Distributed Uniformity Testing

Orr Fischer
Tel-Aviv University
Tel-Aviv, Israel
orrfischer@mail.tau.ac.il

Uri Meir
Tel-Aviv University
Tel-Aviv, Israel
urimeir@mail.tau.ac.il

Rotem Oshman
Tel-Aviv University
Tel-Aviv, Israel
roshman@tau.ac.il

ABSTRACT

In the *uniformity testing* problem, we are given access to samples from some unknown distribution μ on a fixed domain $\{1, .., n\}$, and our goal is to distinguish the case where μ is the uniform distribution from the case where μ is ϵ-*far* from uniform in L_1 distance. Centralized uniformity testing has been extensively studied, and it is known that $\Theta(\sqrt{n}/\epsilon^2)$ samples are necessary and sufficient.

In this paper we study *distributed uniformity testing*: in a network of k nodes, each node i has access to s_i samples from the underlying distribution μ. Our goal is to test uniformity, while minimizing the number of samples per node, as well as the running time. We consider several distributed models: the LOCAL model, the CONGEST model, and a 0-round model where nodes cannot communicate with each other at all. We give upper bounds for each model, and a lower bound for the 0-round model. The key to our results is analyzing the centralized uniformity-testing problem in an unusual error regime, for which we give new upper and lower bounds.

CCS CONCEPTS

• **Theory of computation → Streaming, sublinear and near linear time algorithms**; *Lower bounds and information complexity*; • **Mathematics of computing** → *Information theory*; • **Networks** → *Network algorithms*;

ACM Reference Format:
Orr Fischer, Uri Meir, and Rotem Oshman. 2018. Distributed Uniformity Testing. In *PODC '18: ACM Symposium on Principles of Distributed Computing, July 23–27, 2018, Egham, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3212734.3212772

1 INTRODUCTION

Suppose we have a distributed network tasked with monitoring some environment or random process. Each network node draws *random samples* from its environment, and together the nodes must raise an alarm if the system's state deviates significantly from normal. For example, we could have several routers drawing random samples from the traffic they route, and trying to detect a denial-of-service attack; or a sensor network monitoring temperatures at a manufacturing plant, with their measurements subject to Gaussian noise. In each case, we have a "normal" or "expected" distribution η, and an unknown distribution μ; the network nodes draw samples from μ, and try to decide whether $\mu = \eta$ or whether μ is *far* from μ.

The problem we described above is a specific instance of *distribution testing*: we are given samples from some unknown distribution μ, and must decide whether μ satisfies some property or whether μ is *far* from all distributions that satisfy the property. Here, "far" is usually taken to mean a large L_1-distance, but other distance measures have also been considered in the literature. Distribution testing has been extensively studied, but to our knowledge, the distributed case has not been considered so far.

We take first steps towards understanding the possibilities and limitations of distributed testing of distributions. Specifically, we study *uniformity testing*, where we wish to determine whether our unknown distribution is the uniform distribution $\mathcal{U}_n$ on $\{1, \ldots, n\}$, or whether it is ϵ-far from $\mathcal{U}_n$ in L_1 distance, for a distance parameter ϵ. Uniformity testing is especially interesting because in the centralized setting, it is known that for any fixed distribution μ, the problem of testing equality to μ can be reduced to uniformity testing [10, 15]. This reduction continues to work in the distributed setting: it is a *filter* (essentially, a randomized mapping), which each node can independently apply to its samples using its own private randomness, and then call the uniformity tester.

We consider a network of k nodes, where each node receives s samples drawn independently from the unknown distribution μ. Our goal is to minimize the number of samples s that each node needs to draw, while also taking into consideration communication and round complexity.

In the centralized setting it is known that $\Theta(\sqrt{n}/\epsilon^2)$ samples are both necessary and sufficient for uniformity testing [21], so clearly we cannot have $s = o(\sqrt{n}/(\epsilon^2 k))$, otherwise the entire network does not have enough samples to decide. This gives us a lower bound on the number of samples at each node. At the other extreme, we know that we can always take $s = \Theta(\sqrt{n}/\epsilon^2)$, so that a single network node has enough samples to decide on its own. We ask whether there is a middle ground: when s is large enough that the total number of samples in the network is $s \cdot k = \Omega(\sqrt{n}/\epsilon^2)$, but too small for nodes to decide on their own (i.e., $s = o(\sqrt{n}/\epsilon^2)$), is there an efficient distributed algorithm that allows the nodes to decide whether the distribution is uniform or far from uniform?

We address this question in three models: the LOCAL model, the CONGEST model, and an intermediate "0-round model". Next we outline our results for each model.

0-round uniformity testing. We begin by studying a very simple model where the nodes cannot communicate with each other at all: each node examines its s samples, and decides whether to accept or reject. As usual in distributed decision, the network as a whole

PODC '18, July 23–27, 2018, Egham, United Kingdom

ACM ISBN 978-1-4503-5795-1/18/07…$15.00
https://doi.org/10.1145/3212734.3212772

is said to *accept* if all nodes accepted; otherwise the network *rejects* ("some node raised an alarm"). We refer to this as the "AND" decision rule. Our goal is to ensure that if the underlying distribution is uniform, the network accepts with high probability, but if the distribution is far from uniform, the network rejects with high probability. Our main motivation for studying the 0-round model is that we can use it to develop uniformity testers for the LOCAL and CONGEST models: we *gather* as many samples as possible at as few nodes as possible, and then have these nodes use the 0-round tester to decide.

To solve uniformity testing in the 0-round model, we develop a single-node tester $\mathcal{A}_\delta$, which gives us a "very weak signal": for some very small parameter δ, when we give $\mathcal{A}_\delta$ samples from the uniform distribution $\mathcal{U}_n$, it accepts w.p. $1-\delta$; but when we give it samples from a distribution that is ϵ-far from $\mathcal{U}_n$, it rejects w.p. $\Theta(\delta)$ (i.e., small but noticeable probability). We show that we can implement $\mathcal{A}_\delta$ using $O(\sqrt{\delta n})$ samples, so when we take $\delta = o(1)$, we save on individual sample complexity.

If we set $\delta = \Theta(1/k)$, we expect that the uniform distribution will be accepted by all nodes, while an ϵ-far distribution will be rejected by at least one node. This is the behavior we are aiming for. Unfortunately, moving from having at least one node reject *in expectation* to having at least one node reject *with high probability* is not so easy. The "AND" decision rule, where we reject if at least one node wants to reject, is not robust where error probability is concerned: in some situations, there exists an efficient algorithm with success probability p, but if we want to increase the success probability to $p' > p$, we may not be able to do so efficiently [12]. The success probability we get by using $\mathcal{A}_\delta$ as outlined above depends on the distance parameter ϵ; it is roughly $1/2 + \Theta(\epsilon^2)$, and amplifying to a constant success probability is expensive.

Ultimately, we are able to show the following: for a constant C_p which depends on the desired success probability p,

Theorem 1.1. *There exists a 0-round ϵ-uniformity tester, with error probability at most p, which uses s samples per node, where*

$$s = \Theta\left((C_p/\epsilon^2)\cdot\sqrt{n/k^{\Theta(\epsilon^2/C_p)}}\right).$$

Motivated by the non-robustness of the standard "AND" decision rule, we also consider a setup where the network uses a *threshold rule* to decide: we fix some threshold $T \in [k]$, and the network is considered to *reject* iff at least T nodes want to reject. For this model we can use standard amplification techniques, and our result is substantially better:

Theorem 1.2. *There exists a 0-round ϵ-uniformity tester in the threshold decision model, with threshold $T = \Theta(1/\epsilon^4)$ that uses $\Theta(\sqrt{n/k}/\epsilon^2)$ samples per player.*

We later use the threshold 0-round tester to develop a (multi-round) tester for the CONGEST model.

For 0-round uniformity testing with the AND decision rule, we show the following lower bound, which matches Theorem 1.2 in term of number of samples, up to the dependence on ϵ:

Theorem 1.3. *For a sufficiently large constant $\epsilon \in (0, 1/2)$, any anonymous ϵ-uniformity tester that achieves error at most $1/3$ in network of size k, has sample complexity $\Omega(\sqrt{n/k}/\log(n))$ per node.*

An *anonymous* tester is one where all the nodes execute the same algorithm. Note, however, that the algorithm can be *randomized*, so the nodes can choose random identifiers from some large namespace, which will with high probability be unique.

0-round protocols with asymmetric costs. We develop an *asymmetric* version of our 0-round uniformity tester, where each node v can draw a different number of samples, s_v, and has a different cost per sample, c_v. Our goal now is to minimize the maximum cost paid by any node. We show, for example, that we can implement the 0-round threshold tester using $s = \Theta(\sqrt{n}/(\epsilon^2 \cdot \|T\|_2)$ samples, where T is the inverse cost vector defined by $T_v = 1/c_v$ for each $v \in V$, and $\|\cdot\|_2$ is the L_2-norm. Note that when all costs are equal to 1, we have $\|T\|_2 = \sqrt{k}$, so we recover the symmetric case (Theorem 1.2). We also analyze the asymmetric case with the AND-decision rule, and generalize Theorem 1.1.

The CONGEST *model.* Our algorithms for the LOCAL and the CONGEST model assume for simplicity that each node starts with a single sample ($s = 1$), and that there are enough samples in the whole network to solve uniformity (for $s = 1$, this means $k \geq \sqrt{n}/\epsilon^2$); the results generalize in a straightforward manner to larger s. We develop a "sample-gathering" protocol for the CONGEST model, which then allows us to use the 0-round uniformity tester from Theorem 1.2. This yields the following:

Theorem 1.4. *Uniformity testing can be solved in* CONGEST *in $O(D + n/(k\epsilon^4))$ rounds in k-node networks of diameter D.*

The LOCAL *model.* For the LOCAL model, we gives a simple strategy based on first finding a maximal independent set, and then gathering samples at the MIS nodes. Here again we suffer from the LOCAL model's inability to amplify success probabilities (without running for D rounds). Still, if each node initially has one sample, then r rounds suffice to get success probability $1-p$, where

$$r = \Theta\left((C_p/\epsilon^2)\cdot\sqrt{n/k^{\Theta(\epsilon^2/C_p)}}\right)^{1/(1-\Theta(\epsilon^2/C_p))}.$$

This hairy expression tends to $\Theta(\sqrt{n}/\epsilon^2)$ as $\epsilon \to 0$, as in this case we cannot avoid collecting this many samples at one node (which will take a long time, in case the graph is, e.g., a line).

For lack of space, many technical details are omitted here, and will appear in the full version of the paper.

1.1 Related Work

Uniformity testing was the starting point for distribution testing. It was first considered implicitly in [13, 17], as part of a tester that estimated the *expansion* of a graph by testing whether the distribution of a short random walk in the graph is uniform, or far from uniform.

In [13, 17] it is shown that the number of samples needed for uniformity testing is $\Theta(\sqrt{n})$. Following this work, the first explicit study of property testing for distributions was in[4], which showed that testing whether two distributions are far from each other requires $\Theta(n^{2/3})$ samples. Later on, uniformity testing was shown to be complete for the problem of testing equality between an unknown input distribution and any fixed distribution [10, 15]. Distribution testing has expanded into a wide field of work, in which a large

variety of properties of distributions were considered. We refer to [8, 11, 14, 16, 22] for background on property testing in general, and on distribution testing in particular.

A reduction between uniformity testing and the simultaneous communication complexity of Equality, using private randomness, was shown in [5]. We use this reduction to give a lower bound on distributed uniformity testing, but under a different error regime than the classical model. As for Equality itself, its simultaneous communication complexity with private randomness was studied in [3, 7, 19], and another optimal simultaneous protocol is given in [2]. In [6], the simultaneous Equality lower bound was revisited, using an information theoretic approach. We use the technique of [6] to prove a lower bound on simultaneous Equality with non-standard error.

Very recently, [1] studied testing properties of distributions (including uniformity testing) in a model similar to our 0-round model. The focus of [1] is mostly orthogonal to ours. In [1], each player receives only one sample, and can send a short message to a *referee*; the referee then decides whether the input distribution satisfies the property. The referee's output can be an arbitrary function of the messages it receives from the players. The focus in [1] is on the trade-off between number of players (which controls the number of samples) and communication per player. In contrast, our focus here is on the number of samples required per player for a given number of players, where each player is allowed to output only one bit (accept/reject). We do not allow the "referee" to apply any decision rule, and instead focus on the traditional "AND" rule, and on a threshold rule. As a result, our results and [1] do not overlap.

2 PRELIMINARIES

Distributed uniformity testing. In the distributed ϵ-uniformity testing problem, each network node $i \in V$ is given a set of s iid samples $\boldsymbol{S}_i \sim \mu^s$ from an unknown distribution $\mu : \Omega \to [0, 1]$. We assume that the domain Ω is known in advance, and let $n := |\Omega|$ be its size. (Our testers do not actually need to know Ω, only n, but our lower bound assumes that Ω is known as well.) The goal is to distinguish the case where μ is the uniform distribution $\mathcal{U}$ on Ω, from the case where μ has L_1 distance at least ϵ from $\mathcal{U}$. (The L_1 distance between a distribution μ and the uniform distribution is $\sum_{\omega\in\Omega} \left|\mu(\omega) - \frac{1}{n}\right|$.) We let $\mathcal{F}_\epsilon(\mathcal{U})$ denote the set of distributions on Ω that are at least ϵ-far from uniform.

Although we generally aim to develop testers with constant error probability, e.g. 1/3, along the way we construct testers with much higher, and often asymmetric, error. In particular, we often need to make sure that the uniform distribution is accepted with very high probability (so that all nodes accept it), but any ϵ-far distribution is only rejected with "not too small" probability (but much less than 1/2). This motivates the following definition, for centralized (i.e., single-node) testers with asymmetric error:

Definition 1 ((δ, α)-gap tester). *Fix $\delta \in (0, 1)$ and $\alpha > 1$, and let $\mathcal{P}$ be a family of distributions. An algorithm $\mathcal{A}$ is said to be a* (δ, α)-gap tester *for $\mathcal{P}$ using s samples if for any distribution μ,*

(1) If $\mu \in \mathcal{P}$, then $\Pr_{\mathcal{S}\sim\mu^s}[\mathcal{A}(\mathcal{S}) = 1] \geq 1 - \delta$, and
(2) If μ is ϵ-far from $\mathcal{P}$, then $\Pr_{\mathcal{S}\sim\mu^s}[\mathcal{A}(\mathcal{S}) = 1] \leq 1 - \alpha \cdot \delta$.

Here, $\mathcal{A}(\boldsymbol{S})$ is the random variable denoting the output of $\mathcal{A}$ on samples $\boldsymbol{S}$. We usually take α to be only slightly greater than 1.

When working with such a delicate error regime, the following lemma will be useful. It shows that in order to achieve the separation required for a (δ, α)-tester, some small but non-negligible amount of information is required:

Lemma 2.1. *Let B_p denote the Bernoulli distribution with probability p. For any $\delta \in (0, 1/4)$ and $\tau \in (1, \frac{1}{\delta})$,*

$$\mathrm{D}_{KL}\left(\mathrm{B}_{1-\delta} \parallel \mathrm{B}_{1-\tau\delta}\right) \geq \frac{\delta}{4}\left(\tau - 1 - \log\tau\right).$$

Here, $\mathrm{D}_{KL}(\mu \parallel \eta) = \sum_x \mu(x)\log(\mu(x)/\eta(x))$ is the KL-divergence between distributions μ and η, a measure of how different they are from each other.

Distributed models. In addition to the standard distributed models, LOCAL and CONGEST, we consider a *0-round model*. In this model, each node outputs its decision without communicating with the other nodes. We consider two decision rules: the *AND rule*, where the network is said to accept iff all nodes accept; and the *threshold rule*, where we fix a threshold T, and the network is said to accept iff fewer than T nodes reject.

Simultaneous communication complexity. Our lower bound for uniformity testing takes a detour through *simultaneous communication complexity*. Here, we have three players: Alice, Bob, and a referee. Alice and Bob are given inputs $X, Y \in \{0, 1\}^n$, and each player sends one message to the referee, who then outputs a value. The goal is for the referee to compute some function $f(X, Y)$ of Alice and Bob's inputs (which it cannot see directly). The *cost* of a simultaneous protocol is the worst-case maximum length of a message sent by Alice or Bob.

We are specifically interested in *private-coin* protocols, where Alice and Bob each have private randomness that is not shared with the other or with the referee. We let $\mathrm{SMP}_{\delta_0,\delta_1}(f)$ denote the minimum cost of a private-coin protocol, where if $f(X, Y) = 1$, the referee outputs 1 w.p. $\geq 1 - \delta_1$; and if $f(X, Y) = 0$, the referee outputs 0 w.p. $\geq 1 - \delta_0$.

3 0-ROUND DISTRIBUTED UNIFORMITY TESTING

We begin our investigation of distributed uniformity testing by studying the 0-round model, where no communication is allowed. We show that we can develop a tester where the number of samples at each node is much smaller than $\Theta(\sqrt{n}/\epsilon^2)$ (i.e., no single node can test uniformity on its own). Our strategy is to divide the responsibility for rejecting: we ask each node to reject an ϵ-far distribution with small (but noticeable) probability, but accept the uniform distribution with very high probability. This ensures that the network *as a whole* can distinguish an ϵ-far distribution from the uniform distribution.

First, we show that a single node using $s = \Theta(\sqrt{\delta n})$ samples is able to achieve probability $1 - \delta$ of accepting the uniform distribution, and probability at least $(1 + \Theta(\epsilon^2))\delta$ of rejecting any distribution that is ϵ-far from uniform.

3.1 The Collision-Based Tester, Revisited

It is well established (see, e.g., [16]) that when testing uniformity or any other symmetric property of distributions, all the relevant information is captured by the histogram of the samples: namely, one looks for *collisions* among the samples. Our challenge here is that we want to use a very small number of samples, such that the expected number of collisions is much smaller than one, and the probability of seeing *two* collisions is negligible. Unlike the optimal centralized uniformity tester [21], it is pointless for us to start *counting* collisions, because we will probably never see more than one.

For this reason, the tester we consider simply retrieves s samples from the oracle, and accepts (i.e., returns "uniform") if and only if all the samples are distinct. The question is: how large does s need to be, to guarantee the gap we need for a (δ, α)-gap uniformity tester? We show:

Theorem 3.1 (Gap tester, informal)**.** *There is a $(\delta, 1+\Theta(\epsilon^2))$-gap tester that uses $s = \Theta(\sqrt{\delta n})$ samples, provided δ is not too large compared to ϵ, and n is not too small.*

We then discuss ways of amplifying the success probability.

The more strict we get with the relations between δ, ϵ, n, the better gap we have. Specifically, for the distributed setting we will use a $(\delta, 1+\epsilon^2/2)$-gap tester, that uses $s = \Theta(\sqrt{\delta n})$ samples, and works provided that $\delta < \epsilon^4/64$ and $n > 64/(\epsilon^4\delta)$.

The analysis goes as follows: fix a distribution μ on a domain of n elements, and let χ be the collision probability of μ:

$$\chi = \chi(\mu) = \Pr_{X,Y\sim\mu}[X = Y] = \sum_{x\in \mathrm{Supp}(\mu)} \mu(x)^2.$$

The uniform distribution has $\chi(\mathcal{U}) = 1/n$. At the heart of collision-based uniformity testers is the following well-known property (see the textbook [16]):

Lemma 3.2. *If μ is ϵ-far from the uniform distribution, then $\chi(\mu) > (1+\epsilon^2)/n$.*

Let s be such that $s(s-1) = 2\delta n$; we assume that s is an integer (and will make sure that this assumption is satisfied when we use the tester).

In our analysis, we try to reach a gap as close as possible to $1+\epsilon^2$, but in fact we only reach $1+\gamma\epsilon^2$, where

$$\gamma := 1 - \frac{1}{s} - \sqrt{2\delta(1+\epsilon^2)} - \frac{\frac{1}{s} + \sqrt{2\delta(1+\epsilon^2)}}{\epsilon^2}. \tag{1}$$

This is a "slack term" that approaches 1 as $n/k \to \infty$.

Let $S \sim \mu^s$ be a set of s iid samples from μ, and let Z be an indicator for the event that there is a collision in S (that is, S contains two identical samples). Our tester outputs 0 iff $Z = 1$. Our analysis for the tester relies on a tight bound from [18] for the probability of getting a collision in a set of s samples:

Lemma 3.3 (Theorem 3, [18])**.** *For any distribution μ, we have $\Pr_{S\sim\mu^s}[Z=0] \le e^{-(s-1)\cdot\sqrt{\chi}} \cdot \left(1+(s-1)\cdot\sqrt{\chi}\right)$.*

Using this bound, we show:

Lemma 3.4. *The single-collision tester satisfies:*

(1) $(1-\delta)$-completeness: if $\mu = \mathcal{U}$ is the uniform distribution, then $\Pr_{S\sim\mu^s}[Z=0] \ge 1-\delta$, and

(2) $(\alpha\cdot\delta)$-soundness: if μ is ϵ-far from the uniform distribution, then $\Pr_{S\sim\mu^s}[Z=0] \le 1-\alpha\cdot\delta$, where $\alpha = 1+\Theta(\epsilon^2)$.

Proof sketch. Suppose $\mu = \mathcal{U}$ is the uniform distribution. What is the probability that we see no collisions? It turns out that Markov's inequality gives a tight bound in this case, since the probability for more than one collision is negligible. We have $\Pr[Z \ge 1] \le \mathrm{E}[Z]/1 = \binom{s}{2}/n$. Recall that we chose s so that $s(s-1) = 2\delta n$, that is, $\binom{s}{2}/n = \delta$; thus, $\Pr[Z \ge 1] \le \delta$, as desired.

Now suppose μ is ϵ-far from uniform. Then Lemma 3.3, together with Lemma 3.2, yields

$$\Pr[Z=0] \le e^{-(s-1)\sqrt{(1+\epsilon^2)/n}} \cdot \left(1+(s-1)\sqrt{\frac{1+\epsilon^2}{n}}\right). \tag{2}$$

To analyze the expression in (2), let us denote $t := (s-1)\sqrt{(1+\epsilon^2)/n}$, so that (2) takes the form: $\Pr[Z=0] \le e^{-t}(t+1)$. From the Taylor expansion $e^{-x} = 1-x+x^2/2-O(x^3)$ we know that $e^{-x} \le 1-x+x^2/2$ for all $x \ge 0$, so

$$\Pr[Z=0] \le \left(1-t+\frac{t^2}{2}\right)(1+t) = 1-\frac{1}{2}\left(t^2-t^3\right). \tag{3}$$

Recall that we want to show that $\Pr[Z=0] \le 1-\alpha\cdot\delta$, where α is as large as we can make it — we are aiming for $\alpha = 1+\Theta(\epsilon^2)$. Thus, our goal is to show that $\left(t^2-t^3\right)/(2\delta) \ge 1+\Theta(\epsilon^2)$. We compute the value of t^2, and obtain $t^2 = \left(1-\frac{1}{s}\right)(1+\epsilon^2)\cdot 2\delta$. As for t^3, we are able to bound it from above by $t^3 \le \sqrt{2\delta(1+\epsilon^2)}\cdot t^2$. Plugging these bounds into (3) shows that $\Pr[Z=0] \le 1-(1+\gamma\epsilon^2)\delta$, as required. □

3.2 From the Collision-Based Tester to a Distributed Algorithm

Let $\mathcal{A}_\delta$ be the $(\delta, 1+\Theta(\epsilon^2))$-gap tester we developed in the previous section. This tester has very high failure probability: on a distribution that is ϵ-far from uniform, it almost certainly accepts, even though it should reject. We would like to use $\mathcal{A}_\delta$ to obtain a distributed 0-round algorithm in which the network as a whole has only *constant* (e.g., 1/3) failure probability.

We consider two different approaches: first, we adapt the tester to work with the "standard" distributed decision model, where on YES-instances we require all nodes to accept, and on NO-instances at least one node should reject. Unfortunately, this model is not amplification-friendly: it does not allow us to reduce the error probability by repeating our distributed tester several times and taking, say, the majority. (Here we mean the majority decision of the *entire network*, not the individual nodes.) Hence, the number of samples required degrades significantly when we aim for constant error.

The second model we consider is a *threshold decision* model, where we can set a threshold T, and define that the network has "accepted" an input if at least T nodes accepted, and otherwise we say that the network "rejected". This model *is* amenable to amplification using standard techniques. We will use our results for the threshold-decision model when we develop a multi-round tester for the CONGEST model.

3.2.1 The Standard Distributed Decision Model. Suppose we want the network to decide correctly with probability at least $1-p$ on any distribution. With the standard decision model, it suffices to require that if each node runs a tester $\mathcal{B}$,

- For the uniform distribution, the probability that $\mathcal{B}$ rejects is at most $\ln((1/(1-p))/(1-\beta))/k$, where $\beta \coloneqq \ln(1/(1-p))/(2k)$ is a small slack term approaching 0 as $k \to \infty$.
- For any distribution that is ϵ-far from uniform, the probability that $\mathcal{B}$ rejects is at least $\ln(1/p)/k$.

Essentially, $\mathcal{B}$ needs to be a $(\Theta(1/k), 1+\Omega(1))$-gap tester, where the constant in the $\Omega(1)$ depends on the probability p. This ensures that the uniform distribution is accepted by all nodes with probability $1-p$, while for any distribution that is ϵ-far from uniform, with probability at least $1-p$ some node rejects.

Our tester from the previous section, $\mathcal{A}_\delta$, has a gap of only $1+\Theta(\epsilon^2)$. We need to increase the gap to at least C_p, where C_p is a constant that depends on the desired failure probability p. However, while doing so, we must preserve the very high acceptance probability for the uniform distribution, to make sure that w.h.p. the entire network accepts it.

A natural strategy to amplify the gap is to run the tester $\mathcal{A}_\delta$ independently m times, and reject iff all m trials rejected. To reach our desired gap of C_p, we need to take $m = \log_{1+\Theta(\epsilon^2)}(C_p) = \Theta(C_p/\epsilon^2)$ repetitions. However, the undesired side-effect of this strategy is that the acceptance probability is now decreased: if we started with probability δ of rejecting the uniform distribution, the probability that m independent trials all reject the uniform distribution is now δ^m. We require:

$$\left(1-\delta^m\right)^k \geq 1-p, \tag{4}$$

and therefore $\delta = \Theta(1/k^{1/m})$.

We instantiate this scheme by having every network node run $m = \Theta(C_p/\epsilon^2)$ independent repetitions of the tester $\mathcal{A}_\delta$ with $\delta = \Theta(1/k^{1/m})$. When k, n are sufficiently large compared to $1/\epsilon$, we obtain the tester described in Theorem 1.1. The full proof and the exact calculations are somewhat laborious, and are omitted from this text.

3.2.2 A Threshold-Based Tester. A significant improvement can be made when we change the decision rule to *threshold*. While asking each node to amplify its own gap by itself is very expensive, if we let the *whole network* work together, we can leverage the weak signal each node provides much more efficiently.

Proof of Theorem 1.2. We will again run the single-collision tester $\mathcal{A}_\delta$, but this time we set the parameters so as to create a large gap between the expected number of nodes that reject when the distribution is uniform, and the expected number of rejections when the distribution is ϵ-far from uniform.

Let $\boldsymbol{R}_v$ denote the event that node v rejects when it runs $\mathcal{A}_\delta$, and let $\boldsymbol{R}$ count the number of rejecting nodes in the network. Let $\eta(\mu) \coloneqq \mathrm{E}_{S_1,\ldots,S_k\sim\mu^s}[\boldsymbol{R}]$ denote the expected number of rejections when all nodes run $\mathcal{A}_\delta$ with samples drawn from μ. We simplify the notation by writing the distribution μ in the subscript, instead of the samples $\boldsymbol{S}_1,\ldots,\boldsymbol{S}_k \sim \mu^s$.

Recall that $\mathcal{A}_\delta$ is a $(\delta, 1+\gamma\epsilon^2)$-gap tester: that is, $\Pr_{\mathcal{U}}[\boldsymbol{R}_v = 1] \leq \delta$, and for any $\mu \in \mathcal{F}_\epsilon(\mathcal{U})$ we have $\Pr_\mu[\boldsymbol{R}_v = 1] \geq \left(1+\gamma\epsilon^2\right)\delta$. For our purposes here, it suffices to take $\gamma \geq 1/2$. Therefore, $\eta(\mathcal{U}) \leq k\cdot\delta$, and for all $\mu \in \mathcal{F}_\epsilon(\mathcal{U})$, $\eta(\mu) \geq k\cdot\left(1+\epsilon^2/2\right)\delta$. By Chernoff,

$$\Pr_{\mathcal{U}}[\boldsymbol{R} \geq T] = \Pr_{\mathcal{U}}\left[\boldsymbol{R} \geq \eta(\mathcal{U})\cdot\left(1+\frac{T-\eta(\mathcal{U})}{\eta(\mathcal{U})}\right)\right] \leq e^{-((T-\eta(\mathcal{U}))^2/(3\eta(\mathcal{U}))},$$

and for any $\mu \in \mathcal{F}_\epsilon(\mathcal{U})$,

$$\Pr_{\mu}[\boldsymbol{R} < T] = \Pr_{\mu}\left[\boldsymbol{R} < \eta(\mu)\cdot\left(1-\frac{\eta(\mu)-T}{\eta(\mu)}\right)\right] \leq e^{-(T-\eta(\mu))^2/(2\eta(\mu))}.$$

Of course, we would like to put the threshold T in between the two expectations $\eta(\mathcal{U})$ and $\eta(\mu)$ for any $\mu \in \mathcal{F}_\epsilon(\mathcal{U})$, such that the probabilities above are both bounded by $1/3$. It suffices to require (for all $\mu \in \mathcal{F}_\epsilon(\mathcal{U})$):

$$\eta(\mathcal{U}) + \sqrt{3\ln(3)\eta(\mathcal{U})} \leq T \leq \eta(\mu) - \sqrt{2\ln(3)\eta(\mu)}. \tag{5}$$

In order for a threshold T satisfying (5) to exist, we can take $\delta = \Theta(1/(\epsilon^4 k))$, which allows us to set $T = \Theta(1/\epsilon^4)$ (full details are omitted). Plugging this value of δ into Theorem 3.1, we see that $s = \Theta(\sqrt{n/k}/\epsilon^2)$ samples are sufficient. □

4 THE ASYMMETRIC CASE

We now generalize to the asymmetric case, where each node i can ask to draw s_i samples, at a cost of c_i per sample. The total cost of node i is therefore $s_i \cdot c_i$. We would like to minimize the *maximum individual cost*, $C \coloneqq \max_i\{s_i\cdot c_i\}$.

For convenience, we denote $T_i \coloneqq 1/c_i$. Our results will depend on various norms of the vector $T = (T_1,\ldots,T_k)$ of inverse costs.

The essential idea is to divide the *responsibility* we assign to each node in accordance with the cost it pays, in a way that allows "expensive nodes" (with high c_i) to draw fewer samples than "cheap nodes" (with low c_i). Here, the *responsibility* of node i corresponds to the part it plays in the network's decision.

Since we are interested in optimizing the maximum individual cost, $C = \max_i\{s_i\cdot c_i\}$, we may as well allow *all* nodes to use the same cost C (once we compute what C needs to be). Thus, for each node i, we will choose s_i so that $s_i\cdot c_i = C$, that is, $s_i = C\cdot T_i$.

We will use the collision-based tester fromSection 3, but now each node i will instantiate it with a different value δ_i. However, all nodes will still use the same value of α as in the symmetric case.

4.1 The AND Decision Rule

Let us compute the number of samples s_i that each node i must draw, when the final decision of the network is the AND of the nodes' individual decisions.

Each individual node will run a (δ_i, α_i)-gap tester. We will use the same gap, $\alpha_i = \alpha$, for all nodes, but different thresholds δ_i.

As we saw in the symmetric case, we can obtain a (δ_i, α)-gap tester by taking m repetitions of the tester $\mathcal{A}_{\delta_i'}$ from Section 3, with $\delta_i' = \delta_i^{1/m}$ and $m = \Theta(C_p/\epsilon^2)$. The number of samples required by node i is $s_i = m\cdot\sqrt{2\delta_i' n}$. Recalling that $s_i = C\cdot T_i$, we can express this as:

$$\delta_i = \frac{(C\cdot T_i)^{2m}}{(2n)^m m^{2m}}.$$

How are the various costs δ_i constrained? To ensure that the network accepts the uniform distribution with sufficiently high probability, we set the δ_is so that

$$\prod_{i=1}^{k}(1-\delta_i) = 1-p. \tag{6}$$

We show that if we set the δ_is so that (6) is satisfied, then *soundness* also holds: any distribution that is ϵ-far from uniform is rejected by at least one node, with probability at least $1-p$. Intuitively, this is because: the probability of accepting an ϵ-far distribution is given by $\prod_{i=1}^{k}(1-\alpha\delta_i)$. Under the constraint (6), this expression is maximized when when all δ_is take the same value, i.e., when we are back in the symmetric case. We know that in the symmetric case all players created a tester of the same δ and α, and soundness was satisfied; we show that taking asymmetric values for the δ_is but with all players using the same old α, can only *tighten* the bound, that is, *decrease* the probability that an ϵ-far distribution is accepted.

We prove this intuition using the following technical lemma:

Lemma 4.1. *Fix $c \in (0,1)$, $a \in (1, 1/(1-c))$ and $X = (x_1, \ldots, x_k) \in [0, 1-c]^k$ such that*

$$f_k(X) := \prod_{i=1}^{k}(1-x_i) = c.$$

Denote $Y_k := d_k \cdot (1, \ldots, 1) \in \mathbb{R}^k$, where $d_k := (1 - \sqrt[k]{c})$, so that $f_k(Y_k) = c$. Then the function $g_k(X) := \prod_{i=1}^{k}(1-ax_i)$ satisfies:

$$g_k(X) \le g_k(Y_k).$$

Here, we are looking for a set $\{\delta_i\}_{i=1}^{k}$ satisfying (6) above. For simplicity, let us fix $p = 1/3$.

In the symmetric model, we ended up with each node being a (δ, α)-gap tester (recall the tester $\mathcal{B}$), where $\delta = \Theta(1/k)$ was the value that satisfied the completeness for the whole network. Meaning, $\prod_{i=1}^{k}(1-\delta) \approxeq 1 - 1/3$. We also had $\alpha = \ln(1/p)/\ln(1/(1-p))$ which we denoted as C_p, that was enough to satisfy the soundness of the whole network. For $p = 1/3$, this turns out to be $\alpha \approxeq 2.7$.

We can then use the lemma with $c = \prod_{i=1}^{k}(1-\delta) \approxeq 2/3$, $a = \alpha \approxeq 2.7 < 3 = 1/(1-c)$. The value d_k is (by definition) exactly the same value as δ, since $(1-d_k)^k = c = (1-\delta)^k$. We now apply the lemma with $X = (\delta_1, \ldots, \delta_k)$. Given that the probability of accepting the uniform distribution is good enough (i.e $\prod_{i=1}^{k}(1-\delta_i) = c$), we want to conclude the probability to accept ϵ-far distributions is not too large. Following the lemma, this is given by $g_k(X) = \prod_{i=1}^{k}(1-\alpha\delta_i)$.

Applying the lemma, we get

$$g_k(X) \le g_k(Y_k) = \prod_{i=1}^{k}(1-\alpha d_k) = \prod_{i=1}^{k}(1-\alpha\delta)$$

This last term is exactly the soundness the whole network had when each player applied $\mathcal{B}$ in the symmetric case, which we know to be low enough, by our choosing of $\alpha = C_p$ for $p = 1/3$.

Thus we have our soundness for the asymmetric case "for free" when all players use the same α used in the symmetric case.

We now see that:

$$e^{ln(1-p)} = 1 - p = \prod_{i=1}^{k}(1-\delta_i) \le e^{-\sum \delta_i} = e^{-\frac{C^{2m}}{(2n)^m m^{2m}} \cdot (\sum T_i^{2m})}$$

And therefore our algorithm has the cost of:

$$C = \sqrt{2\sqrt[m]{ln(\frac{1}{1-p})}m\sqrt{n}} \cdot \frac{1}{||T||_{2m}}$$

where $m = \Theta(1/\epsilon^2)$. notice that for the symmetric case with k players, indeed we get $||T||_{2m} = \sqrt{k^{1/(2m)}} = \sqrt{k^{\Theta(\epsilon^2)}}$, as before.

We now turn to prove the lemma:

Proof of Lemma 4.1. We prove the lemma by induction on the dimension k.

For $k = 1$, the claim is trivial: we have a single-dimensional vector, that is, a scalar x, which by the conditions of the lemma satisfies $x = 1 - c$. We also have $Y_1 = d_1 = 1 - c$, and therefore $x = Y_1$ and $g_1(x) = g_1(Y_1)$.

Next, suppose the claim holds for all $j < k$, and let us prove it for k. First, note that $g_k(Y_k) \le g_{k+1}(Y_{k+1})$. Also, we can assume that $x_i \neq 0$ for all $i \in [k]$, otherwise we can reduce the dimension to $k-1$ by eliminating the zero coordinates, and then apply the induction hypothesis.

Consider the compact cube $S = [0, 1-c]^k \subset \mathbb{R}^k$.

First, we look for solutions on the border of our constrained area, that is, the intersection of the manifold $f_k(X) = c$ and the border of S. We note that each point on the border must have $x_i \in \{0, 1-c\}$ for some i, but this implies that either $x_i = 0$ or otherwise $x_j = 0$ for all $j \neq i$. Thus, this case is handled again by the induction hypothesis, in a lower dimension.

Within the inner area of S, we apply the method of Lagrange multipliers over k variables and with $m = 1$ constraints, where our single constraint is $f_k(X) = c$. For brevity, from here and until the end of the proof, we omit the subscript and simply use f, g, Y, d instead.

We define $h(X, \lambda) := g(X) + \lambda \cdot (f(X) - c)$, and write the partial derivatives:

$$\frac{\partial f}{\partial x_i}(X) = -\Pi_{j\neq i}(1-x_j) = \frac{2/3}{x_i - 1},$$

$$\frac{\partial g}{\partial x_i}(X) = -a\Pi_{j\neq i}(1-ax_j) = \frac{-a \cdot g(X)}{(1-ax_i)}.$$

If X is a suspicious point inside of S, it must satisfy:

$$\forall i.\ 0 = \frac{\partial h}{\partial x_i}(X, \lambda) = \frac{\partial g}{\partial x_i}(X) + \lambda\frac{\partial f}{\partial x_i}(X) = \frac{-a \cdot g(X)}{(1-ax_i)} - + \frac{2/3\lambda}{x_i - 1},$$

which implies that for all i,

$$x_i = \frac{ag(X) + 2/3\lambda}{ag(X) + 2/3\lambda a}.$$

Since this holds for any i, and the r.h.s does not depend on i, we get that our point must satisfy $x_i = x_j$ for all $i \neq j$. In other words, if X is a suspicious point, then $X = m \cdot (1, \ldots, 1)$ for some m. Plugging in the last derivative, $\frac{\partial h}{\partial \lambda}(X, \lambda) = f(x) - C = 0$ (that is X is on the manifold $f(X) = c$), we get exactly that $X = Y$, and also $\lambda = \frac{-ag(Y)(1-d)}{2/3(1-ad)}$.

We have now shown that Y is the only suspicious point inside of S. To prove the lemma, we would like to show that Y is indeed an *extremal maximum* inside S. By the bordered Hessian method, a sufficient condition for this is that if we look at the sequence of growing submatrices of the bordered Hessian of $h(X,\lambda)$, the sequence of determinants alternates signs.

We first calculate

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(X) = \Pi_{k\notin\{i,j\}}(1-x_k) = \frac{2/3}{(1-x_i)(1-x_j)},$$

$$\frac{\partial^2 g}{\partial x_i \partial x_j}(X) = a^2\Pi_{k\notin\{i,j\}}(1-ax_k) = \frac{a^2 g(X)}{(1-ax_i)(1-ax_j)}.$$

Therefore, we have:

(1) $w \coloneqq \frac{\partial^2 h}{\partial\lambda\partial x_i}(Y,\lambda) = \frac{\partial(f-c)}{\partial x_i}(Y,\lambda) = \frac{2/3}{d-1}$,

(2) $z \coloneqq \frac{\partial^2 h}{\partial x_j \partial x_i}(Y,\lambda) = \frac{\partial^2 g}{\partial x_j \partial x_i}(Y) + \lambda\frac{\partial^2 f}{\partial x_j \partial x_i}(Y) = \frac{a^2 g(X)}{(1-ax_i)(1-ax_j)} + \lambda\frac{2/3}{(x_i-1)(x_j-1)}$,

(3) $\frac{\partial^2 h}{\partial x_i \partial x_i}(Y,\lambda) = \frac{\partial^2 g}{\partial x_i \partial x_i}(Y) + \lambda\frac{\partial^2 f}{\partial x_i \partial x_i}(Y) = 0.$

The bordered Hessian matrix is given by:

$$H(X,\lambda) = \begin{pmatrix} \frac{\partial^2 h}{\partial\lambda^2}(X,\lambda) & \frac{\partial^2 h}{\partial\lambda\partial X}(X,\lambda) \\ (\frac{\partial^2 h}{\partial\lambda\partial X}(X,\lambda))^\top & \frac{\partial^2 h}{\partial X^2}(X,\lambda) \end{pmatrix}$$

So at point (Y,λ), denoting $L \coloneqq H(Y,\lambda)$ and using the equations above, we have

$$L(i,j) = \begin{cases} 0 & i = j, \\ w & i \neq j \wedge (i = 1 \vee j = 1), \\ z & \text{otherwise.} \end{cases}$$

Now, since we have only one constraint ($m = 1$), we need to look at the upper left submatrices of each size starting from $2m+1 = 3$, and up to $k+m = k+1$, and make sure the signs of their determinants are alternating. We note that each such submatrix preserves the structure of L itself. We now calculate the determinant for general matrix of that form and of size $t \times t$, which we will denote in L_t.

We know that the determinant can be viewed as a sum over all permutation of the parity of the said permutation, times the product of values in entries $(i,\sigma(i))$. Formally we can write:

$$\mathrm{Det}(L_t) \coloneqq \sum_{\sigma\in S_t}(-1)^{\mathrm{sgn}(\sigma)}\Pi_{i=1}^{t}(L_t(i,\sigma(i))).$$

We notice a few facts that help us understand this expression:

(i) For any $\sigma \in S_t$ that has a fixpoint, we get a summand of 0, which we can disregard. Therefore we only sum over the derangements D_t (i.e., permutations with no fixpoint).

(ii) For any $\sigma \in D_t$, we must have $\sigma(1) \neq 1$, as well as $\sigma(j) = 1$ for some $j \neq 1$. So all pairs $(i,\sigma(i))$ are $(1,i_0)$, $(i_1,1)$ for $i_1, i_0 \neq 1$, and all the rest are (i,j) with $i \neq 1, j \neq 1$. Together with the form we have for L_t, we get that $\prod_{i=1}^{t}(L_t(i,\sigma(i))) = w^2 z^{t-2}$, for any $\sigma \in D_t$.

(iii) Letting $E(D_t)$ and $O(D_t)$ denote the number of even and odd derangements of size t, respectively, it is known [20] that $E(D_t) - O(D_t) = (-1)^{t-1}(t-1)$.

Using these facts, we have:

$$\begin{aligned}\mathrm{Det}(L_t) &= \sum_{\sigma\in D_t}(-1)^{\mathrm{sgn}(\sigma)}\prod_{i=1}^{t}(L_t(i,\sigma(i))) \\ &= w^2 z^{t-2}\sum_{\sigma\in D_t}(-1)^{\mathrm{sgn}(\sigma)} = w^2 z^{t-2}(-1)^{t-1}(t-1).\end{aligned}$$

Since $w, z, t-1 > 0$, our sign only depends on the exponent of (-1), so we have $\mathrm{sgn}(\mathrm{Det}(L_t)) = (-1)^{t-1}$.

Starting at L_3, we get a positive determinant, and the signs of $\mathrm{Det}(L_4), \ldots, \mathrm{Det}(L_{k+m})$ alternate . According to the bordered Hessian condition, we get that Y is indeed an extremal maximum inside of S, proving the claim. □

4.2 The Threshold Model

In this model our solution let each player run $\mathcal{A}_{\delta_i}$. Following the same analysis and Chernoff bounds we had before, we need only to replace $k\cdot\delta$ by $\sum_i \delta_i$, and following the same calculations, we will now need:

$$\sum_i \delta_i = \Theta(1/\epsilon^4)$$

In that model, each δ_i required that we take $s_i = \sqrt{2\delta_i n}$ samples, and again setting $C = s_i/T_i$ for all players, we now get that $\delta_i = \frac{C^2 T_i^2}{2n}$, and we can write:

$$\Theta(1/\epsilon^4) = \sum_i \delta_i = \frac{S^2}{2n}\cdot\sum_i T_i^2$$

And therefore:

$$S = \Theta(\frac{\sqrt{n}}{\epsilon^2})\cdot\frac{1}{||T||_2}$$

5 IMPLEMENTATION IN CONGEST

We now show how to solve uniformity testing in the CONGEST model. We assume for simplicity that each node has a single sample; generalizing to more samples is straightforward.

Our goal is to "concentrate" samples at a small number of nodes, who will then use our 0-round uniformity tester with the samples they collected. For this purpose, we define the τ-token-packaging problem.

Definition 2 (Token packaging). *Let $\tau \in \mathbb{N}$. In the τ-token packaging problem, each node $v \in V$ is initially given a single token $t_v \in [n]$. The goal is for the nodes to collectively output a set of multisets ("packages") of tokens, $S_1, \ldots, S_\ell \subseteq \{t_v \mid v \in V\}$, with each node outputting zero or more packages. (The number ℓ of packages is not fixed in advance.)*

We require the following:

(1) Each package is of size exactly τ.

(2) Each token t_v is included in at most one package (that is, we require $|\{i \mid t_v \in S_i\}| \leq 1$).

(3) All but at most $\tau - 1$ tokens are included in some package (that is, $|\{v \mid t_v \notin \bigcup_{i\in[\ell]} S_i\}| \leq \tau - 1$).

Note that even though we did not impose an explicit requirement on the number of packages, the problem definition implies that at most $\lfloor k/\tau \rfloor$ packages are produced, because we start out with n tokens and each token can belong to at most one package.

We also mention that even though the round complexity has the diameter D in it, the players does not need to know the value of D in advance in order to run the algorithm.

Theorem 5.1. *For any $\tau \geq 1$, the τ-token packaging problem can be solved in $O(D + \tau)$ rounds in* CONGEST.

We show how to solve τ-token packaging below, but first let us explain how we use it:

Proof of Theorem 1.4. We start out with k samples, one at each node. Fix a parameter τ for the package size. We package our k tokens into $\ell = \Theta(k/\tau)$ packages, and use the threshold-based tester from Theorem 1.2 with ℓ nodes and τ samples at each node, treating each package as a "virtual node". (If node v outputs $p(v)$ packages, then we have node v simulate $p(v)$ "virtual nodes".)

To use the threshold-based tester, the number of samples at each "virtual node" must satisfy $\tau = \Omega(\sqrt{n/\ell}/\epsilon^2)$. Recalling that $\ell = \Theta(k/\tau)$, we see that we should set $\tau = \Theta(n/(k\epsilon^4))$. Thus, the running time is $O(D + n/(k\epsilon^4))$ rounds: $O(D + \tau)$ rounds to solve τ-token packaging, and an additional D rounds to compute the threshold, by summing up the tree the number of virtual nodes that want to reject. Finally, the root rejects iff the number of rejecting nodes crosses the threshold, and the other nodes always accept. □

Solving token-packaging. The algorithm for token-packaging works as follows: first, the network identifies the vertex $r \in V$ with the largest identifier, and then constructs a BFS tree T rooted at node r. We will forward tokens up the tree: each tree node v that receives a total of $m(v)$ tokens from its subtree will keep $p(v) := \lfloor m(v)/\tau \rfloor$ full packages to itself (i.e., $p(v) \cdot \tau$ tokens), and send the rest of the tokens, $c(v) := m(v) \bmod \tau$, up the tree.

We begin by having each node v compute the number $c(v)$ of tokens it needs to sent upwards. The computation proceeds up the tree, as follows:

- If v is a leaf, then $c(v) = 1$.
- If v is not a leaf, and the children of v are $u_1, \ldots, u_d$, then

$$c(v) = \left(1 + \sum_{i=1}^{d} c(u_i)\right) \bmod \tau.$$

Next, for τ rounds, the nodes propagate tokens up the tree, with each node v passing the first $c(v)$ tokens it receives up to its parent, and keeping the rest for itself. Note that $c(v) < \tau$ for all $v \in V$. After τ rounds, the root r discards $c(r)$ tokens it received. Each node now has an integer multiple of τ tokens, which it packages and outputs.

Correctness of the algorithm. Recall that each tree node first forwards all $c(v)$ tokens it will *ever* forward up the tree, and then keeps the remaining tokens for itself. Intuitively, we show that tokens "pipeline" up the tree, so that every node has some new token to send in each round from the beginning of the algorithm until it has sent $c(v)$ tokens. Since $c(v) < \tau$, this means that after τ rounds, all nodes are done sending up the tree, and all tokens have reached their final destinations.

Let $h(v)$ denote the height of node v in the BFS tree (that is, the depth of its subtree, with leaves having depth one), let $children(v)$ be the children of v.

We show that σ rounds are enough for each node v to receive and send up the $c(v) < \sigma$ tokens it needs to forward up the tree, and thereby establish the correctness of our algorithm. Formally, we argue by induction on the height $h(v)$ that:

Invariant 1. At any time $t \leq c(v)$,

(1) If $\ell_v(t) := |sent_v(t)|$ is the number of tokens v sent up to its parent by time t, then we have $\ell_v(t) = t$; and
(2) If $t < c(v)$, then at time t node v has $tokens_v(t) \neq \emptyset$.

(Recall that for the root, we interpret $\ell_v(t)$ as the number of tokens it discarded.)

This shows that at time $t = \sigma$, each node v has already sent on (or discarded, in the case of the root) $c(v)$ tokens, because $c(v) < \sigma$.

An easy induction up the tree shows that after the algorithm concludes, each node has an integer multiple of τ tokens, which it packages and outputs. Tokens are only ever discarded at the root, and the root r discards $c(r) < \tau$ tokens; therefore, all but at most $\tau - 1$ tokens are packaged. Thus, our algorithm solves τ-token packaging.

6 IMPLEMENTATION IN LOCAL

We assume again that each node initially has one sample. (This is not essential.)

As in the CONGEST tester, we want to concentrate samples at a small number of nodes, and then use the 0-round uniformity tester. Because we do not care about congestion, in the LOCAL model with $O(r)$ rounds, any node can send its sample to any node at distance $O(r)$.

Let $N^t(v)$ denote the t-neighborhood of $v \in V$. Our strategy is to first use Luby's MIS algorithm to find a maximal independent set on the graph G^r (where for any $u, v \in V$, we have $\{u, v\} \in E(G^r)$ iff the distance between u, v in G is at most r). Let S be the MIS we compute. Each non-MIS node $u \in V \setminus S$ must have some MIS node in its r-neighborhood; node u selects some MIS node $v \in S \cap N^r(u)$, and routes its sample to v, by asking the nodes in its r-neighborhood to forward the sample to v.

How many samples does each MIS node collect? For each $v \in S$, we know that there is no other MIS node in the $r/2$-neighborhood of v, otherwise S would not be an independent set of G^r. Therefore, all samples in $N^{r/2}(v)$ are routed to v. There are at least $r/2$ such samples: because the graph is connected, we have $|N^{r/2}(v)| \geq r/2$. This also implies that $|S| \leq \lfloor 2k/r \rfloor$.

Now we call the 0-round tester from Theorem 1.1, with $\lfloor 2k/r \rfloor$ "virtual nodes" (the MIS nodes), each having $r/2$ samples. If we take $r = \Theta\left((C_p/\epsilon^2) \cdot \sqrt{n/k^{\Theta(\epsilon^2/C_p)}}\right)^{1/(1-\Theta(\epsilon^2/C_p))}$, then by Theorem 1.1, we achieve success probability $1 - p$.

7 LOWER BOUND FOR ANONYMOUS 0-ROUND UNIFORMITY TESTING

In this section we show that any *anonymous* 0-round distributed uniformity tester requires at least $\Theta(\sqrt{n/k})$ samples at some network node in order to succeed with probability 2/3. An anonymous tester is one where all network nodes execute the same randomized algorithm. Note, however, that the nodes can make random choices, and in particular, they can choose a random identifier from any domain they desire. Technically, the anonymity requirement can be

weakened further: what we need is that for any distribution μ that is ϵ-far from uniform, all network nodes have *the same probability* of rejecting μ.

Remark. The lower bound technique we use here cannot recover the dependence on the distance parameter ϵ, which we believe is $\Omega(1/\epsilon^2)$, in keeping with the centralized case. Throughout this section we mostly treat ϵ as a constant.

For the lower bound, we start by showing that in order for a single node to solve uniformity testing with $(1-\tau\delta, \delta)$-error, it must draw $\Omega(\sqrt{\delta n})$ samples. This lower bound is shown using the reduction from the 2-party Equality function given in [5]:

Theorem 7.1 ([5]). *Suppose we have a q-sample ϵ-uniformity tester, for a sufficiently small constant ϵ, with error (δ_0, δ_1). Then we must have $\mathsf{SMP}_{\delta_0,\delta_1}(EQ) \le q \cdot \log n$.*

Since we are interested in testers with asymmetric error, we study EQ in the asymmetric-error regime.

7.1 Lower Bound for Equality with Asymmetric Error

In this section we outline a lower bound on the simultaneous communication complexity of the Equality function: $\mathrm{EQ}(X, Y) = 1$ iff $X = Y$. Specifically, we are interested in the case where if $X = Y$, the protocol is *almost always right*: it outputs 1 w.p. at least $1-\delta$. But if $X \neq Y$, we allow the protocol to *almost always be wrong*, and we only require that it outputs 0 w.p. $\tau \cdot \delta$, for some constant $\tau > 1$. Note that this means the *gap* between the acceptance probability for YES-instances and for NO-instances is tiny, only $\Theta(\delta)$. Still, we can show the following lower bound: let $f(\tau) := \tau - 1 - \log(\tau)$.

Theorem 7.2. *Let $\tau > 1$ and $\delta < \min\{1/\tau, 1/4\}$. For any $\tau' > \tau$ we have $\mathsf{SMP}_{(1-\tau'\delta),\delta}(EQ) = \Omega(\sqrt{f(\tau)\delta n})$.*

We adapt the information-theoretic lower bound for EQ with constant error from [6] (fixing a mistake in [6] along the way). We give only a high-level overview of the proof here.

Proof sketch for Theorem 7.2. Fix a simultaneous private-coin protocol Π for equality, and let $\boldsymbol{M}_A, \boldsymbol{M}_B$ be the messages sent by Alice and Bob, respectively. (These are random variables, because the protocol is randomized.) Let $\pi(\eta)$ be the joint distribution of the players' messages, when their inputs are drawn from the distribution η. For a distribution μ on $\{0,1\}^n$, let $\mu_=$ denote the distribution on inputs $(\boldsymbol{X}, \boldsymbol{Y})$ where $\boldsymbol{X} \sim \mu$ and $\boldsymbol{X} = \boldsymbol{Y}$. And finally, let $\mu_\times$ be the distribution where $\boldsymbol{X}, \boldsymbol{Y}$ are drawn *independently* from μ.

We would like to find a "hard distribution" μ on $\{0,1\}^n$, such that: (1) when we draw the players' inputs independently, we have $\Pr_{\mu_\times}[\boldsymbol{X} = \boldsymbol{Y}] = o(1)$; but, (2) the distribution $\pi(\mu_=)$ of messages the referee sees when $(\boldsymbol{X}, \boldsymbol{Y}) \sim \mu_=$ is very similar to the distribution $\pi(\mu_\times)$ of messages it gets when $(\boldsymbol{X}, \boldsymbol{Y}) \sim \mu_\times$. Together, (1) and (2) mean the protocol has high error: the referee cannot distinguish the distribution $\mu_=$, where the inputs are *always* equal, from the distribution $\mu_\times$, where they are almost *never* equal.

For a starting point, we examine the protocol's behavior on the uniform distribution $\mathcal{U}$ on $\{0,1\}^n$. This distribution certainly satisfies (1), as $\Pr_{\mathcal{U}_\times}[\boldsymbol{X} = \boldsymbol{Y}] = 2^{-n}$. However, it does not necessarily satisfy (2). For example, if Alice and Bob each send the referee the parity of their input, then $\pi(\mu_=)$ looks very different from $\pi(\mu_\times)$. Thus, we modify the distribution, by *fixing* those parts of the input that "the players talk about" to some constant value. In our example, we would fix the parity of the input to, say, 0, and set our new distribution to be uniform *subject* to a parity of 0. After this fixing, the referee can no longer distinguish $\pi(\mu_=)$ from $\pi(\mu_\times)$, as in both cases it gets the bit '0' from each player.

Technically, we proceed as follows. We consider an "average" set of messages $B_1, \ldots, B_m$ from Bob, and show that if we take $m = \Theta(|\boldsymbol{M}_A|/(f(\tau)\delta))$ messages and *condition* the input on these messages, then with good probability we "neutralize" all but at most $f(\tau)\cdot\delta$ of the difference, measured in KL-divergence [9] between $\pi(\mu_=)$ and $\pi(\mu_\times)$. Intuitively, after this many messages, we have probably already seen "everything Bob wants to say" that might significantly tie his input with Alice's message $\boldsymbol{M}_A$.

We also show that since the referee accepts w.p. $\ge 1-\delta$ on getting messages from $\pi(\mu_=)$, but accepts w.p. $\le 1-\tau\delta$ on seeing $\pi(\mu_\times)$, these two distributions need to be "different": they must have KL-divergence at least $f(\tau)\cdot\delta$ (see Lemma 2.1). Thus, we can fix a set of m messages from Bob and condition the input on them, such that the referee can no longer distinguish $\pi(\mu_=)$ from $\pi(\mu_\times)$ with a gap of $\Theta(\tau\delta)$.

So far we glossed over an important point: if we are too aggressive in fixing parts of the input, then we might violate our first requirement, that $\Pr_{\mu_\times}[\boldsymbol{X} = \boldsymbol{Y}]$ be very small. (For example, if we fix the *entire* input to 0^n, then of course we satisfy requirement (2), but at the cost of always having $\boldsymbol{X} = \boldsymbol{Y}$.) We must show that there is a set $B_1, \ldots, B_m$ of messages from Bob that both "neutralizes" the dependence with Alice *and* has low probability that $\boldsymbol{X} = \boldsymbol{Y}$. To do this, we show that for "typical" messages $B_1, \ldots, B_m$, when we condition on these messages, the resulting distribution only loses $O(m \cdot |\boldsymbol{M}_B|)$ bits of *collision-entropy*. The collision-entropy of a distribution η is defined as $\mathrm{H}_2(\eta) = -\log \Pr_{(\boldsymbol{A},\boldsymbol{B})\sim\eta_\times}[\boldsymbol{A} = \boldsymbol{B}]$, so high collision-entropy implies low probability of collision.[1] Recall that we started from the uniform distribution, which has $\mathrm{H}_2(\mathcal{U}) = n$, so if we make sure that $m \cdot |\boldsymbol{M}_B| \ll n$, we guarantee that the requirement (2) holds.

Combining our constraints we see that we require $m \ge |\boldsymbol{M}_A| \cdot f(\tau)\delta$ to make sure that the dependence between the players' inputs is sufficiently reduced, but we also need $m \cdot |\boldsymbol{M}_B| \ll n$ to guarantee small collision probability. If $|\boldsymbol{M}_A| \cdot |\boldsymbol{M}_B| = o(f(\tau)\delta n)$, then indeed there exists such an m, and we can construct our hard distribution which fools the protocol. Therefore, by contradiction, we must have $\max\{|\boldsymbol{M}_A|, |\boldsymbol{M}_B|\} = \Omega(\sqrt{f(\tau)\delta n})$. □

At this point we have proven the lower bound for simultaneous Equality with a gap of $\Theta(\delta)$. Since we see this problem as having independent interest, we show that the bound is tight:

Lemma 7.3. *Fix a constant $\tau > 1$. Then $\mathsf{SMP}_{(1-\tau\delta),\delta}(EQ) = O(\sqrt{\delta n})$.*

It is known that $\mathsf{SMP}_{1/3,1/3}(\mathrm{EQ}) = O(\sqrt{n})$ [2], so this claim would be trivial if we were interested in *expected* communication; but here we want small *worst-case* communication.

[1] The mistake in [6] is that they used Shannon entropy to make this argument, but high Shannon entropy does not imply low collision probability. We changed this part of the proof.

Proof of Lemma 7.3. We modify the protocol from [2].

Let m be an integer such that $3n \le m \le 4n$ and also $m = (6m_0)^2$ for some $m_0 \in \mathbb{N}$. (For sufficiently large n, there exists such m.)

The protocol uses the *Justesen error-correcting code*: a mapping $C : \{0,1\}^{m/3} \to \{0,1\}^m$, such that whenever $X \neq Y$, we have $\Delta(C(X), C(Y)) \ge m/6$ (that is, $C(X)$ and $C(Y)$ differ in at least $m/6$ places). Since $m = (6m_0)^2$, we can view each codeword as a table of size $(6m_0) \times (6m_0)$. We denote by $C(X)_{i,j}$ entry (i,j) of the table (that is, location $6m_0 \cdot i + j$ of $C(X)$). For convenience, we view the table as a *torus*, so that once we reach the last row or column we wrap-around back to the first row or column.

Now, let $t := \left\lceil \sqrt{24\tau\delta n} \right\rceil$. Alice and Bob receive inputs X, Y, and both compute $C(X), C(Y) \in \{0,1\}^m$. Alice picks a random entry $(a_1, a_2) \in [6m_0]^2$ of the matrix, and sends (a_1, a_2), together with a vertical "chunk" of t bits of her encoded input starting at (a_1, a_2): she sends

$$C(X)_{a_1,a_2}, C(X)_{a_1+1 \bmod (6m_0),a_2}, \ldots, C(X)_{a_1+t-1 \bmod (6m_0),a_2}.$$

Similarly, Bob picks a random entry $(b_1, b_2) \in [6m_0]^2$ of the matrix, and sends (b_1, b_2), together with a horizontal "chunk" of t bits: bits $C(X)_{b_1,b_2}, C(X)_{b_1+1 \bmod (6m_0),b_2}, \ldots, C(X)_{b_1+t-1 \bmod (6m_0),b_2}$.

The referee checks whether there is a location (i,j) sent by both players. If so, he compares this location, and accepts iff $C(X)_{i,j} = C(Y)_{i,j}$; otherwise he simply accepts. this protocol accepts "yes" instances with probability $1 - \delta$, and rejects "no" instances with probability $\tau\delta$. □

7.2 Small-Gap Uniformity Testing

We now return to our original question: uniformity testing with asymmetric error. Using the reduction from [5] (Theorem 7.1), our Equality lower bound implies:

Corollary 7.4. *For a sufficiently large constant $\epsilon \in (0, 1/2)$, and for any $\delta < 1, \alpha > 1$, the query complexity of a (δ, α)-gap ϵ-uniformity tester is $\Omega\left(\sqrt{f(\alpha)\delta n}/\log(n)\right)$.*

Using a constant α, we can now prove our lower bound for distributed 0-round uniformity testing:

Proof of Theorem 7.2. Fix an anonymous distributed uniformity tester $\mathcal{A}$ with sample complexity s per node, which achieves error probability at most $1/3$. Let us find lower bounds on δ and α, such that each network node must be a (δ, α)-gap tester.

When the distribution is uniform, the probability that the network accepts must satisfy $(1-\delta)^k \ge 2/3$, so we must have $\delta \le 1 - (2/3)^{1/k} \le \ln(2/3)\frac{1}{k}$. On the other hand, when the distribution is ϵ-far from uniform, the probability that the network rejects should be $(1-\alpha\delta)^k \le 1/3$, and so $\alpha\delta \ge 1 - (1/3)^{1/k} \ge \ln(\sqrt{3})(1/k)$. We see that we must have $\alpha \ge (\ln(\sqrt{3})(1/k))/\delta > 5/4$. We now plug in these parameters into Corollary 7.4, and see that each node must draw $\Omega(\sqrt{n/k})$ samples. □

REFERENCES

[1] Jayadev Acharya, ClÃĪment L. Canonne, and Himanshu Tyagi. 2018. *Distributed Simulation and Distributed Inference.* Technical Report arXiv:1804.06952. ArXiV. https://arxiv.org/abs/1804.06952

[2] A. Ambainis. 1996. Communication complexity in a 3-computer model. *Algorithmica* 16, 3 (1996), 298–301.

[3] L. Babai and P. G. Kimmel. 1997. Randomized Simultaneous Messages: Solution Of A Problem Of Yao In Communication Complexity. In *Proceedings of the 12th Annual IEEE Conference on Computational Complexity (CCC '97).* IEEE Computer Society, 239–.

[4] Tugkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D. Smith, and Patrick White. 2000. Testing that distributions are close. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA.* 259–269.

[5] Eric Blais, Clément L. Canonne, and Tom Gur. 2017. Distribution Testing Lower Bounds via Reductions from Communication Complexity. In *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia.* 28:1–28:40.

[6] Ralph Bottesch, Dmitry Gavinsky, and Hartmut Klauck. 2015. Equality, Revisited. In *Mathematical Foundations of Computer Science 2015 - 40th International Symposium, MFCS 2015, Milan, Italy, August 24-28, 2015, Proceedings, Part II.* 127–138.

[7] J. Bourgain and A. Wigderson. [n. d.]. Personal communication (see [3]). ([n. d.]).

[8] Clément L. Canonne. 2015. A Survey on Distribution Testing: Your Data is Big. But is it Blue? *Electronic Colloquium on Computational Complexity (ECCC)* 22 (2015), 63.

[9] Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing).* Wiley-Interscience.

[10] Ilias Diakonikolas and Daniel M. Kane. 2016. A New Approach for Testing Properties of Discrete Distributions. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA.* 685–694.

[11] Eldar Fischer. 2001. The Art of Uninformed Decisions. *Bulletin of the EATCS* 75 (2001), 97.

[12] Pierre Fraigniaud, Amos Korman, and David Peleg. 2011. Local Distributed Decision. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011.* 708–717.

[13] Goldreich and Ron. 2002. Property Testing in Bounded Degree Graphs. *Algorithmica* 32, 2 (01 Feb 2002), 302–343.

[14] Oded Goldreich. 1998. Combinatorial property testing – a survey. *Randomization Methods in Algorithm Design* (1998).

[15] Oded Goldreich. 2016. The uniform distribution is complete with respect to testing identity to a fixed distribution. *Electronic Colloquium on Computational Complexity (ECCC)* 23 (2016), 15.

[16] Oded Goldreich. 2017. *Introduction to Property Testing.* Cambridge University Press.

[17] Oded Goldreich and Dana Ron. 2000. On Testing Expansion in Bounded-Degree Graphs. *Electronic Colloquium on Computational Complexity (ECCC)* 7, 20 (2000).

[18] Michael J. Wiener. 2005. Bounds on Birthday Attack Times. 2005 (01 2005), 318.

[19] Ilan Newman and Mario Szegedy. 1996. Public vs. Private Coin Flips in One Round Communication Games (Extended Abstract). In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing (STOC '96).* 561–570.

[20] C.D. Olds. 1950. Odd and even derangements, Solution E907. *Amer. Math. Monthly* 57 (1950).

[21] L. Paninski. 2008. A Coincidence-Based Test for Uniformity Given Very Sparsely Sampled Discrete Data. *IEEE Transactions on Information Theory* 54, 10 (2008), 4750–4755.

[22] Dana Ron. 2009. Algorithmic and Analysis Techniques in Property Testing. *Foundations and Trends in Theoretical Computer Science* 5, 2 (2009), 73–205.