma 3.5). The upshot of this generalization is that even if we make public which part of the mixture distribution was used, the resulting privacy loss is still bounded by 2ε. Larger values of ε increase our chance of sampling from a data-independent distribution when simulating a local randomizer, while increasing the privacy cost incurred by a user in the event that we sample from the data-dependent mixture component. This tradeoff will be crucial for us in the proof of our main result.

Lemma 3.5 (Data Independent Decomposition)**.** *Let $R : \mathcal{X} \to \mathcal{Y}$ be an ε'-differentially private local randomizer and let $\varepsilon \geq \varepsilon'$. Then there exists a mapping $\tilde{R}$ and fixed data-independent distribution μ such that $\tilde{R}(\cdot)$ is a $2\varepsilon-$differentially private local randomizer and*

$$R(x) \stackrel{d}{=} \gamma\tilde{R}(x) + (1-\gamma)\mu,$$

where $\gamma = \frac{e^{-\varepsilon'}-1}{e^{-\varepsilon}-1}$.

Proof. Let $0 < \varepsilon' \leq \varepsilon$, fix any $x_0 \in \mathcal{X}$, let $\gamma = \frac{e^{-\varepsilon'}-1}{e^{-\varepsilon}-1}$, and let $r(x)$ denote the density function of the local randomizer R with input x implicitly evaluated at some arbitrary point in the range, which we suppress. Since $\varepsilon \geq \varepsilon' > 0$, $\gamma \in (0, 1]$ is a valid mixture probability. Thus we can write

$$r(x) = (r(x) - (1-\gamma)r(x_0)) + (1-\gamma)r(x_0)$$

and, rewriting the first term,

$$r(x) - (1-\gamma)r(x_0) = \gamma(r(x_0) + \frac{1}{\gamma}(r(x) - r(x_0))) = \gamma\tilde{r}(x).$$

$\tilde{r}$ defines a new mapping $\tilde{R}(\cdot)$ by mapping x to the random variable $\tilde{R}(x)$ with density function $\tilde{r}(x) = (r(x_0) + \frac{1}{\gamma}(r(x) - r(x_0)))$. Thus, it suffices to show that the mapping $\tilde{R}(x)$ is a 2ε-private local randomizer.

We first show that for any x, $\tilde{r}(x)$ is a well-defined density function. Since R is an ε'-private local randomizer, $r(x) - r(x_0) \geq (e^{-\varepsilon'} - 1)r(x_0)$, and so

$$r(x_0) + \frac{1}{\gamma}(r(x) - r(x_0)) \geq r(x_0)\left(1 + \frac{e^{-\varepsilon'} - 1}{\gamma}\right) = r(x_0)e^{-\varepsilon}.$$

This establishes that $\tilde{r}(x)$ is non-negative. Then since

$$\int_\Omega \tilde{r}(x) = \int_\Omega r(x_0) + \frac{1}{\gamma}\int_\Omega (r(x) - r(x_0)) = 1 + \frac{1}{\gamma}(1-1) = 1,$$

$\tilde{r}(x)$ defines a valid density function for any x.

To see that $\tilde{r}$ is also a 2ε-private local randomizer, fix any outcome $o \in \mathcal{Y}$ and any other $x' \in \mathcal{X}$. Since r is an ε'-local randomizer, $r(x) - r(x_0) \leq r(x_0)(e^{\varepsilon'} - 1)$ and we get

$$\begin{aligned}
\tilde{r}(x) &= r(x_0) + \frac{1}{\gamma}(r(x) - r(x_0)) \\
&\leq r(x_0)\left[1 + \frac{1}{\gamma}\left(e^{\varepsilon'} - 1\right)\right] \\
&= r(x_0)\left[1 + \frac{1 - e^{-\varepsilon}}{1 - e^{-\varepsilon'}}\left(e^{\varepsilon'} - 1\right)\right] \\
&= r(x_0)\left[1 + e^{\varepsilon'} \cdot \left(1 - e^{-\varepsilon}\right)\right] \\
&\leq r(x_0)\left[1 + e^{\varepsilon} \cdot \left(1 - e^{-\varepsilon}\right)\right] = r(x_0)e^{\varepsilon}.
\end{aligned}$$

We already showed $\tilde{r}(x') \geq e^{-\varepsilon} r(x_0)$, so

$$\frac{\tilde{r}(x)(o)}{\tilde{r}(x')(o)]} \leq \frac{e^{\varepsilon} r(x_0)(o)}{e^{-\varepsilon} r(x_0)(o)} \leq e^{2\varepsilon}.$$

□

3.4 Putting it All Together: The Complete Simulation

Finally, we combine rejection sampling and decomposition to give our complete reduction, Algorithm 4. We use rejection sampling to convert from a fully interactive mechanism to a sequentially interactive one and use our data-independent decomposition of local randomizers to reduce the sample complexity of the converted mechanism.

We now prove that Reduction has the desired interactivity, privacy, transcript, and sample complexity guarantees. We again denote by N the sample complexity of Reduction, i.e. the number of samples drawn from the prior $\mathcal{D}$ over the run of the algorithm, either in Step 15 (which is bounded by n), or over the runs of RejSamp in line 10. We observe that sampling from the prior $\mathcal{D}$ simply corresponds to using a new datapoint drawn from $\mathcal{D}$. Fixing a protocol $\mathcal{A}$, let Π^r denote the transcript random variable generated by Reduction($\mathcal{A}$), and let Π^b denote the transcript random variable generated by BayesExpt($\mathcal{A}$).

Theorem 3.6. *Let $\mathcal{A}$ a fully-interactive k-compositional $(\varepsilon, 0)$-locally private protocol. Then*

Algorithm 4 Reduction

procedure Reduction(Fully interactive $(\varepsilon, 0)-$LDP Protocol $\mathcal{A}, \mathcal{D}, n$)
  Initialize $s_1, \ldots, s_n \leftarrow 0$. ▷ indicator if user i has been selected yet
  for $t = 1 \ldots$ **do**
    if $\mathcal{A}(\pi_{<t}) =\perp$ **then**
      Output transcript $\pi_{<t}$
    else
      $(i^t, R_t, \varepsilon_t) \leftarrow \mathcal{A}(\pi_{<t})$
      if $s_i^t = 1$ **then**
        Let $\gamma \leftarrow \frac{e^{-\varepsilon_t}-1}{e^{-\varepsilon}-1}$
        Let $R_t = \gamma \tilde{R}_t + (1-\gamma) R_t(x_0)$ ▷ Data Decomposition
        Draw $\rho \sim \mathsf{Unif}(0,1)$
        if $\rho \leq \gamma$ **then**
          Draw $Y_t \sim \mathsf{RejSamp}(i^t, \pi_{<t}, \varepsilon, 2\varepsilon, \tilde{R}(\cdot), \mathcal{D})$
        else
          Draw $Y_t \sim R_t(x_0, \varepsilon_t)$ ▷ Data independent distribution
        end if
      else
        Draw $x_{i^t} \sim Q_{i,t} = \mathcal{D}$, then draw $Y_t \sim R_t(x_{i^t}, \varepsilon_t)$ ▷ $Q_{i,t} = \mathcal{D}$ since $s_{i^t} = 0$
        Let $s_{i^t} \leftarrow 1$
      end if
    end if
  end for
end procedure

1. $\mathsf{Reduction}(\mathcal{A})$ *is sequentially interactive,*

2. $\mathsf{Reduction}(\mathcal{A})$ *is* $(3\varepsilon, 0)$*-locally private,*

3. $\Pi^r \stackrel{d}{=} \Pi^b$,

4. $\mathbb{E}[N] \leq n(\frac{2e^{\varepsilon}\cdot\varepsilon}{1-e^{-\varepsilon}}k + 1)$*, and with probability* $1-\delta$*,* $N = O(nk + \sqrt{nk\log\frac{1}{\delta}})$.

Proof of Theorem 3.6. **1. Interactivity:** Since each user i's data is only used once (before s_i is set to 1), Reduction is sequentially interactive.

2. Privacy: Consider a data point x corresponding to an arbitrary user over the run of $\mathsf{Reduction}(\mathcal{A})$. Then either x is drawn in line 18, or x is drawn during a rejection sampling step. In the first case, x is only used once in step 18, as input to an ε_t-local randomizer, preserving $(\varepsilon, 0)$-LDP, since $\varepsilon_t \leq \varepsilon$. If x is drawn during the rejection sampling step, then it is used during the use of rejection sampling to simulate a draw from a $(2\varepsilon, 0)$-local randomizer $\tilde{R}(\cdot)$, where the input transcript $\pi_{<t}$ has been generated $(\varepsilon, 0)$-privately. The privacy of the input transcript is relevant because it bounds the privacy of the user's rejection sampling step. By Lemma 3.2, this is $(3\varepsilon, 0)$-private.

3. Transcripts: We prove this claim by a similar argument as that of Lemma 3.1: we show by induction that the transcript distribution at each step t is the same for $\mathsf{Reduction}(\mathcal{A})$ and $\mathsf{BayesExpt}(\mathcal{A})$. This is trivially true at $t = 1$. Now suppose it is true up to time $t+1$, i.e. $\Pi^r_{<t+1} \stackrel{d}{=} \Pi^b_{<t+1}$. Then since the joint distributions $\Pi_{<t+2}$ factor as $(i^{t+1}, R_{t+1}, \varepsilon_{t+1}, Y_{t+1} | \Pi_{<t+1}) \cdot \Pi_{<t+1}$, it suffices to show that the conditional distributions on $i^{t+1}, R_{t+1}, \varepsilon_{t+1}, Y_{t+1} | \Pi_{<t+1}$ coincide.

Note that under both $\mathsf{Reduction}(\mathcal{A})$ and $\mathsf{BayesExpt}(\mathcal{A})$, protocol $\mathcal{A}$ is used to select $i^{t+1}, R_{t+1}, \varepsilon_{t+1}$ as a function of $\Pi_{<t+1}$, so we can condition on $i^{t+1}, R_{t+1}, \varepsilon_{t+1}$ as well, and need only show that the distribution on Y_{t+1} is the same. Under $\mathsf{BayesExpt}(\mathcal{A})$, Y_{t+1} is drawn from $R_{t+1}(u, \varepsilon_{t+1}), u \sim Q_{i,t+1}$. There are two cases for $\mathsf{Reduction}(\mathcal{A})$:

- If $s_i^{t+1} = 0$, then under $\mathsf{Reduction}(\mathcal{A}), Y_{t+1}$ is drawn in line 18 from $R_{t+1}(u, \varepsilon_{t+1}), u \sim Q_{i,t+1}$, as desired.

- If $s_i^{t+1} = 1$, then $\mathsf{Reduction}(\mathcal{A})$ uses Lemma 3.5 to write $R_{t+1}(\cdot)$ as a mixture. Hence if we sample from the mixture with input $u \sim Q_{i,t+1}$, we sample from $R_{t+1}(u)$, which is the desired sampling distribution. To see that $\mathsf{Reduction}(\mathcal{A})$ does sample from the target, we need only show that Y_{t+1} drawn in line 13 is sampled from $\tilde{R}_t(u)$ where $u \sim Q_{i,t+1}$. This is true by Lemma 3.2.

4. Sample Complexity: Here we bound the expected sample complexity, deferring the high probability bound to Section A.3 in the Appendix. Let N_i be the number of fresh samples drawn over all rounds t where $i^t = i$, i.e. the number of samples drawn when simulating follow-up queries to i. Let N_i^t be the number of samples drawn during rejection sampling in round t; we imagine that regardless of the coin-flip in line 11 of the pseudocode of Reduction, N_i^t is always drawn. Then the total number of samples is $N_i = \sum_{t=1}^{T} \gamma_t N_i^t$. (Note that for simplicity, we are summing over all rounds T, since equivalently we may imagine that each user is given a local randomizer at each round, with privacy cost 0 in any round in which $i_t \neq i$.) Then by Lemma 3.2

$$\mathbb{E}[N_i] = \sum_{t=1}^{T} \gamma_t 2e^{\varepsilon} = \frac{2e^{\varepsilon}}{1-e^{-\varepsilon}} \sum_{t=1}^{T} (1 - e^{-\varepsilon_t}).$$

Since $1 - x \leq e^{-x}$, we get that $1 - \varepsilon_t \leq e^{-\varepsilon_t}$ and so $1 - e^{-\varepsilon_t} \leq \varepsilon_t$. Hence

$$\mathbb{E}[N_i] \leq \frac{2e^{\varepsilon}}{1 - e^{-\varepsilon}} \sum_{t=1}^{T} \varepsilon_t \leq \left(\frac{2e^{\varepsilon} \cdot \varepsilon}{1 - e^{-\varepsilon}} \right) k.$$

Summing over i, and including the at most n samples drawn in line 18 bounds the expected sample complexity by $((\frac{2e^{\varepsilon}\cdot\varepsilon}{1-e^{-\varepsilon}})k + 1)n$, as desired. □

4 Separating Full and Sequential Interactivity

We now prove that our reduction in Section 3 is tight in the sense that any generic reduction from a fully interactive protocol to a sequentially interactive protocol must have a sample complexity blowup of $\tilde{\Omega}(k)$ when applied to a k-compositional protocol. Specifically, we define a family of problems such that for every k, there is a fully interactive k-compositional protocol that can solve the problem with sample complexity $n = n(k)$, but such that *any* sequentially interactive protocol solving the problem must have sample complexity $\tilde{\Omega}(k \cdot n)$.

Informally, the family of problems (Multi-Party Pointer Jumping, or $\mathcal{MPJ}(d)$) we introduce is defined as follows. An *instance* of $\mathcal{MPJ}(d)$ is given by a complete tree of depth d. Every vertex of the tree is labelled by one of its children. By following the labels down the tree, starting at the root, an instance defines a unique root-to-leaf path. Given an instance of $\mathcal{MPJ}(d)$, the data distribution is defined as follows: to sample a new user, first select a level of the tree uniformly $\ell \in [d]$ at random, and provide that user with the vertex-labels corresponding to level ℓ (note that fixing an instance of the problem, every user corresponding to the same level of the tree has the same data). The problem we wish to solve privately is to identify the unique root to leaf path specified by the instance.

We first show that there is a fully interactive protocol which can solve this problem with sample complexity $n = \tilde{O}(d^2/\varepsilon^2)$. The protocol is k-compositional for $k = \Theta(d)$. Roughly speaking, the protocol works as follows: it identifies the path one vertex at a time, starting from the root, and proceeding to the leaf, in d rounds. In each round, given the most recently identified vertex v_i in level ℓ, it attempts to identify the child that vertex v_i is labelled with. It queries every user with the same local randomizer, which asks them to use randomized response to identify the labelled child of v_i if their data corresponds to level ℓ, and to respond with a uniformly random child otherwise (recall that the level that a user's data corresponds to is itself private, and hence is not known to the protocol). Since there are roughly $\tilde{\Theta}(\sqrt{n}/\varepsilon^2)$ users with relevant data, out of n users total, it is possible to identitify the child in question subject to local differential privacy. Although every user applies an ε-local randomizer d times in sequence, because each user's data corresponds to only a single level in the tree, the protocol is still $(\varepsilon, 0)$-locally private. Note that this privacy analysis mirrors the "histogram" structure of the non-compositional protocol in Example 2.2.

Informally, the reason that any sequentially interactive protocol must have sample complexity that is larger by a factor of d, is that even to identify the child of a single vertex in the local model, $\Omega(d^2/\varepsilon^2)$ datapoints are required (this is exactly what our randomized response protocol achieves). But a sequentially interactive protocol cannot re-use these datapoints across levels of the tree, and so must expend $\Omega(d^2/\varepsilon^2)$ samples for *each* of the d levels of the tree. This intuition is formalized in a delicate and technical induction on the depth of the tree, using information theoretic tools to bound the success probability of any protocol as a function of its sample complexity. The precise definition of $\mathcal{MPJ}(d)$ is somewhat more complicated, in which half of the weight on the underlying distribution is assigned to "level 0" dummy agents whose purpose is to break correlations between levels of the tree in the argument.

4.1 The Multi-Party Pointer Jumping Problem

We now formally define the *Multi-party Pointer Jumping* ($\mathcal{MPJ}$) problem.

Definition 4.1. *Given integer parameter $d > 1$, an instance of Multi-party Pointer Jumping $\mathcal{MPJ}(d)$ is defined by a vector $Z = Z_1 \circ \cdots \circ Z_d$, a concatenation of d vectors of increasing length. Letting $s = d^4$, for each $i \in [d]$ Z_i is a vector of s^{i-1} integers in $\{0, 1, \ldots, s-1\}$. For each Z_i, $Z_{i,j}$ is its j^{th} coordinate.*

Viewed as a tree, Z is a complete s-ary tree of depth d where each $Z_{i,j}$ marks a child of the j-th vertex at depth i. $P = P(Z)$ then denotes the vector of d integers representing the unique root to leaf path *down this tree through the children marked by Z. Formally, P is defined in a recursive way: $P_1 = Z_{1,1}$, ...,$P_i = Z_{i,P_1 \cdot s^{i-1}+P_2 \cdot s^{i-2}+\cdots+P_{i-1}+1}$,...,$P_d = Z_{d,P_1 \cdot s^{d-1}+P_2 \cdot s^{d-2}+\cdots+P_{d-1}+1}$.*

Finally, an instance $\mathcal{MPJ}(d)$ defines a data distribution $\mathcal{D}$. For each $x \sim \mathcal{D}$, with probability $1/2$, $x = (0, \emptyset)$ is a "dummy datapoint", and with the remaining probability $x = (\ell, Z_\ell)$ where ℓ is a level drawn uniformly at random from $[d]$. A protocol solves $\mathcal{MPJ}(d)$ if it recovers P using samples from $\mathcal{D}$.

A graphical representation of $\mathcal{MPJ}(d)$ where $s = 2$ appears in Figure 1 (we set $s = 2$ in this figure for easier graphical representation).

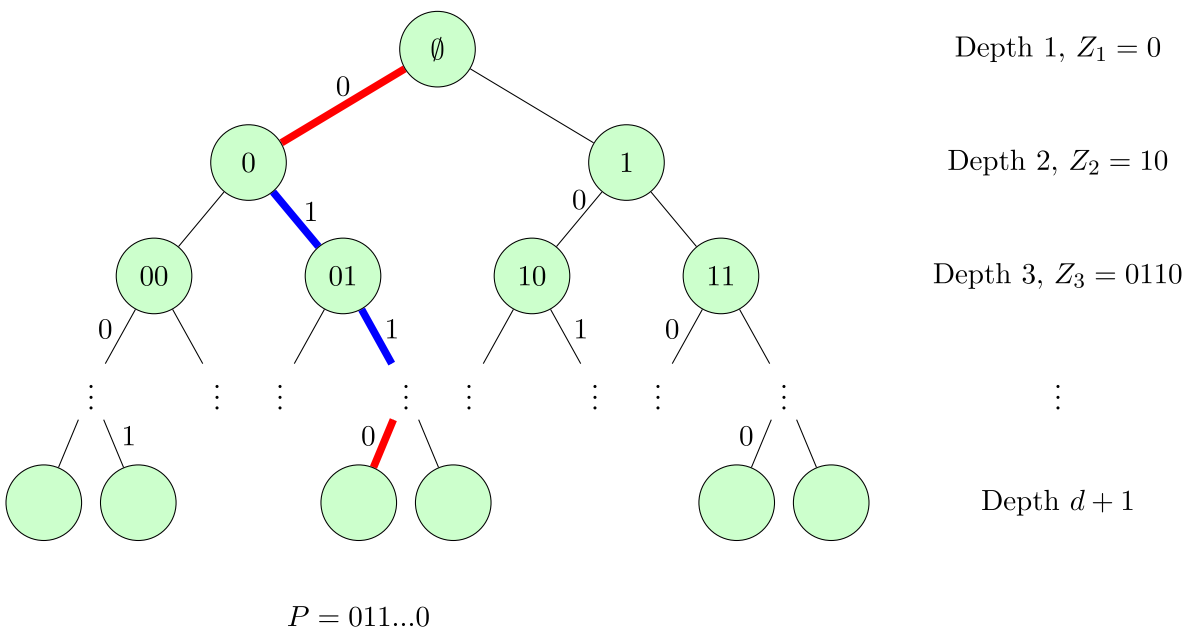

Figure 1: Multi-party Pointer Jumping

Algorithm 5 A fully interactive $(\varepsilon, 0)$-locally private protocol for $\mathcal{MPJ}(d)$

Divide users into $u = \lceil \log(s)/\log(2) \rceil$ groups each of $m = 512d^2 \log(d) \cdot \frac{(e^\varepsilon+1)^2}{(e^\varepsilon-1)^2}$ users.
Initialize $Q \leftarrow 0$
for $r = 1, 2, \ldots, d$ **do**
  $Q_r \leftarrow 0$
  for each group $g = 1, 2, \ldots, u$ **do**
    for each user $i = 1, 2, \ldots, m$ **do**
      $\ell_i \leftarrow$ level of user x_i
      if $\ell_i = r$ **then**
        $b_{i,r} \leftarrow g$-th bit of binary representation of $Z_{r,Q+1}$
        User i publishes randomized response $y_i \sim \mathsf{RR}(b_{i,r}, \varepsilon)$
      else
        User i publishes $y_i \sim \mathsf{Ber}(0.5)$
      end if
    end for
    g-th bit of $Q_r \leftarrow$ majority bit of $\{y_i\}_{i=1}^m$
  end for
  $Q \leftarrow s \cdot Q + Q_r$
end for
Output $Q_1 \circ \cdots \circ Q_d$

4.2 An Upper Bound for Fully Interactive Mechanisms

Theorem 4.1. *There exists a fully interactive $(\varepsilon, 0)$-locally private protocol (Algorithm 5) with sample complexity $n = O(d^2 \log^2(d)(e^\varepsilon + 1)^2/(e^\varepsilon - 1)^2)$ that, on any instance Z of $\mathcal{MPJ}(d)$, correctly identifies $P(Z)$ with probability at least $1 - 1/d$.*

Proof. First, it is easy to check that the total sample complexity of Algorithm 5 is $n = u \cdot m = O\left(d^2 \log^2(d) \cdot \frac{(e^\varepsilon+1)^2}{(e^\varepsilon-1)^2}\right)$. Privacy follows from the same line of logic used in Example 2.2: each agent sends d bits in total, and at most one of these bits is not sampled uniformly at random. Therefore, the probability of an agent sending any binary string of d bits is bounded between $\frac{1}{2^{d-1}} \cdot \frac{1}{1+e^\varepsilon}$ and $\frac{1}{2^{d-1}} \cdot \frac{e^\varepsilon}{1+e^\varepsilon}$, for any datapoint that they might hold. Algorithm 5 is therefore $(\varepsilon, 0)$-locally private.

It remains to prove correctness. We first show that each group contains enough users from each level. For each group $g \in [u]$, define $X_{i,g,r}$ to be 1 if the i-th user in group g has level r and 0 otherwise. By definition, for any $r \in [d]$, $\mathbb{P}[X_{i,g,r} = 1] = 1/(2d)$. Therefore we have $\mathbb{E}[\sum_{i=1}^m X_{i,g,r}] = m/(2d)$, and by a Chernoff bound

$$\mathbb{P}\left[\sum_{i=1}^m X_{i,g,r} < m/(4d)\right] \leq \exp\left(-\frac{m}{16d}\right) \leq 1/(d^4).$$

Define W to be the event that for every $r \in [d]$ and $g \in [u]$, there are at least $m/(4d)$ users in group g with level r. By a union bound, we know $\mathbb{P}[W] \geq 1 - (ud)/d^4 \geq 1 - 1/d^2$, so with high probability we have enough users in each level in each group.

We now analyze the quantities Q_r. For each $r \in [d]$, we want to show

$$\mathbb{P}[Q_r = P_r | Q_1 = P_1, ..., Q_{r-1} = P_{r-1}, W] \geq 1 - 1/d^3,$$

i.e. that the output Q actually matches P. Conditioning on $Q_1 = P_1, \dots, Q_{r-1} = P_{r-1}$ and W, $Z_{r,Q+1} = P_r$. Define $Y_{i,g,r}$ to be 1 if the bit sent by the i-th user of group g is equal to the j-th bit of P_r and 0 otherwise. If the i-th user has level r then they send their bit using randomized response and $\mathbb{P}\left[Y_{i,g,r} = 1\right] = \frac{e^\varepsilon}{e^\varepsilon+1}$. If the i-th user's level is not r then they send a uniform random bit $\mathbb{P}\left[Y_{i,g,r} = 1\right] = 1/2$. Since we conditioned on W, there are at least $m/(4d)$ users in group g with level r. Thus

$$\mathbb{E}\left[\sum_{i=1}^{m} Y_{i,g,r}\right] = \frac{m}{4d} \cdot \frac{e^\varepsilon}{e^\varepsilon + 1} + \left(m - \frac{m}{4d}\right) \cdot \frac{1}{2}. \tag{1}$$

Then we have

$$\begin{aligned}
&\mathbb{P}\left[Q_r, P_r \text{ have the same } g\text{-th bit} | Q_1 = P_1, ..., Q_{r-1} = P_{r-1}, W\right] \\
=&\mathbb{P}\left[\sum_{i=1}^{m} Y_{i,g,r} > \frac{m}{2}\right] \\
\geq&\mathbb{P}\left[\sum_{i=1}^{m} Y_{i,g,r} > \mathbb{E}\left[\sum_{i=1}^{m} Y_{i,g,r}\right] + \frac{m}{2} - \frac{m}{4d} \cdot \frac{e^\varepsilon}{e^\varepsilon + 1} - \left(m - \frac{m}{4d}\right) \cdot \frac{1}{2}\right] && \text{(Equation 1)} \\
\geq&\mathbb{P}\left[\sum_{i=1}^{m} Y_{i,g,r} > \mathbb{E}\left[\sum_{i=1}^{m} Y_{i,g,r}\right] - \frac{m}{8d} \cdot \frac{e^\varepsilon - 1}{e^\varepsilon + 1}\right] \\
\geq&1 - \exp\left(-\frac{1}{2m} \cdot \left(\frac{m}{8d} \cdot \frac{e^\varepsilon - 1}{e^\varepsilon + 1}\right)^2\right) && \text{(Chernoff bound)} \\
=&1 - \exp\left(-m \cdot \frac{1}{128d^2} \cdot \frac{(e^\varepsilon - 1)^2}{(e^\varepsilon + 1)^2}\right) \\
\geq&1 - \exp(-4\log(d)) = 1 - 1/d^4.
\end{aligned}$$

Union bounding over all u groups yields

$$\mathbb{P}\left[Q_r = P_r | Q_1 = P_1, ..., Q_{r-1} = P_{r-1}, W\right] \geq 1 - u/d^4 \geq 1 - 1/d^3.$$

Putting this all together, Algorithm 5 outputs $P(Z)$ with probability at least

$$\begin{aligned}
\mathbb{P}\left[Q_1 = P_1, ..., Q_d = P_d\right] &\geq \mathbb{P}\left[W\right] \cdot \mathbb{P}\left[Q_1 = P_1, ..., Q_d = P_d | W\right] \\
&\geq \mathbb{P}\left[W\right] \prod_{r=1}^{d} \mathbb{P}\left[Q_r = P_r | Q_1 = P_1, ..., Q_{r-1} = P_{r-1}, W\right] \\
&\geq (1 - 1/d^2)(1 - 1/d^3)^d \\
&\geq 1 - 1/d.
\end{aligned}$$

□

Note that Algorithm 5 is k-compositional only for $k \geq \Omega(d)$. The lower bound that we prove next (Theorem 4.2) shows that any sequentially interactive protocol for the same problem must have a larger sample complexity by a factor of $\tilde{\Omega}(d) = \tilde{\Omega}(k)$, showing that in general, the sample-complexity dependence that our reduction (Theorem 3.6) has on k cannot be improved.

4.3 A Lower Bound for Sequentially Interactive Mechanisms

We prove our lower bound for sequentially interactive $(\varepsilon, 0)$-locally private protocols. As previous work [9, 12] has established that $(\varepsilon, 0)$- and (ε, δ)-local privacy are approximately equivalent for reasonable parameter ranges, our lower bound also holds for sequentially interactive (ε, δ)-locally private protocols. For an extended discussion of this equivalence, see Section 5.1.2.

Theorem 4.2. *Let $\mathcal{A}$ be a sequentially interactive $(\varepsilon, 0)$-locally private protocol that, for every instance Z of $\mathcal{MPJ}(d)$, correctly identifies $P(Z)$ with probability $\geq 2/3$. Then $\mathcal{A}$ must have sample complexity $n \geq d^3/(216(e^\varepsilon - 1)^2 \log(d))$.*

Proof. We will prove that any sequentially interactive $(\varepsilon, 0)$-locally private protocol with $n = d^3/(216(e^\varepsilon - 1)^2 \log(d))$ samples fails to solve $\mathcal{MPJ}(d)$ correctly with probability $> 1/3$ when Z is sampled uniformly randomly. This is a distributional lower bound which is only stronger than the theorem statement (a worst case lower bound). For notational simplicity,we assume in this argument that all local randomizers have discrete message spaces. However, this assumption is without loss of generality and can be removed (e.g. using the rejection sampling technique from Bassily and Smith [5]).

We will prove our lower bound even for protocols to which we "reveal" some information about the hidden instance Z and users' inputs to the protocol and users. This only makes our lower bound stronger, as the mechanism can ignore this information if desired. Before the protocol starts, each user i publishes a quantity R_i. $R_i = \ell_i$, user i's level, if $\ell_i \neq 0$ (i.e., user i is not a "dummy" user). Otherwise R_i is set to be $\lfloor \frac{i-1}{n/d} \rfloor + 1$. At a high level, we reveal these $\{R_i\}_{i=1}^n$ to break the dependence between Z_i's during the execution of the protocol (see Claim 4.3 for a formalization of this intuition). Throughout the proof and its claims, we fix realizations $R_1 = r_1, R_2 = r_2, \ldots, R_n = r_n$. We will show that even given such $r_1, ..., r_n$, any sequentially interactive $(\varepsilon, 0)$-locally private protocol with n users fails with probability more than 1/3.

For each $i \in [n]$, denote by Π_i the message — i.e., portion of the transcript — sent by user i via their local randomizer. Note that there is at most one such message since the protocol is sequentially interactive. We begin with a result about how conditioning on messages and revealed values affects the distribution of Z.

Claim 4.3. *Suppose $Z_1, ..., Z_d$ are sampled from a product distribution. Conditioned on the messages $\Pi_1, ..., \Pi_i$ of the first i users and the revealed values $R_1, \ldots, R_n$, $Z_1, ..., Z_d$ are still distributed according to a product distribution.*

Proof. We proceed via induction on the number of messages i. The base case $i = 0$ is immediate from the assumption. Now suppose the statement of the claim is true for $i-1$. Use $\mathcal{D}_1 \times \mathcal{D}_2 \times \cdots \mathcal{D}_d$ to denote the product distribution of $Z_1, ..., Z_d$ conditioned on $\Pi_1, ..., \Pi_{i-1}$ and $R_1, \ldots, R_n$ (all quantities that follow are conditioned on $R_1, \ldots, R_n$, and so for notational simplicity we elide the explicit conditioning).

Since the protocol is sequentially interactive, conditioned on $\Pi_1, ..., \Pi_{i-1}$, Π_i depends only on Z_{r_i}, user i's internal randomness, and their level ℓ_i (recall that when $r_i = \lfloor \frac{i-1}{n/d} \rfloor + 1$, it may be that $\ell_i = 0$ or $\ell_i = r_i$). Therefore, conditioned on $\Pi_1, ..., \Pi_i$, $Z_1, ..., Z_d$ distribute as

$$\mathcal{D}_1 \times \mathcal{D}_2 \times \cdots \times (\mathcal{D}_{r_i} | \Pi_i) \times \cdots \times \mathcal{D}_d,$$

a product distribution. □

We also use induction to prove the overall theorem. It has d steps. For step $\ell \in [d]$, we let Δ_ℓ be the following set of distributions on Z.

Definition 4.2 (Δ_ℓ)**.** *For each $\ell \in [d]$, the set Δ_ℓ is composed of distributions $\mathcal{D}$ on Z such that*

1. *$\mathcal{D}$ is a product distribution on $Z_1, ..., Z_d$,*
2. *for each $i = 1, ..., d-\ell$, Z_i is deterministically fixed to be z_i, and*
3. *since $Z_1, ..., Z_{d-\ell}$ are fixed, by the definition of $\mathcal{MPJ}$, $P_1, ..., P_{d-\ell}$ are also fixed to some $p_1, ..., p_{d-\ell}$. The marginal distribution on $Z_{|p_1,...,p_{d-\ell}}$ is the uniform distribution.*

In the induction step, we consider locally private sequentially interactive protocols with fewer users. The idea is that for any sequentially interactive $(\varepsilon, 0)$-locally private protocol of n users, if we fix the messages of the first i users, then what remains is a sequentially interactive $(\varepsilon, 0)$-locally private protocol on $n-i$ users. Accordingly, we want to lower bound the failure probability of this remaining protocol. More concretely:

Inductive Statement Any sequentially interactive $(\varepsilon, 0)$-locally private protocol with $n \cdot \frac{\ell}{d}$ users (the $\left(n \cdot \frac{d-\ell}{d} + 1\right)$-th user to the n-th user) fails to solve $\mathcal{MPJ}(d)$ correctly with probability at $> 2/3 - \ell/(3d)$ when Z is sampled from a distribution in Δ_ℓ.

It will be slightly more convenient to establish the inductive case first and then establish the base case second.

Induction step ($\ell > 1$): Assume the above statement is true for $\ell - 1$.

In this induction, let $\mathcal{A}$ be a sequentially interactive $(\varepsilon, 0)$-locally private protocol with $n \cdot \frac{\ell}{d}$ users and let $\mathcal{D}$ be the distribution generating Z before $\mathcal{A}$ starts. Let Π be the messages sent by the first n/d users of $\mathcal{A}$ (the $\left(n \cdot \frac{d-\ell}{d} + 1\right)$-th user to the $\left(n \cdot \frac{d-\ell+1}{d}\right)$-th user) and let $\mathcal{A}_\pi$ be the sequentially interactive $(\varepsilon, 0)$-locally private protocol with $n \cdot \frac{\ell-1}{d}$ users conditioned on $\Pi = \pi$. For notational convenience, define $n_\ell = n \cdot \frac{d-\ell}{d}$, $\Pi_{<i} = \Pi_{n_\ell+1}, ..., \Pi_{i-1}$ and $\Pi_{\leq i} = \Pi_{n_\ell+1}, ..., \Pi_i$.

For each prefix of messages, π, let $\mathcal{D}'(\pi)$ be some mixture of distributions in $\Delta_{\ell-1}$ (to be specified later). By the induction hypothesis on $\ell - 1$,

$$\mathbb{P}_{Z\sim\mathcal{D}'(\pi)}\left[\mathcal{A}_\pi \text{ outputs } P(Z)\right] < 1/3 + (\ell-1)/(3d).$$

Thus

$$\begin{aligned}\mathbb{P}_{Z\sim\mathcal{D}}\left[\mathcal{A} \text{ outputs } P(Z)\right] &= \sum_\pi \mathbb{P}\left[\Pi = \pi\right] \cdot \mathbb{P}_{Z\sim(\mathcal{D}|\Pi=\pi)}\left[\mathcal{A}_\pi \text{ outputs } P(Z)\right] \\ &\leq \sum_\pi \mathbb{P}\left[\Pi = \pi\right] \cdot \left(\mathbb{P}_{Z\sim\mathcal{D}'(\pi)}\left[\mathcal{A}_\pi \text{ outputs } P(Z)\right] + \|(\mathcal{D}|\Pi = \pi) - \mathcal{D}'(\pi)\|_1\right) \\ &< 1/3 + (\ell-1)/(3d) + \sum_\pi \mathbb{P}\left[\Pi = \pi\right] \cdot \|(\mathcal{D}|(\Pi = \pi)) - \mathcal{D}'(\pi)\|_1.\end{aligned}$$

It therefore suffices to show that

$$\sum_\pi \mathbb{P}\left[\Pi = \pi\right] \cdot \|(\mathcal{D}|(\Pi = \pi)) - \mathcal{D}'(\pi)\|_1 \leq 1/(3d).$$

We show this via a sequence of three claims (Claims 4.4, 4.6, and 4.8), where $\mathcal{D}'(\pi)$ is defined in Claim 4.8.

First we define some notation for the path we need to reason about. Since $\mathcal{D} \in \Delta_\ell$, by the definition of Δ_ℓ we know that for $Z \sim \mathcal{D}$, the first $d-\ell$ levels of the tree $Z_1, ..., Z_{d-\ell}$ deterministically take fixed values $z_1, ..., z_{d-\ell}$. Thus, the first $d-\ell$ nodes in the path $P_1, ..., P_{d-\ell}$ are also fixed to take particular values $p_1, ..., p_{d-\ell}$. For the induction step, we write $P = P_1, ..., P_{d-\ell+1}$ to denote the first $d-\ell+1$ vertices of the path. Since $P_{d-\ell+1}$ is the only value that is not fixed, and the path is through an s-ary tree, P can take on at most s different possible values and is determined by $Z_{d-\ell+1}$.

In the first claim, we show that after observing the messages sent by n/d agents, there remains substantial uncertainty about P.

Claim 4.4. *For $i \in \{n_\ell + 1, ..., n_\ell + n/d\}$,*

$$\sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\max_p \mathbb{P}\left[P = p | \Pi_{\leq i} = \pi_{\leq i}\right]\right) \leq 3/d^4.$$

Proof. Denoting by $\mathbf{1}_E$ the indicator function for event E,

$$\begin{aligned}
&\sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\max_p \mathbb{P}\left[P = p | \Pi_{\leq i} = \pi_{\leq i}\right]\right) \\
&\leq \sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\mathbf{1}_{\max_p \mathbb{P}\left[P=p|\Pi_{\leq i}=\pi_{\leq i}\right] > 2/s} \cdot 1 + \mathbf{1}_{\max_p \mathbb{P}\left[P=p|\Pi_{\leq i}=\pi_{\leq i}\right] \leq 2/s} \cdot \frac{2}{s}\right) \\
&\leq \frac{2}{s} + \sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\mathbf{1}_{\max_p \mathbb{P}\left[P=p|\Pi_{\leq i}=\pi_{\leq i}\right] > 2/s}\right) \\
&\leq \frac{2}{s} + \sum_p \sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\mathbf{1}_{\mathbb{P}\left[P=p|\Pi_{\leq i}=\pi_{\leq i}\right] > 2/s}\right). \qquad (2)
\end{aligned}$$

Now consider some specific p. We know that

$$\begin{aligned}
\mathbb{P}\left[P = p | \Pi_{\leq i} = \pi_{\leq i}\right] &= \frac{\mathbb{P}\left[P = p, \Pi_{\leq i} = \pi_{\leq i}\right]}{\mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right]} \\
&= \mathbb{P}\left[P = p\right] \cdot \frac{\mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i} | P = p\right]}{\mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right]} \qquad \text{(Bayes' rule)} \\
&= \frac{1}{s} \cdot \frac{\mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i} | P = p\right]}{\mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right]} \qquad \text{(uniformity of } P\text{)}.
\end{aligned}$$

For $j = n_\ell + 1, ..., i$, define random variable

$$X_j = \log\left(\frac{\mathbb{P}\left[\Pi_j | \Pi_{<j}, P = p\right]}{\mathbb{P}\left[\Pi_j | \Pi_{<j}\right]}\right).$$

As we ultimately want to upper bound the quantity in Equation 2, we now focus on bounding these X_j.

Recall that r_j is user j's level if that level is non-zero (i.e. user j is not a "dummy" user). Otherwise r_j is $d-\ell+1$ for $j = n_\ell+1, ..., n_\ell+n/d$. If $r_j \neq d-\ell+1$, by Claim 4.3, we know that conditioned on $\Pi_{<j}$, Π_j is independent of P. Therefore when $r_j \neq d-\ell+1$, $X_j = \log(1) = 0$.

If instead $r_j = d-\ell+1$, we know the level ℓ_j of the user j is 0 with probability $d/(d+1)$ and $d-\ell+1$ with probability $1/(d+1)$. If $\ell_j = 0$, then the user is a "dummy" and since the user has no private data about P, Π_j is independent of P conditioned on $\Pi_{<j}$. Call the input distribution of the j-th user q_j. Here, we recall Lemmas 3 and 4 from Duchi et al. [17] and restate a simplified version as Lemma 4.5

Lemma 4.5. *Let m_1 and m_2 be the output distributions of an $(\varepsilon, 0)$-local randomizer in a sequentially interactive protocol given, respectively, input distributions $q_j \mid \Pi_{<j}, P = p$ and $q_j \mid \Pi_{<j}$. Then*

$$\left|\log\left(\frac{m_1(z)}{m_2(z)}\right)\right| \leq \min(2, e^\varepsilon)(e^\varepsilon - 1) \cdot \|(q_j \mid \Pi_{<j}, P = p) - (q_j \mid \Pi_{<j})\|_{TV}.$$

We know that $\|(q_j|\Pi_{<j} = \pi_{<j}, P = p) - (q_j|\Pi_{<j} = \pi_{<j})\|_{TV} \leq 1/(d+1)$, as the difference stems from the event that $\ell_j = d-\ell+1$. Thus, by Lemma 4.5

$$|X_j| \leq 2(e^\varepsilon - 1)/(d+1) < 2(e^\varepsilon - 1)/d.$$

Next, we bound the conditional expectation of X_j:

$$\begin{aligned}
\mathbb{E}\left[X_j|\Pi_{<j} = \pi_{<j}\right] &= \sum_{\pi_j} \mathbb{P}\left[\Pi_j = \pi_j|\Pi_{<j} = \pi_{<j}\right] \cdot \log\left(\frac{\mathbb{P}\left[\Pi_j = \pi_j|\Pi_{<j} = \pi_{<j}, P = p\right]}{\mathbb{P}\left[\Pi_j = \pi_j|\Pi_{<j} = \pi_{<j}\right]}\right) \\
&= -D_{KL}\left((\Pi_j|\Pi_{<j} = \pi_{<j}, P = p) \,||\, (\Pi_j|\Pi_{<j} = \pi_{<j})\right) \leq 0.
\end{aligned}$$

Therefore $X_{n_\ell+1}, X_{n_\ell+1} + X_{n_\ell+2}, ..., X_{n_\ell+1} + \cdots + X_i$ form a supermartingale. Next, we use the above bounds on these X_j to control their sum using the Azuma-Hoeffding inequality:

$$\begin{aligned}
\mathbb{P}\left[X_{n_\ell+1} + \cdots + X_i > \log(2)\right] &\leq \exp\left(-\frac{\log^2(2)}{2(2(e^\varepsilon - 1)/d)^2(i - n_\ell)}\right) \\
&\leq \exp\left(-\frac{\log^2(2)}{2(2(e^\varepsilon - 1)/d)^2(n/d)}\right) \\
&\leq \frac{1}{d^8} = \frac{1}{sd^4}.
\end{aligned}$$

By the chain rule and Bayes' rule, we know

$$X_{n_\ell+1} + \cdots + X_i = \log\left(\frac{\mathbb{P}\left[\Pi_{\leq i}|P = p\right]}{\mathbb{P}\left[\Pi_{\leq i}\right]}\right) = \log\left(s \cdot \mathbb{P}\left[P = p|\Pi_{\leq i}\right]\right).$$

Therefore

$$\begin{aligned}
\sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\mathbf{1}_{\mathbb{P}\left[P = p|\Pi_{\leq i} = \pi_{\leq i}\right] > 2/s}\right) &= \sum_{\pi_{\leq i}} \mathbb{P}\left[\Pi_{\leq i} = \pi_{\leq i}\right] \cdot \left(\mathbf{1}_{s \cdot \mathbb{P}\left[P = p|\Pi_{\leq i} = \pi_{\leq i}\right] > 2}\right) \\
&= \mathbb{P}\left[X_{n_\ell+1} + \cdots + X_i > \log(2)\right] \\
&\leq \frac{1}{sd^4}.
\end{aligned}$$

Tracing the above inequality back through Equation 2, we have

$$\sum_{\pi_{\le i}} \mathbb{P}\left[\Pi_{\le i} = \pi_{\le i}\right] \cdot \left(\max_{e} \mathbb{P}\left[P = p | \Pi_{\le i} = \pi_{\le i}\right]\right) \le \frac{2}{s} + \sum_{p}\sum_{\pi_{\le i}} \mathbb{P}\left[\Pi_{\le i} = \pi_{\le i}\right] \cdot \left(\mathbf{1}_{\mathbb{P}\left[P=p|\Pi_{\le i}=\pi_{\le i}\right]>2/s}\right)$$
$$\le \frac{2}{s} + s \cdot \frac{1}{sd^4} = \frac{3}{d^4}.$$

□

In Claim 4.6, we bound the information Π contains about $Z_{|P}$ (for a primer on information theory, see Appendix B). Intuitively, by Claim 4.4 users have little information about P, and as a result they cannot know which potential subtree $Z_{|p}$ to focus their privacy budget on.

Claim 4.6.

$$\sum_{p} \mathbb{P}\left[P = p\right] \cdot I(\Pi; Z_{|p} | P = p) \le 1/(18d^2).$$

Proof. By the inductive hypothesis, Z is sampled from $\mathcal{D} \in \Delta_\ell$. Define $Z_{|<p}$ to be $Z_{|p_1,\dots,p_{d-\ell},0}, \dots, Z_{|p_1,\dots,p_{d-\ell},p_{d-\ell+1}-1}$. By the definition of Δ_l, we know $Z_{|<p}$ and $Z_{|p}$ are independent given P, so $I(Z_{|<p}; Z_{|p} | P = p) = 0$. Therefore by the chain rule for mutual information, we get

$$\begin{aligned} I(\Pi; Z_{|p} | P = p) &\le I(\Pi, Z_{|<p}; Z_{|p} | P = p) \\ &= I(Z_{|<p}; Z_{|p} | P = p) + I(\Pi; Z_{|p} | P = p, Z_{|<p}) \\ &= 0 + I(\Pi; Z_{|p} | P = p, Z_{|<p}). \end{aligned}$$

The main step of the proof of this claim is to compare $I(\Pi_i; Z_{|p} | P = p, \Pi_{<i} = \pi_{<i}, Z_{|<p})$ and $I(\Pi_i; Z_{|p} | \Pi_{<i} = \pi_{<i}, Z_{|<p})$. First, by Claim 4.3, conditioning on $\Pi_{<i} = \pi_{<i}$ induces a product distribution for $Z_1, \dots, Z_d$. We also know that (as mentioned in the proof of Claim 4.3) conditioned on $\Pi_{<i} = \pi_{<i}$, Π_i only depends on Z_{r_i}, the internal randomness of the user i, and their level ℓ_i. By item 3 in the definition of Δ_ℓ, P only depends on $Z_{d-\ell+1}$. We prove

$$I(\Pi_i; Z_{|p} | P = p, \Pi_{<i} = \pi_{<i}, Z_{|<p}) = I(\Pi_i; Z_{|p} | \Pi_{<i} = \pi_{<i}, Z_{|<p}).. \tag{3}$$

There are two cases depending on r_i.

- When $r_i \le d - \ell + 1$, user i either has $\ell_i \le d - \ell + 1$ or is a "dummy" user. Therefore, whether or not we condition on $P = p$, Π_i is independent of $Z_{|p}, Z_{|<p}$. Thus

$$I(\Pi_i; Z_{|p} | P = p, \Pi_{<i} = \pi_{<i}, Z_{|<p}) = 0 = I(\Pi_i; Z_{|p} | \Pi_{<i} = \pi_{<i}, Z_{|<p}).$$

- When $r_i > d - \ell + 1$, once we've conditioned on $\Pi_{<i} = \pi_{<i}$, additionally conditioning on $P = p$ does not change the joint distribution of $Z_{d-\ell+2}, \dots, Z_d$. This is because $P = P_1, \dots, P_{d-\ell+1}$ and by above conditioning on $\Pi_{<i} = \pi_{<i}$ induces a product distribution on $Z_1, \dots, Z_d$ (and in particular on $Z_{d-\ell+2}, \dots, Z_d$). It follows that conditioning on $P = p$ does not change the joint distribution of $Z_{|p}, Z_{|<p}, \Pi_i$. Thus

$$I(\Pi_i; Z_{|p} | P = p, \Pi_{<i} = \pi_{<i}, Z_{|<p}) = I(\Pi_i; Z_{|p} | \Pi_{<i} = \pi_{<i}, Z_{|<p}).$$

Putting things together, we have

$$
\begin{aligned}
&\sum_p \mathbb{P}\left[P=p\right] \cdot I(\Pi; Z_{|p}|P=p) \\
&\leq \sum_p \mathbb{P}\left[P=p\right] \cdot I(\Pi; Z_{|p}|P=p, Z_{|<p}) \\
&= \sum_p \sum_{i=n_\ell+1}^{n_\ell+n/d} \mathbb{P}\left[P=p\right] \cdot I(\Pi_i; Z_{|p}|P=p, \Pi_{<i}, Z_{|<p}) \\
&= \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \sum_p \mathbb{P}\left[P=p\right] \cdot \mathbb{P}\left[\Pi_{<i}=\pi_{<i}|P=p\right] \cdot I(\Pi_i; Z_{|p}|P=p, \Pi_{<i}=\pi_{<i}, Z_{|<p}) \\
&= \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \sum_p \mathbb{P}\left[\Pi_{<i}=\pi_{<i}\right] \cdot \mathbb{P}\left[P=p|\Pi_{<i}=\pi_{<i}\right] \cdot I(\Pi_i; Z_{|p}|P=p, \Pi_{<i}=\pi_{<i}, Z_{|<p}) \qquad \text{(Bayes' rule)} \\
&= \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \sum_p \mathbb{P}\left[\Pi_{<i}=\pi_{<i}\right] \cdot \mathbb{P}\left[P=p|\Pi_{<i}=\pi_{<i}\right] \cdot I(\Pi_i; Z_{|p}|\Pi_{<i}=\pi_{<i}, Z_{|<p}) \qquad \text{(Equation 3)} \\
&\leq \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \left(\sum_p \mathbb{P}\left[\Pi_{<i}=\pi_{<i}\right] \cdot I(\Pi_i; Z_{|p}|\Pi_{<i}=\pi_{<i}, Z_{|<p})\right) \cdot \left(\max_p \mathbb{P}\left[P=p|\Pi_{<i}=\pi_{<i}\right]\right) \\
&\leq \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \mathbb{P}\left[\Pi_{<i}=\pi_{<i}\right] \cdot I(\Pi_i; Z|\Pi_{<i}=\pi_{<i}) \cdot \left(\max_p \mathbb{P}\left[P=p|\Pi_{<i}=\pi_{<i}\right]\right). \qquad (4)
\end{aligned}
$$

We now bound $I(\Pi_i; Z|\Pi_{<i}=\pi_{<i})$ using Theorem 1 from Duchi et al. [17], simplified here as Lemma 4.7.

Lemma 4.7. *Let Π be the distribution over randomizer outputs for an ε-local randomizer with inputs drawn from a distribution family parametrized by $\mathcal{V}$. Then $I(\Pi;\mathcal{V}) \leq 4(e^\varepsilon-1)^2$.*

In particular, the proof of Lemma 4.7 implies that $I(\Pi_i; Z|\Pi_{<i}=\pi_{<i}) \leq 4(e^\varepsilon-1)^2$. We continue our chain of inequalities.

$$
\begin{aligned}
(4) &\leq \sum_{i=n_\ell+1}^{n_\ell+n/d} \sum_{\pi_{<i}} \mathbb{P}\left[\Pi_{<i}=\pi_{<i}\right] \cdot 4(e^\varepsilon-1)^2 \cdot \left(\max_p \mathbb{P}\left[P=p|\Pi_{<i}=\pi_{<i}\right]\right) \\
&\leq \frac{n}{d} \cdot (e^\varepsilon-1)^2 \cdot \frac{12}{d^4} \quad \text{(Claim 4.4)} \\
&\leq \frac{1}{18d^2}.
\end{aligned}
$$

□

In our last claim, we convert the bound on mutual information from Claim 4.6 into a bound on the L_1 distance between distributions.

Claim 4.8. *There exists a distribution $\mathcal{D}'(\pi)$ which is a mixture of distributions in $\Delta_{\ell-1}$ for each π such that*

$$\sum_{\pi} \Pr[\Pi = \pi] \cdot \|(\mathcal{D}|(\Pi = \pi)) - \mathcal{D}'(\pi)\|_1 \leq 1/(3d).$$

Proof. By the definition of mutual information in terms of KL-divergence,

$$I(\Pi; Z_{|p}|P = p) = D_{KL}\left(\mathbb{P}\left[\Pi, Z_{|p}|P = p\right] \,||\, \mathbb{P}\left[\Pi|P = p\right] \times \mathbb{P}\left[Z_{|p}|P = p\right]\right).$$

Next, by Pinsker's inequality,

$$\begin{aligned}
&\sum_{\pi, z_{|p}} \left|\mathbb{P}\left[\Pi = \pi, Z_{|p} = z_{|p}|P = p\right] - \mathbb{P}\left[\Pi = \pi|P = p\right] \times \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right]\right| \\
&\leq \sqrt{2D_{KL}\left(\mathbb{P}\left[\Pi, Z_{|p}|P = p\right] \,||\, \mathbb{P}\left[\Pi|P = p\right] \times \mathbb{P}\left[Z_{|p}|P = p\right]\right)}
\end{aligned}$$

so we may upper bound

$$\begin{aligned}
&\sum_{p} \mathbb{P}\left[P = p\right] \cdot \sum_{\pi, z_{|p}} \left|\mathbb{P}\left[\Pi = \pi, Z_{|p} = z_{|p}|P = p\right] - \mathbb{P}\left[\Pi = \pi|P = p\right] \times \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right]\right| \\
\leq &\sum_{p} \mathbb{P}\left[P = p\right] \cdot \sqrt{2D_{KL}\left(\mathbb{P}\left[\Pi, Z_{|p}|P = p\right] \,||\, \mathbb{P}\left[\Pi|P = p\right] \times \mathbb{P}\left[Z_{|p}|P = p\right]\right)} \\
= &\sum_{p} \mathbb{P}\left[P = p\right] \cdot \sqrt{2I(\Pi; Z_{|p}|P = p)} \qquad \text{(definition of mutual information)} \\
\leq &\sqrt{2\sum_{p} \mathbb{P}\left[P = p\right] \cdot 2I(\Pi; Z_{|p}|P = p)} \qquad \text{(Jensen's inequality and concavity of } \sqrt{\cdot}\text{)} \\
\leq &1/(3d) \qquad \text{(Claim 4.6)}.
\end{aligned}$$

On the other hand, we can also lower bound

$$\begin{aligned}
&\sum_{p} \mathbb{P}\left[P = p\right] \cdot \sum_{\pi, z_{|p}} \left|\mathbb{P}\left[\Pi = \pi, Z_{|p} = z_{|p}|P = p\right] - \mathbb{P}\left[\Pi = \pi|P = p\right] \times \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right]\right| \\
= &\sum_{p} \mathbb{P}\left[P = p\right] \cdot \sum_{\pi} \mathbb{P}\left[\Pi = \pi|P = p\right] \cdot \sum_{z_{|p}} \left|\mathbb{P}\left[Z_{|p} = z_{|p}|\Pi = \pi, P = p\right] - \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right]\right| \\
= &\sum_{\pi} \mathbb{P}\left[\Pi = \pi\right] \cdot \sum_{p} \mathbb{P}\left[P = p|\Pi = \pi\right] \cdot \sum_{z_{|p}} \left|\mathbb{P}\left[Z_{|p} = z_{|p}|\Pi = \pi, P = p\right] - \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right]\right| \\
= &\sum_{\pi} \mathbb{P}\left[\Pi = \pi\right] \cdot \sum_{p} \mathbb{P}\left[P = p|\Pi = \pi\right] \cdot \\
&\sum_{z} \left|\mathbb{P}\left[Z = z|\Pi = \pi, P = p\right] - \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right] \cdot \mathbb{P}\left[Z = z|\Pi = \pi, P = p, Z_{|p} = z_{|p}\right]\right| \\
\geq &\sum_{\pi} \mathbb{P}\left[\Pi = \pi\right] \cdot \sum_{z} \left|\mathbb{P}\left[Z = z|\Pi = \pi\right] - \right. \\
&\left. \sum_{p} \mathbb{P}\left[P = p|\Pi = \pi\right] \cdot \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right] \cdot \mathbb{P}\left[Z = z|\Pi = \pi, P = p, Z_{|p} = z_{|p}\right]\right| \qquad (5)
\end{aligned}$$

where the last equality comes from multiplying by

$$1 = \sum_z \mathbb{P}\left[Z = z \mid \Pi = \pi, P = p, Z_{|p} = z_{|p}\right]$$

and the inequality uses

$$\mathbb{P}\left[Z = z \mid \Pi = \pi\right] = \sum_p \mathbb{P}\left[Z = z \mid \Pi = \pi, P = p\right] \cdot \mathbb{P}\left[P = p \mid \Pi = \pi\right]$$

and the triangle inequality. With the preceding upper bound, the quantity in Equation 5 is $\leq 1/(3d)$.

Now, define $\mathcal{D}'(\pi)$ to be the distribution on Z such that for all z,

$$\mathbb{P}_{Z\sim\mathcal{D}'(\pi)}\left[Z = z\right] = \sum_p \mathbb{P}\left[P = p|\Pi = \pi\right] \cdot \mathbb{P}\left[Z_{|p} = z_{|p}|P = p\right] \cdot \mathbb{P}\left[Z = z|\Pi = \pi, P = p, Z_{|p} = z_{|p}\right].$$

Equivalently, $Z \sim \mathcal{D}'(\pi)$ is sampled through the following procedure: (1) sample P according to $P \mid \Pi = \pi$, (2) sample $Z_{|p}$ according to $Z_{|p} \mid P = p$, and (3) sample Z according to $Z \mid \Pi = \pi, P = p, Z_{|p} = z_{|p}$.

Noting that $\mathbb{P}_{Z\sim\mathcal{D}|(\Pi=\pi)}\left[Z = z\right] = \mathbb{P}\left[Z = z|\Pi = \pi\right]$ for all z, since the quantity in Equation 5 is $\leq 1/(3d)$ we get

$$\sum_\pi \Pr[\Pi = \pi] \cdot \|(\mathcal{D}|(\Pi = \pi)) - \mathcal{D}'(\pi)\|_1 \leq 1/(3d).$$

It remains to show that $\mathcal{D}'(\pi)$ is a mixture of distributions in $\Delta_{\ell-1}$; doing so will complete our proof of the original inductive step. We will show for any $z_1, ..., z_{d-\ell+1}$ such that $\mathbb{P}_{Z\sim\mathcal{D}'(\pi)}\left[Z_1, ..., Z_{d-\ell+1} = z_1, ..., z_{d-\ell+1}\right] \neq 0$, $\mathcal{D}'(\pi) \mid (Z_1, ..., Z_{d-\ell+1} = z_1, ..., z_{d-\ell+1})$ is a distribution in $\Delta_{\ell-1}$. Recalling that membership in $\Delta_{\ell-1}$ requires meeting three conditions, we verify these conditions below.

1. By Claim 4.3, we know $\mathcal{D}|(\Pi = \pi)$ is a product distribution on $Z_1, ..., Z_d$. It is easy to check that as $\mathcal{D}'(\pi)$ is sampled according to $\mathcal{D}|(\Pi = \pi)$, $\mathcal{D}'(\pi)$ is also a product distribution on $Z_1, ..., Z_d$, and after the conditioning, $\mathcal{D}'(\pi)|(Z_1, ..., Z_{d-\ell+1} = z_1, ..., z_{d-\ell+1})$ remains a product distribution on $Z_1, ..., Z_d$.

2. Since we draw the final Z conditioned on $Z_{|p} = z_{|p}$, Z_i is deterministically fixed for $i = 1, \ldots, d - \ell$.

3. First, note that the marginal distribution of $\mathcal{D}|(P = p)$ on $Z_{|p}$ is uniform since $D \mid (\Pi = \pi)$ induces a product distribution on $Z_1, \ldots, Z_d$, and conditioning on $P = p$ only fixes $Z_{\leq d-\ell+1}$ and leaves $Z_{d-\ell+2} \times \cdots Z_d$ as a product distribution. Thus

$$\mathbb{P}_{Z\sim\mathcal{D}'(\pi)|(Z_{\leq d-\ell+1}=z_{\leq d-\ell+1})}\left[Z_{|p} = z_{|p}\right] = \mathbb{P}\left[Z_{|p} = z_{|p} \mid P = p\right]$$

so the marginal distribution of $\mathcal{D}'(\pi)|(Z_1, ..., Z_{d-\ell+1} = z_1, ..., z_{d-\ell+1})$ on $Z_{|p}$ is also the uniform distribution. Therefore $\mathcal{D}'(\pi)|(Z_1, ..., Z_{d-\ell+1} = z_1, ..., z_{d-\ell+1})$ is a distribution in $\Delta_{\ell-1}$ and $\mathcal{D}'(\pi)$ is a mixture of distributions in $\Delta_{\ell-1}$.

$\square$

Base case ($\ell = 1$): We finally discuss the base case of our induction. Define $\mathcal{A}$, Π and P as in the induction step. Since the output of $\mathcal{A}$ is a function of Π,

$$\mathbb{P}\left[\mathcal{A} \text{ outputs } P(Z)\right] \leq \sum_{\pi} \mathbb{P}\left[\Pi = \pi\right] \cdot \max_{p} \mathbb{P}\left[P = p | \Pi = \pi\right].$$

Since Claim 4.4 also applies to the base case, we get

$$\mathbb{P}\left[\mathcal{A} \text{ outputs } P(Z)\right] \leq 3/d^4 < 1/3 < 1/3 + 1/(3d).$$

□

5 Hypothesis Testing

We now turn our attention to the role of interactivity in hypothesis testing. We first show that for the simple hypothesis testing problem, there exists a non-interactive $(\varepsilon, 0)$-LDP protocol that achieves optimal sample complexity. This result extends to the compound hypothesis testing case, when we make the additional assumption that the sets of distributions are convex and compact.

5.1 Simple Hypothesis Testing

Let P_0 and P_1 be two known distributions such that $\|P_0 - P_1\|_{TV} \geq \alpha$, and suppose one of P_0 and P_1 generates n i.i.d. samples $x_1, \ldots, x_n$. The goal in *simple hypothesis testing* is to determine whether the samples are generated by P_0 or P_1. The Neyman-Pearson lemma [29] establishes that the likelihood ratio test is optimal for this problem absent privacy, and recent work [10] extends this idea to give an optimal (up to constants) private simple hypothesis test in the centralized model of differential privacy. We recall a simple folklore non-interactive hypothesis test in the local model, and then prove that it is optimal even among the set of all fully interactive locally private tests.

5.1.1 (Folklore) Upper Bound

Consider the following simple variant $\mathcal{A}$ of the likelihood ratio test: each user i with input x_i outputs $\mathsf{RR}\left(\varepsilon\right) \arg\max_{j \in \{0,1\}} P_j(x_i)$. For $j \in \{0, 1\}$ let $\hat{N}_j$ denote the resulting count of responses and let $\hat{N}'_j = \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \cdot \left(\hat{N}_j - \frac{n}{e^\varepsilon - 1}\right)$ be the corresponding de-biased count. The analyst computes both quantities $\hat{N}'_j$ and outputs $P_{\arg\max_j \hat{N}'_j}$.

It is immediate that $\mathcal{A}$ is noninteractive and, since it relies on randomized response, satisfies $(\varepsilon, 0)$-local differential privacy. Since we can bound its sample complexity by simple concentration arguments, we defer the proof to Appendix A.2.

Theorem 5.1. *With probability at least* $2/3$*,* $\mathcal{A}$ *distinguishes between* P_0 *and* P_1 *given* $n = \Omega\left(\frac{1}{\varepsilon^2 \alpha^2}\right)$ *samples.*

5.1.2 A Lower Bound for Arbitrarily Adaptive (ε, δ)-Locally Private Tests

We now show that the folklore ε-private non-interactive test is optimal amongst all (ε, δ)-private fully interactive tests. First, combining (slightly modified versions of) Theorem 6.1 from Bun et al. [9] and Theorem A.1 in Cheu et al. [12], we get the following result[5]

[5] Bun et al. [9] and Cheu et al. [12] prove their results for noninteractive protocols. However, their constructions both rely on replacing a single (ε, δ)-local randomizer call for each user with an $(O(\varepsilon), 0)$-local randomizer call and

Algorithm 6 Locally Private Simple Hypothesis Tester $\mathcal{A}$

```
procedure NONINTERACTIVE PROTOCOL({x_i}_{i=1}^n)
    for i = 1 ... n do
        User i publishes y_i ← RR(arg max_{j∈{0,1}} P_j(x_i), ε)
    end for
    for j = 0, 1 do
        Analyst computes N̂_j ← |{y_i | y_i = j}|
        Analyst computes N̂'_j ← (e^ε+1)/(e^ε−1) · (N̂_j − n/(e^ε−1))
    end for
    Analyst outputs P_{arg max_j N̂'_j}
end procedure
```

Lemma 5.2. *Given $\varepsilon > 0$, $\delta < \min\left(\frac{\varepsilon\beta}{48n\ln(2n/\beta)}, \frac{\beta}{64n\ln(n/\beta)e^{7\varepsilon}}\right)$ and sequentially interactive (ε, δ)-locally private protocol $\mathcal{A}$, there exists a sequentially interactive $(10\varepsilon, 0)$-locally private protocol $\mathcal{A}'$ such that for any dataset U, $\|\mathcal{A}(U) - \mathcal{A}'(U)\|_{TV} \leq \beta$.*

Lemma 5.2 enables us to apply existing lower bound tools for ε-locally private protocols to (sequentially interactive) (ε, δ)-locally private protocols. At a high level, our proof relies on controlling the Hellinger distance between transcript distributions induced by an (ε, δ)-locally private protocol when samples are generated by P_0 and P_1. We borrow a simulation technique used by Braverman et al. [8] for a similar (non-private) problem and find that we can control this Hellinger distance by bounding the KL divergence between a simpler, *noninteractive* pair of transcript distributions. We accomplish this last step using existing tools from Duchi et al. [17].

Theorem 5.3. *Let $\|P_0 - P_1\|_{TV} = \alpha$ and let Π be an arbitrary (possibly fully interactive) (ε, δ)-locally private simple hypothesis testing protocol distinguishing between P_0 and P_1 with probability $\geq 2/3$ using n samples where $\varepsilon > 0$ and $\delta < \min\left(\frac{\varepsilon^3\alpha^2}{48n\ln(2n/\beta)}, \frac{\varepsilon^2\alpha^2}{64n\ln(n/\beta)e^{7\varepsilon}}\right)$. Then $n = \Omega\left(\frac{1}{\varepsilon^2\alpha^2}\right)$.*

Proof. Let $\Pi_{\vec{0}}, \Pi_{\vec{1}}$, and $\Pi_{\vec{e}_i}$ respectively denote the distribution over transcripts induced by protocol Π when samples are drawn from P_0, P_1, and $x_i \sim P_1$ but the remaining $x_{i'} \sim P_0$. Let h^2 denote the square of the Hellinger distance, $h^2(f,g) = 1 - \int_{\mathcal{X}} \sqrt{f(x)g(x)}dx$. We begin with Lemma 5.4, originally proven as Lemma 2 in Braverman et al. [8].

Lemma 5.4. $h^2(\Pi_{\vec{0}}, \Pi_{\vec{1}}) = O\left(\sum_{i=1}^n h^2(\Pi_{\vec{0}}, \Pi_{\vec{e}_i})\right)$.

Since our goal is now to bound these squared Hellinger distances, we will use a few facts collected below.

Fact 5.5. *For any distributions f, g, and h,*

1. $h^2(f,g) \leq 2(h^2(f,h) + h^2(h,g))$.
2. $h^2(f,g) \leq d_{TV}(f,g) \leq \sqrt{2}h(f,g)$.

proving that these randomizers induce similar output distributions. Since each user still makes a single randomizer call in sequential interactive protocols, essentially the same argument applies. For fully interactive protocols, a naive modification of the same result forces a stronger restriction on δ, roughly $\delta = \tilde{o}\left(\frac{\varepsilon\beta}{\max(n,T)}\right)$.

3. $h^2(f,g) \le \frac{1}{2} D_{KL}(f \,||\, g)$.

Choose an arbitrary term i of the sum in Lemma 5.4. Suppose we have user i simulate $\mathcal{A}$ using draws from P_0 for the inputs of other users and their input x_i for input i. Since Π is (ε,δ)-locally private, this simulation can be viewed as a single (ε,δ)-local randomizer applied to x_i. We can therefore use Lemma 5.2 to get a $(10\varepsilon, 0)$-local randomizer Π' inducing distributions $\Pi'_{\vec{0}}$ and $\Pi'_{\vec{e}_i}$ such that $\|\Pi'_{\vec{0}} - \Pi_{\vec{0}}\|_{TV} \le \varepsilon^2\alpha^2$ and $\|\Pi'_{\vec{e}_i} - \Pi_{\vec{e}_i}\|_{TV} \le \varepsilon^2\alpha^2$. Then,

$$\begin{aligned}
h^2\left(\Pi_{\vec{0}}, \Pi_{\vec{e}_i}\right) &\le 2(h^2\left(\Pi_{\vec{0}}, \Pi'_{\vec{e}_i}\right) + h^2\left(\Pi'_{\vec{e}_i}, \Pi_{\vec{e}_i}\right)) \\
&\le 4(h^2\left(\Pi_{\vec{0}}, \Pi'_{\vec{0}}\right) + h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right)) + 2h^2\left(\Pi'_{\vec{e}_i}, \Pi_{\vec{e}_i}\right) \\
&\le 4(\|\Pi_{\vec{0}} - \Pi'_{\vec{0}}\|_{TV} + h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right)) + 2\|\Pi'_{\vec{e}_i} - \Pi_{\vec{e}_i}\|_{TV} \\
&\le 6\varepsilon^2\alpha^2 + 4h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right)
\end{aligned}$$

where the first two inequalities follow from item 1 in Fact 5.5, the third inequality follows from item 2, and the last inequality follows from our use of Lemma 5.2 above.

It remains to bound $h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right)$. By item 3 in Fact 5.5, $4h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right) \le 2D_{KL}\left(\Pi'_{\vec{0}} \,||\, \Pi'_{\vec{e}_i}\right)$. Since the transcript distributions $\Pi'_{\vec{0}}$ and $\Pi'_{\vec{e}_i}$ can be simulated by noninteractive $(10\varepsilon, 0)$-local randomizers, we can apply Theorem 1 from Duchi et al. [17], restated for our setting as Lemma 5.6.

Lemma 5.6. *Let Q be an $(\varepsilon, 0)$-local randomizer and let P_0 and P_1 be distributions defined on common space $\mathcal{X}$. Let $x_0 \sim P_0$ and $x_1 \sim P_1$. Then*

$$D_{KL}(Q(x_0) \,||\, Q(x_1)) + D_{KL}(Q(x_1) \,||\, Q(x_0)) \le \min\{4, e^{2\varepsilon}\}(e^\varepsilon - 1)^2 \|P_0 - P_1\|_{TV}^2.$$

Thus

$$D_{KL}\left(\Pi'_{\vec{0}} \,||\, \Pi'_{\vec{e}_i}\right) + D_{KL}\left(\Pi'_{\vec{e}_i} \,||\, \Pi'_{\vec{0}}\right) = O(\varepsilon^2 \cdot \|P_0 - P_1\|_{TV}^2) = O\left(\varepsilon^2\alpha^2\right).$$

It follows that $h^2\left(\Pi'_{\vec{0}}, \Pi'_{\vec{e}_i}\right) = O(\varepsilon^2\alpha^2)$. Moreover, since our original choice of i was arbitrary, tracing back to Lemma 5.4 yields $h^2\left(\Pi_{\vec{0}}, \Pi_{\vec{1}}\right) = O(n\varepsilon^2\alpha^2)$. By Fact 5.5, $h^2\left(\Pi_{\vec{0}}, \Pi_{\vec{1}}\right) \ge \frac{1}{2}\|\Pi_{\vec{0}} - \Pi_{\vec{1}}\|_{TV}^2 = \Omega(1)$. Thus $n = \Omega\left(\frac{1}{\varepsilon^2\alpha^2}\right)$. ☐

5.2 Compound Hypothesis Testing

We now extend the reasoning of Section 5 to *compound* hypothesis testing. Here P_0 and P_1 are replaced by (disjoint) collections of discrete hypotheses H_0 and H_1 such that

$$\inf_{(P,Q) \in H_0 \times H_1} \|P - Q\|_{TV} \ge \alpha.$$

The goal is to determine whether samples are generated by a distribution in H_0 or one in H_1.

Theorem 5.7. *Let H_0 and H_1 be convex and compact sets of distributions over ground set X such that $\inf_{(P,Q)\in H_0 \times H_1} \|P - Q\|_{TV} \ge \alpha$. Then there exists noninteractive $(\varepsilon, 0)$-locally private protocol $\mathcal{A}$ that with probability at least $2/3$ distinguishes between H_0 and H_1 given $n = \Omega\left(\frac{1}{\varepsilon^2\alpha^2}\right)$ samples.*

Proof. Let X be the ground set for distributions in H_0 and H_1, and consider the two-player zero-sum game

$$\sup_{S \in \Delta(2^X)} \inf_{(P,Q) \in H_0 \times H_1} \mathbb{E}_{E \sim S}\left[P(E) - Q(E)\right].$$

Here, the sup player chooses a distribution over events, and the inf player chooses distributions in H_0 and H_1. We will use (a simplified version of) Sion's minimax theorem [31].

Lemma 5.8 (Sion's minimax theorem)**.** *For $f\colon A \times B \to \mathbb{R}$, if*

1. *for all $a \in A$ $f(a, \cdot)$ is continuous and concave on B,*
2. *for all $b \in B$ $f(\cdot, b)$ is continuous and convex on A, and*
3. *A and B are convex and A is compact,*

then

$$\sup_{b \in B} \inf_{a \in A} f(a, b) = \inf_{a \in A} \sup_{b \in B} f(a, b).$$

We first verify that the three conditions of Lemma 5.8 hold. Let

$$f(S, (P, Q)) = \mathbb{E}_{E \sim S}\left[P(E) - Q(E)\right].$$

Linearity of expectation implies that $f(\cdot, (P, Q))$ is linear in $\Delta(2^X)$ and $f(S, \cdot)$ is linear in $H_0 \times H_1$. Therefore conditions 1 and 2 hold. Moreover, since $\Delta(2^X)$ is convex and we assumed H_0 and H_1 to be convex and compact — properties which are both closed under Cartesian product — condition 3 holds as well. As a result,

$$\sup_{S \in \Delta(2^X)} \inf_{(P,Q) \in H_0 \times H_1} \mathbb{E}_{E \sim S}\left[P(E) - Q(E)\right] = \inf_{(P,Q) \in H_0 \times H_1} \sup_{S \in \Delta(2^X)} \mathbb{E}_{E \sim S}\left[P(E) - Q(E)\right] \geq \alpha$$

and there exists fixed distribution S over events such that for all $(P, Q) \in H_0 \times H_1$,

$$\mathbb{E}_{E \sim S}\left[P(E) - Q(E)\right] \geq \alpha.$$

This leads to the following hypothesis testing protocol $\mathcal{A}$: for each $i \in [n]$, user i computes $y_i = \mathbb{E}_{E \sim S}\left[\mathbf{1}_{x_i \in E}\right]$ and publishes $y_i + \mathsf{Lap}\left(\frac{1}{\varepsilon}\right)$. This protocol is immediately noninteractive, and since $y_i \in [0, 1]$, this protocol is $(\varepsilon, 0)$-locally private over $\{x_i\}_{i=1}^n$. Finally, by the same analysis used to prove Theorem 5.1 (replacing concentration of randomized responses with concentration of $\mathsf{Lap}(1)$ noise [11]) it distinguishes between H_0 and H_1 with probability at least 2/3 using $n = \Omega\left(\frac{1}{\varepsilon^2 \alpha^2}\right)$ samples. ∎

Since Theorem 5.3 still applies, this establishes that the above non-interactive protocol is also optimal.

A Appendix

A.1 Properties of Differential Privacy

Differentially private computations enjoy two nice properties:

Theorem A.1 (Post Processing [19]). *Let $A : \mathcal{X}^* \to \mathcal{O}$ be any (ε, δ)-differentially private algorithm, and let $f : \mathcal{O} \to \mathcal{O}'$ be any function. Then the algorithm $f \circ A : \mathcal{X}^n \to \mathcal{O}'$ is also (ε, δ)-differentially private.*

Post-processing implies that, for example, every *decision* process based on the output of a differentially private algorithm is also differentially private.

Theorem A.2 (Basic Composition [19]). *Let $A_1 : \mathcal{X}^* \to \mathcal{O}$, $A_2 : \mathcal{O} \times \mathcal{X}^* \to \mathcal{O}'$ be such that A_1 is $(\varepsilon_1, \delta_1)$-differentially private, and $A_2(o, \cdot)$ is $(\varepsilon_2, \delta_2)$-differentially private for every $o \in \mathcal{O}$. Then the algorithm $A : \mathcal{X}^* \to \mathcal{O}'$ defined as $A(x) = A_2(A_1(x), x)$ is $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differentially private.*

A.2 Proof of Theorem 5.1

Proof. For $j \in \{0, 1\}$, let N_j denote the (unknown) true count of 0 and 1 responses, i.e. $N_j = |\{x_i \mid \arg\max_{j' \in \{0,1\}} P_{j'}(x_i) = j\}|$. Then for both j, $\mathbb{E}\left[\hat{N}_j\right] = \frac{N_j(e^\varepsilon - 1) + n}{e^\varepsilon + 1}$. By a Chernoff bound, with high probability $|\hat{N}_j - \frac{N_j(e^\varepsilon-1)+n}{e^\varepsilon+1}| = O(\sqrt{n})$. Then since $N_j' = \frac{e^\varepsilon+1}{e^\varepsilon-1} \cdot \left(\hat{N}_j - \frac{n}{e^\varepsilon-1}\right)$ we get $|\hat{N}_j' - N_j| = O\left(\frac{e^{\varepsilon+1}}{e^\varepsilon-1} \cdot \sqrt{n}\right) = O\left(\frac{\sqrt{n}}{\varepsilon}\right)$. It is therefore sufficient that $\frac{1}{\varepsilon\sqrt{n}} = O(\alpha)$ to distinguish between P_0 and P_1, which implies the claim. □

A.3 High Probability Sample Complexity from Theorem 3.6

We first prove a multiplicative Azuma-Hoeffding Inequality which will drive the high probability bound.

Lemma A.3 (Multiplicative Azuma-Hoeffding Inequality). *Let $(\gamma_t)_{t=1}^T, \gamma_t \in [0, 1]$ a collection of dependent random variables, and let $(\mathcal{F}_t)_{t=1}^T$ a filtration such that $\sigma(\gamma_1, \ldots \gamma_{t-1}) \subset \mathcal{F}_{t-1}$. Suppose $\forall t, \mathbb{E}[\gamma_t \mid \mathcal{F}_{t-1}] \le \mu_t$. Then if w.p. 1, $\sum_t \mu_t \le \mu$, we have for any $\delta \in [e^{-3/4\mu}, 1]$:*

$$\mathbb{P}\left[\sum_{t=1}^T \gamma_t > \sqrt{3\mu \log(1/\delta)} + \mu\right] \le \delta$$

Proof. By convexity of $e^{l\gamma_t}, \gamma_t \in [0, 1], \mathbb{E}[\gamma_t \mid \mathcal{F}_{t-1}] \le \mu_t$, $\forall t, l$:

$$\mathbb{E}\left[e^{l\gamma_t} \mid \mathcal{F}_{t-1}\right] \le 1 + (e^l - 1)\mathbb{E}[\gamma_t | \mathcal{F}_{t-1}] \le 1 + (e^l - 1)\mu \le e^{(e^l-1)\mu_t}$$

If we define $S_j = \sum_{t=1}^j \gamma_t$, then:

$$\mathbb{E}\left[e^{lS_j}\right] = \mathbb{E}_{\mathcal{F}_{j-1}}\left[\mathbb{E}\left[e^{lS_j} \mid \mathcal{F}_{j-1}\right]\right] = \mathbb{E}_{\mathcal{F}_{j-1}}\left[e^{lS_{j-1}} \mid F_{j-1}\right]\mathbb{E}\left[e^{l\gamma_j} \mid \mathcal{F}_{j-1}\right] \le \mathbb{E}\left[e^{lS_{j-1}}\right] e^{(e^l-1)\mu_t}$$

Inducting on j, we have:

$$\mathbb{E}\left[e^{lS_T}\right] \le e^{(e^l-1)\sum_t \mu_t} \le e^{(e^l-1)\mu}$$

For $\varepsilon > 0$, taking $l = \log(1 + \varepsilon), a = (1 + \varepsilon)\mu$ and using Markov's inequality:

$$\mathbb{P}[S_T \ge a] \le e^{-(1+\varepsilon)\mu l}\mathbb{E}\left[e^{lS}\right] \le e^{-(1+\varepsilon)\mu l + \mu(e^l-1)} = e^{-\mu\phi(\varepsilon)},$$

where $\phi(z) = z - (1+z)\log(1+z)$. Since $\phi(z) \leq -z^2/3$ for $z \in [0, 3/2]$, we get:

$$\mathbb{P}\left[S_T \geq (1+\varepsilon)\mu\right] \leq e^{-\mu\varepsilon^2/3}$$

Setting $\varepsilon = \sqrt{\frac{3\log(1/\delta)}{\mu}}$ gives the desired bound. Note that the condition $\varepsilon \in [0, 3/2]$ forces $\delta \geq e^{-3/4\mu}$.

□

Proof. There are at most n users drawn in line 18 of Reduction, hence it suffices to bound with high probability the number of users drawn during rejection sampling steps in line 13. For a given user i drawn during a rejection sampling step, the sample complexity on rounds where i is selected can be written as $\sum_{t:i_t=i}^{T} \gamma_t N_t$, where $\gamma_t \sim \mathsf{Ber}(\frac{e^{-\varepsilon_t}-1}{e^{-\varepsilon}-1})$, $N_t \overset{ind}{\sim} \mathsf{Geom}(p_t)$ where $p_t \geq \frac{e^{-2\varepsilon}}{2}, \varepsilon_t \leq \varepsilon$ are random variables that depend on $\Pi_{<t}$. Hence the total sample complexity over the rejection sampling rounds can be written as:

$$S = \sum_{t=1}^{T} \gamma_t N_t$$

First consider $\sum_{t=1}^{T} \gamma_t$. Let $\mathcal{F}_t$ be the σ-algebra generated as $\mathcal{F}_t = \sigma(\Pi_{<t}, \varepsilon_t, (\gamma_l)_{l=1}^{t-1})$. Then $\mathbb{E}\left[\gamma_t \mid \mathcal{F}_{t-1}\right] = \frac{e^{-\varepsilon_t}-1}{e^{-\varepsilon}-1} = \mu_t$. Define $\mu = \sum_t \mu_t \leq \frac{nk\varepsilon}{1-e^{-\varepsilon}}$. Hence by Lemma A.3, with probability $1 - \delta/2$, for $\delta \geq 2e^{-\frac{3}{4}\mu}$:

$$\mathbb{P}\left[\sum_{t=1}^{T} \gamma_t > \sqrt{3\mu\log(2/\delta)} + \mu\right] \leq \frac{\delta}{2}$$

Let E_γ be the above event $\left\{\sum_t \gamma_t \leq \sqrt{3\mu\log(2/\delta)} + \mu\right\}$. Then for any t,

$$\mathbb{P}[S \geq t | E_\gamma] \leq \mathbb{P}[Z \geq t],$$

where $Z = \sum_{t=1}^{K} N'_t$, $K = \sqrt{3\mu\log(2/\delta)} + \mu$, and $N'_t \overset{iid}{\sim} \mathsf{Geom}(\frac{e^{-\varepsilon}}{2})$. Let $\mu' = \mathbb{E}[Z] = 2e^{\varepsilon}(\sqrt{3\mu\log(2/\delta)} + \mu)$. By Theorem 2.1 in [23] for any $t \geq 1$:

$$\mathbb{P}[Z \geq t\mu'] \leq e^{\frac{-e^{-\varepsilon}}{2}\mu'(t-1-\log t)}$$

Setting $t = 2(\frac{\log(2/\delta)2e^{\varepsilon}}{\mu'} + 1)$ gives $Z \leq 2(\log(2/\delta)2e^{\varepsilon} + \mu')$ with probability at least $1 - \delta/2$. Hence $\mathbb{P}\left[S \geq 2(\log(2/\delta)2e^{\varepsilon} + \mu') | E_\gamma\right] \leq \frac{\delta}{2}$. Finally,

$$\mathbb{P}\left[S \geq 2(\log(2/\delta)2e^{\varepsilon} + \mu'\right] \leq \mathbb{P}\left[S \geq 2(\log(2/\delta)2e^{\varepsilon} + \mu' | E_\gamma\right]\mathbb{P}\left[E_\gamma\right] + (1 - \mathbb{P}\left[E_\gamma\right]) \leq \frac{\delta}{2} + \frac{\delta}{2} = \delta$$

Substituting in the expression for μ' gives $S = O(nk + \sqrt{nk\log\frac{1}{\delta}})$ with probability $1 - \delta$, as desired.

□

B Information Theory

We briefly review some standard facts and definitions from information theory, starting with entropy. Throughout, our log is base e.

Definition B.1. *The* entropy *of a random variable X, denoted by $H(X)$, is defined as $H(X) = \sum_x \Pr[X = x] \log(1/\Pr[X = x])$, and the* conditional entropy *of random variable X conditioned on random variable Y is defined as $H(X|Y) = \mathbb{E}_y[H(X|Y = y)]$.*

Next, we can use entropy to define the mutual information between two random variables.

Definition B.2. *The* mutual information *between two random variables X and Y is defined as $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$, and the* conditional mutual information *between X and Y given Z is defined as $I(X;Y|Z) = H(X|Z) - H(X|YZ) = H(Y|Z) - H(Y|XZ)$.*

Fact B.1. *Let X_1, X_2, Y, Z be random variables, we have $I(X_1X_2;Y|Z) = I(X_1;Y|Z)+I(X_2;Y|X_1Z)$.*

Definition B.3. *The* Kullback-Leibler divergence *between two random variables X and Y is defined as $D_{KL}(X \,||\, Y) = \sum_x \Pr[X = x] \log(\Pr[X = x]/\Pr[Y = x])$.*

Fact B.2. *Let X, Y, Z be random variables, we have*

$$I(X;Y|Z) = \mathbb{E}_{x,z}[D_{KL}((Y|X = x, Z = z) \,||\, (Y|Z = z))].$$

Lemma B.3 (Pinsker's inequality)**.** *Let X, Y be random variables,*

$$\sqrt{2D_{KL}(X \,||\, Y)} \geq \sum_x |\Pr[X = x] - \Pr[Y = x]|$$

.

References

[1] Jayadev Acharya, Clément Canonne, Cody Freitag, and Himanshu Tyagi. Test without trust: Optimal locally private distribution testing. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

[2] Jayadev Acharya, Clément L Canonne, and Himanshu Tyagi. Inference under information constraints: Lower bounds from chi-square contraction. In *Conference on Learning Theory (COLT)*, 2019.

[3] Differential Privacy Team Apple. Learning with privacy at scale. Technical report, Apple, 2017.

[4] Borja Balle, James Bell, Adria Gascon, and Kobbi Nissim. The privacy blanket of the shuffle model. In *International Cryptology Conference (CRYPTO)*, 2019.

[5] Raef Bassily and Adam Smith. Local, private, efficient protocols for succinct histograms. In *Symposium on the Theory of Computing (STOC)*, 2015.

[6] Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Syposium on Operating System Principles (SOSP)*, 2017.

[7] Jaroslaw Błasiok, Mark Bun, Aleksandar Nikolov, and Thomas Steinke. Towards instance-optimal private query release. In *Symposium on Discrete Algorithms (SODA)*, 2019.

[8] Mark Braverman, Ankit Garg, Tengyu Ma, Huy L. Nguyen, and David P. Woodruff. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. In *Symposium on the Theory of Computing (STOC)*, 2016.

[9] Mark Bun, Jelani Nelson, and Uri Stemmer. Heavy hitters and the structure of local privacy. In *Symposium on Principles of Database Systems (PODS)*, 2018.

[10] Clément L. Canonne, Gautam Kamath, Audra McMillan, Adam Smith, and Jonathan Ullman. The structure of optimal private tests for simple hypotheses. In *Symposium on the Theory of Computing (STOC)*, 2018.

[11] T-H Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. *ACM Transactions on Information and System Security (TISSEC)*, 2011.

[12] Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via mixnets. In *Conference on Theory and Application of Cryptographic Techniques (EUROCRYPT)*, 2019.

[13] Rachel Cummings, Katrina Ligett, Kobbi Nissim, Aaron Roth, and Zhiwei Steven Wu. Adaptive learning with robust generalization guarantees. In *Conference on Learning Theory (COLT)*, 2016.

[14] Amit Daniely and Vitaly Feldman. Learning without interaction requires separation. In *Neural Information and Processing Systems (NeurIPS)*, 2019.

[15] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *Neural Information Processing Systems (NIPS)*, 2017.

[16] John Duchi and Ryan Rogers. Lower bounds for locally private estimation via communication complexity. 2019.

[17] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Local privacy and statistical minimax rates. In *Symposium on the Foundations of Computer Science (FOCS)*, 2013.

[18] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. Minimax optimal procedures for locally private estimation. *Journal of the American Statistical Association*, 2018.

[19] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography Conference (TCC)*, 2006.

[20] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Conference on Computer and Communications Security (CCS)*, 2014.

[21] Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *Symposium on Discrete Algorithms (SODA)*, 2019.

[22] Ankit Garg, Tengyu Ma, and Huy Nguyen. On communication cost of distributed statistical estimation and dimensionality. In *Neural Information Processing Systems (NIPS)*, 2014.

[23] Svante Janson. Tail bounds for sums of geometric and exponential variables. *arXiv e-prints*, art. arXiv:1709.08157, Sep 2017.

[24] Matthew Joseph, Aaron Roth, Jonathan Ullman, and Bo Waggoner. Local differential privacy for evolving data. In *Neural Information Processing Systems (NeurIPS)*, 2018.

[25] Matthew Joseph, Janardhan Kulkarni, Jieming Mao, and Zhiwei Steven Wu. Locally private gaussian estimation. In *Neural Information and Processing Systems (NeurIPS)*, 2019.

[26] Matthew Joseph, Jieming Mao, and Aaron Roth. Exponential separations in local differential privacy. In *Symposium on Discrete Algorithms (SODA)*, 2020.

[27] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 2011.

[28] Seth Neel, Aaron Roth, and Zhiwei Steven Wu. How to use heuristics for differential privacy. In *Symposium on the Foundations of Computer Science (FOCS)*, 2019.

[29] Jerzy Neyman and Egon Sharpe Pearson. Ix. on the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 1933.

[30] Ryan M. Rogers, Aaron Roth, Jonathan Ullman, and Salil Vadhan. Privacy odometers and filters: Pay-as-you-go composition. In *Neural Information Processing Systems (NIPS)*, 2016.

[31] Maurice Sion. On general minimax theorems. *Pacific Journal of mathematics*, 1958.

[32] Adam Smith, Abhradeep Thakurta, and Jalaj Upadhyay. Is interaction necessary for distributed private learning? In *Symposium on Security and Privacy (SP)*, 2017.

[33] Jonathan Ullman. Tight lower bounds for locally differentially private selection. *arXiv preprint arXiv:1802.02638*, 2018.

[34] Stanley L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 1965.