Given samples from two distributions over an \( n \)-element set, we wish to test whether these distributions are statistically close. We present an algorithm which uses sublinear in \( n \), specifically, \( O(n^{2/3} \epsilon^{-8/3} \log n) \), independent samples from each distribution, runs in time linear in the sample size, makes no assumptions about the structure of the distributions, and distinguishes the cases when the distance between the distributions is small (less than \( \max\{\epsilon^{4/3} n^{-1/3}/32, \epsilon n^{-1/2}/4\} \)) or large (more than \( \epsilon \)) in \( \ell_1 \) distance. This result can be compared to the lower bound of \( \Omega(n^{2/3} \epsilon^{-2/3}) \) for this problem given by Valiant [2008].

Our algorithm has applications to the problem of testing whether a given Markov process is rapidly mixing. We present sublinear algorithms for several variants of this problem as well.


General Terms: Algorithms, Theory

Additional Key Words and Phrases: Testing properties of distributions, statistical distance, testing Markov chains for mixing


T. Batu, Department of Mathematics, London School of Economics and Political Science, London, UK. Email: t.batu@lse.ac.uk.

L. Fortnow, Department of Electrical Engineering and Computer Science, Northwestern University, Chicago, IL, USA. Email: fortnow@eecs.northwestern.edu. Research done while at NEC Research Institute.

R. Rubinfeld, CSAIL, MIT, Cambridge, MA, USA and the Blavatnik School of Computer Science, Tel Aviv University. Email: ronitt@csail.mit.edu. Research done while at NEC Research Institute.

W.D. Smith, 21 Shore Oaks Drive, Stony Brook, NY, USA. Email: warren.wds@gmail.com. Research done while at NEC Research Institute.

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

© 20YY ACM 0004-5411/20YY/0100-0001 $5.00

1. INTRODUCTION

Suppose we have two distributions over the same \( n \)-element set, such that we know nothing about their structure and the only access we have to these distributions is the ability to take independent samples from them. Suppose further that we want to know whether these two distributions are close to each other in \( \ell_1 \) norm: namely, for distribution vectors \( \mathbf{p} \) and \( \mathbf{q} \), whether \( \| \mathbf{p} - \mathbf{q} \|_1 = \sum_i |p_i - q_i| \leq \epsilon. \)

A first approach, which we refer to as the naive approach, would be to sample at least a linear number of elements from each distribution so that we can approximate the distribution and then compare the approximations.

In this paper, we develop a method for distinguishing between the case that the distance between two distributions is greater than \( \epsilon \) and the case that they are significantly closer, using considerably fewer samples. If the distributions have \( \ell_1 \) distance at most \( \max\{\epsilon^{3/4} n^{-1/3}/32, \epsilon n^{-1/2}/4\} \), then the algorithm will accept with probability at least \( 1 - \delta \). If the distributions have \( \ell_1 \) distance more than \( \epsilon \) then the algorithm will accept with probability at most \( \delta \). The number of samples used is \( O(n^{2/3} \epsilon^{-8/3} \log(n/\delta)) \). This is almost optimal (in terms of the dependence on \( n \)) as recently proven by Valiant [2008]; the methods of Valiant, fixing the incomplete arguments in the preliminary version of the current paper [Batu et al. 2000] (see Section 3), yield an \( \Omega(n^{2/3} \epsilon^{-2/3}) \) lower bound for testing \( \ell_1 \) distance in this model. Another result in Valiant’s paper gives an almost linear lower bound on the sample complexity of distinguishing the case that a pair of distributions have \( \ell_1 \) distance less than \( \epsilon_1 \) and the case that a pair of distributions have \( \ell_1 \) distance greater than \( \epsilon_2 \), justifying the weaker choice of “close” pairs in our problem description.

Our test relies on a procedure for distinguishing between the case that two distributions have small \( \ell_2 \) distance (at most \( \epsilon/2 \)) and the case that they have large \( \ell_2 \) distance (at least \( \epsilon \)). This procedure can be implemented by using a sample whose size does not depend on \( n \). However, small \( \ell_2 \) distance does not in general give a good measure of the closeness of two distributions according to \( \ell_1 \) distance. For example, two distributions can have disjoint support and still have \( \ell_2 \) distance of \( O(1/\sqrt{n}) \). Still, we can get a very good estimate of the \( \ell_2 \) distance, say to within \( O(1/\sqrt{n}) \) additive error, and then use the fact that the \( \ell_1 \) distance is at most \( \sqrt{n} \) times the \( \ell_2 \) distance. Unfortunately, the number of queries required by this approach is too large in general. To overcome this difficulty, our \( \ell_1 \) test distinguishes between two cases.

For distributions with small \( \ell_2 \) norm, we show how to use the \( \ell_2 \) test to get an efficient test for \( \ell_1 \) distance. For distributions with large \( \ell_2 \) norm, we use the fact that such distributions must have elements which occur with relatively high probability. We create a filtering test that partitions the domain into those elements with relatively high probability and all the other elements (those with relatively low probability). The test estimates the \( \ell_1 \) distance due to these high-probability elements directly, using the naive approach mentioned above. The test then approximates the \( \ell_1 \) distance due to the low-probability elements using the test for \( \ell_2 \) distance. Optimizing the notion of “high probability” yields our \( O(n^{2/3} \epsilon^{-8/3} \log(n/\delta)) \) algorithm. The \( \ell_2 \) distance test uses \( O(\epsilon^{-4} \log(1/\delta)) \) samples.

\footnote{Half of \( \ell_1 \) distance between two distributions is also referred to as total variation distance.}
Applying our techniques to Markov chains, we use the above algorithm as a basis for constructing tests for determining whether a Markov chain is rapidly mixing. We show how to test whether iterating a Markov chain for \( t \) steps causes it to reach a distribution close to the stationary distribution. Our testing algorithm works by following \( \tilde{O}(tn^{5/3}) \) edges in the chain. When the Markov chain is dense enough and represented in a convenient way (such a representation can be computed in linear time and we give an example representation in Section 4), this test remains sublinear in the size of the Markov chain for small \( t \). We then investigate two notions of being close to a rapidly mixing Markov chain that fall within the framework of property testing, and show how to test that a given Markov chain is close to a Markov chain that mixes in \( t \) steps by following only \( \tilde{O}(tn^{2/3}) \) edges. In the case of Markov chains that come from directed graphs and pass our test, our theorems show the existence of a directed graph that is both close to the original one and rapidly mixing.

1.1 Related Work

1.1.1 Testing Properties of Distributions. The use of collision statistics in a sample has been proposed as a technique to test whether a distribution is uniform (see, for example, Knuth [1973]). Goldreich and Ron [2000] give the first formal analysis that using \( O(\sqrt{n}) \) samples to estimate the collision probability yields an algorithm which gives a very good estimate of the \( \ell_2 \) distance between the given distribution and the uniform distribution. Their “collision count” idea underlies the present paper. More recently, Paninski [2008] presents a test to determine whether a distribution is far from the uniform distribution with respect to \( \ell_1 \) distance using \( \Theta(\sqrt{n}/\epsilon^2) \) samples, characterizing the exact dependence on \( \epsilon \) via a lower bound. Ma [1981] also uses collisions to measure the entropy of a distribution defined by particle trajectories. After the publication of the preliminary version of this paper [Batu et al. 2000], a long line of publications appeared regarding testing properties of distributions including independence, entropy, and monotonicity (see, for example, [Batu et al. 2001; Batu et al. 2004; Batu et al. 2005; Brautbar and Samorodnitsky 2007; Alon et al. 2007; Valiant 2008; Rubinfeld and Servedio 2009; Raskhodnikova et al. 2009; Rubinfeld and Xie 2010; Adamsaszek et al. 2010]).

1.1.2 Expansion, Rapid Mixing, and Conductance. Goldreich and Ron [2000] present a test that they conjecture can be used to give an algorithm with \( O(\sqrt{n}) \) query complexity which tests whether a regular graph is close to being an expander. A graph is close to being an expander if it can be turned into an expander by changing a small fraction of its edges. Their test is based on picking a random node and testing whether random walks from this node reach a distribution that is close to the uniform distribution on the nodes. Our tests for Markov chains are based on similar principles. Mixing and expansion are known to be related [Sinclair and Jerrum 1989], but our techniques only apply to the mixing properties of random walks on directed graphs, since the notion of closeness we use does not preserve the symmetry of the adjacency matrix. More recently, a series of papers [Czumaj and Sohler 2007; Kale and Seshadhri 2008; Nachmias and Shapira 2007] answer Goldreich and Ron’s conjecture in the affirmative. In a previous work, Goldreich and Ron [2002] show that testing that a graph is close to an expander requires

Ω(\(n^{1/2}\)) queries.

The conductance [Sinclair and Jerrum 1989] of a graph \(G = (V,E)\) is defined as the minimum \(C_S/E_S\), where minimum is taken over \(S \subseteq V\) such that \(|S| \leq |V|/2\), \(C_S\) is the number of edges in the \((S,V \setminus S)\) cut, and \(E_S\) is the number of edges with both endpoints in \(S\). It is known to be closely related to expansion and rapid-mixing properties of the graph [Kannan 1994; Sinclair and Jerrum 1989]. Frieze and Kannan [1999] show, given a graph \(G\) with \(n\) vertices and \(\alpha\), one can approximate the conductance of \(G\) to within additive error \(\alpha\) in time \(n \cdot 2^{O(1/\alpha^2)}\). Their techniques also yield a \(2^{\text{poly}(1/\epsilon)}\)-time test that determines whether the adjacency matrix of a graph can be changed in at most an \(\epsilon\)-fraction of the locations to get a graph with high conductance. However, for the purpose of testing whether an \(n\)-vertex, \(m\)-edge graph is rapid mixing, we would need to approximate its conductance to within \(\alpha = O(m/n^2)\); thus, only when \(m = \Theta(n^2)\), would the algorithm in [Frieze and Kannan 1999] run in \(O(n)\) time.

We now discuss some other known results for testing rapid mixing through eigenvalue computations. It is known that mixing [Sinclair and Jerrum 1989; Kannan 1994] is related to the separation between the two largest eigenvalues [Alon 1986]. Standard techniques for approximating the eigenvalues of a dense \(n \times n\) matrix run in \(\Theta(n^3)\) floating-point operations and consume \(\Theta(n^2)\) words of memory [Golub and van Loan 1996]. However, for a sparse \(n \times n\) symmetric matrix with \(m\) nonzero entries, \(n \leq m\), “Lanczos algorithms” [Parlett 1998] accomplish the same task in \(\Theta(n(m + \log n))\) floating-point operations, consuming \(\Theta(n + m)\) storage. Furthermore, it is found in practice that these algorithms can be run for far fewer, even a constant number, of iterations while still obtaining highly accurate values for the outer and inner few eigenvalues.

### 1.1.3 Streaming

There is much work on the problem estimating the distance between distributions in data streaming models where space rather than time is limited (c.f., [Gibbons and Matias 1999; Alon et al. 1999; Feigenbaum et al. 1999; Fong and Strauss 2000]). Another line of work [Broder et al. 2000] estimates the distance in frequency count distributions on words between various documents, where again space is limited. Guha et al. [2009] have extended our result to estimating the closeness of distribution with respect to a range of \(f\)-divergences, which include \(\ell_1\) distance. They also have adapted our work to a streaming data setting and extended it. Testing distributions in streaming data models has been an active area of research in the recent years (see, for example, [Bhuvanagiri and Ganguly 2006; Chakrabarti et al. 2006; Indyk and McGregor 2008; Guha et al. 2008; Chakrabarti et al. 2010; Chien et al. 2010; Braverman and Ostrovsky 2010a; 2010b]).

### 1.1.4 Other Related Models

In an interactive setting, Sahai and Vadhan [1997] show that, given distributions \(p\) and \(q\) generated by polynomial-size circuits, the problem of distinguishing whether \(p\) and \(q\) are close or far in \(\ell_1\) norm is complete for statistical zero knowledge. Kannan and Yao [1991] outline a program checking framework for certifying the randomness of a program’s output. In their model, one does not assume that samples from the input distribution are independent.

### 1.1.5 Related Notions in Statistics and Information Theory

There is a vast literature on testing statistical hypotheses. In these works, one is given examples
chosen from the same distribution out of two possible choices, say \( p \) and \( q \). The goal is to decide which of two distributions the examples are coming from. More generally, the goal can be stated as deciding which of two known classes of distributions contains the distribution generating the examples. This can be seen to be a generalization of our model as follows: Let the first class of distributions be the set of distributions of the form \( q \times q \). Let the second class of distributions be the set of distributions of the form \( q_1 \times q_2 \) where the \( \ell_1 \) difference of \( q_1 \) and \( q_2 \) is at least \( \epsilon \). Then, given examples from two distributions \( p_1, p_2 \), create a set of example pairs \((x, y)\) where \( x \) is chosen according to \( p_1 \) and \( y \) according to \( p_2 \) independently. Note that the second class of distributions is defined in terms of parameter \( \epsilon \) and we study the dependency of the sample complexity of the task on this parameter as well as the domain size. Bounds and an optimal algorithm for the general problem for various distance measures are given in [Cover and Thomas 1991; Neyman and Pearson 1933; Cressie and Morgan 1989; Csiszár 1967; Lehmann 1986]. None of these give sublinear bounds in the domain size for our problem. The specific model of singleton hypothesis classes is studied by Yamanishi [1995].

1.2 Notation

We use the following notation. We denote the set \( \{1, \ldots, n\} \) with \([n]\). The notation \( x \in_R [n] \) denotes that \( x \) is chosen uniformly at random from the set \([n]\). The \( \ell_1 \) norm of a vector \( v \) is denoted by \( \|v\|_1 \) and is equal to \( \sum_{i=1}^{n} |v_i| \). Similarly, the \( \ell_2 \) norm is denoted by \( \|v\|_2 \) and is equal to \( \sqrt{\sum_{i=1}^{n} v_i^2} \), and \( \ell_\infty \) norm \( \|v\|_\infty = \max_i |v_i| \). We assume our distributions are discrete distributions over \( n \) elements, with labels in \([n]\), and will represent such a distribution as a vector \( p = (p_1, \ldots, p_n) \), where \( p_i \) is the probability of outputting element \( i \). Throughout the paper, we use parameter \( \epsilon \) for \( \ell_1 \) distance, where \( 0 < \epsilon < 2 \).

The collision probability of two distributions \( p \) and \( q \) is the probability that a sample from each of \( p \) and \( q \) yields the same element. Note that, for two distributions \( p, q \), the collision probability is \( p \cdot q = \sum_i p_i q_i \). To avoid ambiguity, we refer to the collision probability of \( p \) and \( q \) as the self-collision probability of \( p \). Note that the self-collision probability of \( p \) is \( \|p\|_2^2 \).

2. TESTING CLOSENESS OF DISTRIBUTIONS

The main goal of this section is to show how to test whether two distributions \( p \) and \( q \) are close in \( \ell_1 \) norm in sublinear time in the size of the domain of the distributions. We are given access to these distributions via black boxes which upon a query respond with an element of \([n]\) generated according to the respective distribution. Our main theorem is:

**Theorem 2.1.** Given parameters \( \delta \) and \( \epsilon \), and sample access to distributions \( p, q \) over a set of \( n \) elements, there is a test which runs in time \( O(n^{3/3} e^{-8/3 \log(n/\delta)}) \) such that, if \( \|p - q\|_1 \leq \max(\frac{\sqrt{n}}{3\log(n/\delta)}, \frac{\epsilon}{3\log(n/\delta)}) \), then the test accepts with probability at least \( 1 - \delta \) and, if \( \|p - q\|_1 > \epsilon \), then the test rejects with probability at least \( 1 - \delta \).

In order to prove this theorem, we give a test which determines whether \( p \) and \( q \) are close in \( \ell_2 \) norm. The test is based on estimating the self-collision and collision probabilities of \( p \) and \( q \). In particular, if \( p \) and \( q \) are close, one would expect that

---

the self-collision probabilities of each are close to the collision probability of the pair. Formalizing this intuition, in Section 2.1, we prove:

**Theorem 2.2.** Given parameter δ and ϵ, and sample access to distributions p and q over a set of n elements, there exists a test such that, if ∥p − q∥2 ≤ ϵ/2, then the test accepts with probability at least 1 − δ and, if ∥p − q∥2 > ϵ, then the test rejects with probability at least 1 − δ. The running time of the test is \(O(\epsilon^{-4}\log(1/\delta))\).

The test used to prove Theorem 2.2 is given in Figure 1. The number of pairwise self-collisions in multiset \(F = \{F^1, F^2, \ldots, F^m\}\) is the count of \(i < j\) such that \(F^i = F^j\). Similarly, the number of collisions between multisets \(Q_p = \{Q^1_p, Q^2_p, \ldots\}\) and \(Q_q = \{Q^1_q, Q^2_q, \ldots\}\) is the count of \((i,j)\) such that \(Q^i_p = Q^i_q\).

We use the parameter \(m\) to indicate the number of samples used by the test to get constant confidence. In order to bound the \(\ell_2\) distance between \(p\) and \(q\) by \(\epsilon\), setting \(m = O(\frac{1}{\epsilon^2})\) suffices. By maintaining arrays which count the numbers of times, for example, \(N_p(i)\) for \(F_p\), that each element \(i\) is sampled and summing \({m \choose 2}\) over all sampled \(i\) in the domain, one can achieve the claimed running time bounds for computing an estimate of the collision probability. Although sparse arrays of size \(n\) would be needed for counting, we can use a well-known trick for initializing an array in constant time [Aho et al. 1974, page 71] to avoid incurring a heavy running time cost. In this way, essentially \(m^2\) terms can be summed up to give an estimation of the collision probability in \(O(m)\) time.

Since \(\|v\|_1 \leq \sqrt{n} \cdot \|v\|_2\), a simple way to extend the above test to an \(\ell_1\) distance test is by setting \(\epsilon' = \epsilon / \sqrt{n}\). This would give the correct output behavior for the tester. Unfortunately, due to the order of the dependence on \(\epsilon\) in the \(\ell_2\) distance test, the resulting running time is quadratic in \(n\). It is possible, though, to achieve sublinear running times if the input distributions are known to be reasonably evenly distributed. We make this precise by a closer analysis of the variance of the estimator in the test in Lemma 2.5. In particular, we analyze the dependence of the variances of \(s\) and \(r\) on the parameter \(b = \max(\|p\|_{\infty}, \|q\|_{\infty})\). There we show that, if \(b = O(n^{-\alpha})\), then it suffices to run the \(\ell_2\)-Distance-Test with a sample size of \(m = \Theta(\epsilon^{-4}(n^{1-\alpha/2} + m^{2-2\alpha}))\) in order to distinguish between the case that \(\|p - q\|_2 \leq \epsilon' / 2 = \frac{\epsilon'}{2\sqrt{n}}\) and the case that \(\|p - q\|_2 > \epsilon'\) (rather than \(m = \Theta((1/\epsilon')^4))\). Thus, when the maximum probability of any element is bounded, the \(\ell_2\) distance test can in fact yield a sublinear-time algorithm for testing closeness in \(\ell_1\) distance.
\( \ell_1 \text{-Distance-Test}(p, q, \epsilon, \delta) \)

1. Let \( b = (\epsilon/n)^{2/3} \).
2. Let \( M = O((\epsilon^{-8/3})n^{2/3} \log(n/\delta)) \). Take \( M \) samples from \( p \) and \( M \) samples from \( q \).
3. Let \( S_p \) and \( S_q \) be the sample multisets obtained from \( p \) and \( q \), respectively, by discarding elements that occur less than \((1 - \epsilon/26)Mb\) times.
4. If \( S_p \cup S_q \neq \emptyset \),
   i. Let \( f_p^i \) (resp., \( f_q^i \)) be the number of times element \( i \) appears in \( S_p \) (resp., \( S_q \)).
   ii. Reject if \( \sum_{i \in S_p \cup S_q} |f_p^i - f_q^i| > \epsilon M/8 \).
5. Define \( p' \) as follows: Sample an element from \( p \). If this sample is not in \( S_p \cup S_q \), output it; otherwise, output an \( x \in \mathbb{R} \). Define \( q' \) similarly.
6. Call \( \ell_2 \)-Distance-Test\((p', q', cn^{2/3}/\epsilon^{8/3}, \frac{\sqrt{\epsilon}}{2\sqrt{n}}, \delta/2)\) for a sufficiently large constant \( c \).

Fig. 2. Algorithm \( \ell_1 \)-Distance-Test

In the previous paragraph, we have noted that, for distributions with a bound on the maximum probability of any element, it is possible to test closeness with time and queries sublinear in the domain size. On the other hand, when the minimum probability element is quite large, the naive approach that we referred to in the introduction can be significantly more efficient. This suggests a filtering algorithm, which separates the domain of the distributions being tested into two parts – the big elements, or those elements to which the distributions assign relatively high probability weight, and the small elements, which are all other elements. Then, the naïve tester is applied to the distributions restricted to the big elements, and the tester that is based on estimating the \( \ell_2 \) distance is applied to the distributions restricted to the small elements.

More specifically, we use the following definition to identify the elements with large weights.

**Definition 2.3.** *Big element* An element \( i \) is called *big with respect to a distribution* \( p \) if \( p_i > (\epsilon/n)^{2/3} \).

The complete test is given in Figure 2. The proof of Theorem 2.1 is presented in Section 2.2.

### 2.1 Closeness in \( \ell_2 \) Norm

In this section, we analyze Algorithm \( \ell_2 \)-Distance-Test and prove Theorem 2.2. The statistics \( r_p, r_q \) and \( s \) in Algorithm \( \ell_2 \)-Distance-Test are estimators for the self-collision probability of \( p \), of \( q \), and of the collision probability between \( p \) and \( q \), respectively. If \( p \) and \( q \) are statistically close, we expect that the self-collision probabilities of each are close to the collision probability of the pair. These probabilities are exactly the inner products of these vectors. In particular, if the multiset \( F_p \) of samples from \( p \) is given by \( \{ F_p^1, \ldots, F_p^m \} \), then, for any pair \( i, j \in [m], i \neq j \), we have that \( \Pr [F_p^i = F_p^j] = p \cdot p = \|p\|_2^2 \). By combining these statistics, we show that \( r - s \) is an estimator for the desired value \( \|p - q\|_2^2 \).

In order to analyze the number of samples required to estimate \( r - s \) to a high enough accuracy, we must also bound the variance of the variables \( s \) and \( r \) used in the test. One distinction to make between self-collisions and collisions between \( p \) and \( q \) is that, for the self-collisions, we only consider samples for which \( i \neq j \), but this is not necessary for the collisions between \( p \) and \( q \). We accommodate this in our algorithm by scaling \( r_p \) and \( r_q \) appropriately. By this scaling and from the
above discussion we see that \( E[\delta] = 2m^2(p \cdot q) \) and that \( E[r - \delta] = m^2(\|p\|^2 + \|q\|^2 - 2(p \cdot q)) = m^2(\|p - q\|^2) \).

A complication which arises from this scheme is that the pairwise samples are not independent. We use Chebyshev’s inequality to bound the quality of the approximation, which in turn requires that we give a bound on the variance, as we do in this section.

Our techniques extend the work of Goldreich and Ron [2000], where self-collision probabilities are used to estimate \( \ell_2 \) norm of a vector, and in turn the deviation of a distribution from uniform. In particular, their work provides an analysis of the statistics \( r_p \) and \( r_q \) above through the following lemma.

**Lemma 2.4 [Goldreich and Ron 2000].** Consider the random variable \( r_p \) in Algorithm \( \ell_2 \)-Distance-Test. Then, \( E[r_p] = \left(\frac{n}{m}\right)^2 \|p\|^2 \) and \( Var(r_p) \leq 2E[r_p]^{3/2} \).

We next present a tighter variance bound given in terms of the largest weight in \( p \) and \( q \).

**Lemma 2.5.** There is a constant \( c \) such that

\[
Var(r_p) \leq m^2\|p\|^2 + m^3\|p\|^3 \leq c(m^3b^2 + m^2b),
\]

\[
Var(r_q) \leq m^2\|q\|^2 + m^3\|q\|^3 \leq c(m^3b^2 + m^2b), \text{ and }
\]

\[
Var(s) \leq c(m^3b^2 + m^2b),
\]

where \( b = \max(\|p\|_{\infty}, \|q\|_{\infty}) \).

**Proof.** Let \( I \) be the set \( \{1, \ldots, m\} \). For \( (i, j) \in I \times I \), define the indicator variable \( C_{i,j} = 1 \) if the \( i \)th element of \( Q_p \) and the \( j \)th element of \( Q_q \) are the same. Then, the variable from the algorithm \( s_{pq} = \sum_{i,j} C_{i,j} \). Also define the notation \( \bar{C}_{i,j} = C_{i,j} - E[C_{i,j}] \). Given these definitions, we can write

\[
Var \left( \sum_{(i,j) \in I \times I} C_{i,j} \right) = E \left[ \left( \sum_{(i,j) \in I \times I} C_{i,j} \right)^2 \right] = E \sum_{(i,j) \in I \times I} (\bar{C}_{i,j})^2 + 2 \sum_{(i,j) \neq (k,l) \in I \times I} \bar{C}_{i,j}\bar{C}_{k,l}
\]

\[
\leq E \sum_{(i,j) \in I \times I} C_{i,j} + 2 \cdot E \sum_{(i,j) \neq (k,l) \in I \times I} \bar{C}_{i,j}\bar{C}_{k,l}
\]

\[
= m^2(p \cdot q) + 2 \cdot E \sum_{(i,j) \neq (k,l) \in I \times I} \bar{C}_{i,j}\bar{C}_{k,l}
\]

To analyze the last expectation, we use two facts. First, it is easy to see, by the definition of covariance, that \( E[C_{i,j}\bar{C}_{k,l}] = E[C_{i,j}C_{k,l}] - E[C_{i,j}]E[C_{k,l}] \). Secondly, we note that \( C_{i,j} \) and \( C_{k,l} \) are not independent only when \( i = k \) or \( j = l \). For \( C_{i,j} \) and \( C_{k,l} \) independent, we have that \( E[C_{i,j}\bar{C}_{k,l}] = E[C_{i,j}]E[C_{k,l}] = 0 \). Expanding

the sum, we get
\[ E \left[ \sum_{(i,j),(k,l) \in I \times I, (i,j) \neq (k,l)} C_{i,j}C_{k,l} \right] = E \left[ \sum_{(i,j),(k,l) \in I \times I, j \neq l} C_{i,j}C_{i,l} + \sum_{(i,j),(k,l) \in I \times I, i \neq k} C_{i,j}C_{k,j} \right] \]
\[ \leq E \left[ \sum_{(i,j),(k,l) \in I \times I, j \neq l} C_{i,j}C_{i,l} + \sum_{(i,j),(k,l) \in I \times I, i \neq k} C_{i,j}C_{k,j} \right] \]
\[ \leq cm^3 \sum_{t \in [n]} (p_t q_t^2 + p_t^2 q_t) \leq cm^3 b^2 \sum_{t \in [n]} q_t \leq 2cm^3 b^2 \]
for some constant \( c \). Next, we bound \( \text{Var} (r) \) similarly to \( \text{Var} (s) \) using the argument in the proof of Lemma 2.4 from [Goldreich and Ron 2000]. Consider an analogous calculation to the preceding inequality for \( \text{Var} (r_p) \) (similarly, for \( \text{Var} (r_q) \)) where \( X_{ij} = 1 \) for \( 1 \leq i < j \leq m \) if \( F^i_p = F^j_p \). Similarly to above, define \( \bar{X}_{ij} = X_{ij} - E[X_{ij}] \). Then, we get
\[ \text{Var} (r_p) = E \left[ \left( \sum_{1 \leq i < j \leq m} \bar{X}_{ij} \right)^2 \right] \]
\[ = \sum_{1 \leq i < j \leq m} E \left[ \bar{X}_{ii}^2 \right] + 4 \sum_{1 \leq i < j < k \leq m} E \left[ \bar{X}_{ij} \bar{X}_{ik} \right] \]
\[ \leq \left( \frac{m}{2} \right) \cdot \sum_{t \in [n]} p_t^2 + 4 \left( \frac{m}{3} \right) \sum_{t \in [n]} p_t^3 \]
\[ \leq O(m^2) \cdot b + O(m^3) \cdot b^2. \]
Thus, we get the upper bound for both variances. \( \square \)

**Corollary 2.6.** There is a constant \( c \) such that \( \text{Var} (r - s) \leq c(m^3b^2 + m^2b) \), where \( b = \max(\|p\|_\infty, \|q\|_\infty) \).

**Proof.** Since variance is additive for independent random variables, we get \( \text{Var} (r - s) \leq c(m^3b^2 + m^2b) \). \( \square \)

Finally, we can analyze the behavior of the algorithm.

**Theorem 2.7.** Let \( p \) and \( q \) be two distributions such that \( b = \max(\|p\|_\infty, \|q\|_\infty) \) and let \( m = \Omega((b^2 + e^2/\sqrt{b})/\epsilon) \). If \( \|p - q\|_2 \leq \epsilon/2 \), then \( f_2\text{-Distance-Test}(p, q, m, \epsilon, \delta) \) accepts with probability at least \( 1 - \delta \). If \( \|p - q\|_2 > \epsilon \), then \( f_2\text{-Distance-Test}(p, q, m, \epsilon, \delta) \) accepts with probability less than \( \delta \). The running time is \( O(m \log(1/\delta)) \).

**Proof.** For our statistic \( A = (r - s) \), we can say, using Chebyshev’s inequality and Corollary 2.6, that for some constant \( c \),
\[ \Pr \left[ |A - E[A]| > \rho \right] \leq \frac{c(m^3b^2 + m^2b)}{\rho^2}. \]
Recalling that $\mathbb{E} [A] = m^2(\|p - q\|_2^2)$, we observe that the $\ell_2$-**Distance-Test** can distinguish between the cases $\|p - q\|_2 \leq \epsilon/2$ and $\|p - q\|_2 > \epsilon$ if $A$ is within $m^2\epsilon^2/4$ of its expectation. We can bound the error probability by

$$\Pr [\|A - \mathbb{E} [A]\| > m^2\epsilon^2/4] \leq \frac{16c(m^3b^2 + m^2b)}{m^4\epsilon^4}. $$

Thus, for $m = \Omega((b^2 + \epsilon^2\sqrt{b})/\epsilon^4)$, the probability above is bounded by a constant. This error probability can be reduced to $\delta$ by $O(\log(1/\delta))$ repetitions. \hfill $\square$

### 2.2 Closeness in $\ell_1$ Norm

Note that, by plugging $\epsilon = \epsilon/\sqrt{n}$ in Theorem 2.7, we get the following result as an immediate corollary.

**Corollary 2.8.** Given two black-box distributions $p, q$ over $[n]$, with $\|p\|_\infty \leq \|q\|_\infty$, there is a test requiring $O((n^2\|p\|_2^2\epsilon^{-4} + n\sqrt{\|q\|_\infty\epsilon^{-2}})\log(1/\delta))$ samples that (1) if $\|p - q\|_1 \leq \epsilon/2\sqrt{n}$, it accepts with probability at least $1 - \delta$ and (2) if $\|p - q\|_1 > \epsilon$, it rejects with probability at least $1 - \delta$.

Recently, Levi et al. [2011] improve this result to achieve a sample complexity of $O((n^2\|p\|_\infty^3\|q\|_\infty\epsilon^{-4} + n\sqrt{\|p\|_\infty\epsilon^{-2}})\log(1/\delta))$, by looking at the dependence of the variance calculation in Section 2.1 on the $\ell_\infty$ norms of each distribution.

For the rest of this section, we analyze the $\ell_1$-**Distance-Test**, which achieves a near-optimal sample complexity regardless of the $\ell_\infty$ norms of the input distributions. The $\ell_1$-**Distance-Test** proceeds in two phases. The first phase of the algorithm (Steps 1–4) determines which elements of the domain are the big elements (as defined in Definition 2.3) and estimates their contribution to the distance $\|p - q\|_1$. In this first phase, some elements which are not big but close to being big might also be determined as big. The second phase (Steps 5 and 6) filters out the big elements and invokes the $\ell_2$-**Distance-Test** on the filtered distribution with closeness parameter $\epsilon/(2\sqrt{n})$. The correctness of this subroutine call is given by Theorem 2.7 with $b = 2e^{2/3}n^{-2/3}$. With these substitutions, the number of samples $m$ is $O(e^{-8/3}n^{2/3})$. The choice of threshold $b$ in $\ell_1$-**Distance-Test** for the weight of the big elements arises from optimizing the running-time trade-off between the two phases of the algorithm.

We need to show that by using a sample of size $O(e^{-8/3}n^{2/3}\log(n/\delta))$, we can estimate the weights of each of the big elements to within a multiplicative factor of $1 + O(\epsilon)$, with probability at least $1 - \delta/2$.

**Lemma 2.9.** Let $b = e^{2/3}n^{-2/3}$. There exists $M_0 = O(\frac{n^{2/3}\log(n/\delta)}{\epsilon^{9/2}})$ that satisfies the following: In $\ell_1$-**Distance-Test**, given $M \geq M_0$ samples from a distribution $p$, we define $\bar{p}_i = f_p^n / M$. Then, with probability at least $1 - \delta/2$, the following hold for all $i$: (1) if $p_i \geq (1 - \epsilon/13)b$, then $|\bar{p}_i - p_i| < \frac{\epsilon}{20} \max(p_i, b)$, (2) if $p_i < (1 - \epsilon/13)b$, then $\bar{p}_i < (1 - \epsilon/26)b$.

**Proof.** We analyze two cases (divided somewhat differently from the lemma statement). We use Chernoff bounds to show that, for each $i$, the following holds:
If \( p_i > b \), then
\[
\Pr[|\hat{p}_i - p_i| > \epsilon p_i / 26] < \exp(-O(\epsilon^2 M p_i)) < \exp(-O(\epsilon^2 M b)) \leq \frac{\delta}{2n}.
\]
If \( p_i \leq b \), then
\[
\Pr[|\hat{p}_i - p_i| > \epsilon b / 26] \leq \Pr[|\hat{p}_i - p_i| > \frac{\epsilon b}{26 p_i}]
\[
< \exp(-O(\epsilon^2 b^2 M / p_i))
\[
\leq \exp(-O(\epsilon^2 M b))
\[
\leq \frac{\delta}{2n}.
\]
Note that the case \((1 - \epsilon/13)b \leq p_i \leq b\) is proved by the inequalities above. The lemma follows by the union bound.

Now we are ready to prove our main theorem.

**Theorem 2.10.** For \( \epsilon \geq 1 / \sqrt{n} \), \( \ell_1\)-Distance-Test accepts distributions \( p, q \) such that \( \|p - q\|_1 \leq \max\left(\frac{\sqrt{\epsilon}}{2\epsilon^{1/2}}, \frac{\epsilon}{2\sqrt{n}}\right) \), and rejects when \( \|p - q\|_1 > \epsilon \), with probability at least \( 1 - \delta \). The running time of the test is \( O(\epsilon^{8/3} \cdot n^{2/3} \log(n / \delta)) \).

**Proof.** Suppose items (1) and (2) from Lemma 2.9 hold for all \( i \), and for both \( p \) and \( q \). By Lemma 2.9, this event happens with probability at least \( 1 - \delta/2 \).

Let \( S = S^p \cup S^q \). By our assumption, all the big elements of both \( p \) and \( q \) are in \( S \), and no element that has weight less than \( (1 - \epsilon/13)b \) in both distributions is in \( S \). Let \( \Delta_1 \) be the \( \ell_1 \) distance attributed to the elements in \( S \); that is, \( \sum_{i \in S} |p_i - q_i| \). Let \( \Delta_2 = \|p' - q'\|_1 \) (in the case that \( S \) is empty, \( \Delta_1 = 0, p = p' \) and \( q = q' \)). Note that \( \Delta_1 \leq \|p - q\|_1 \). We can show that \( \Delta_2 \leq \|p - q\|_1 \), and \( \|p - q\|_1 \leq 2\Delta_1 + \Delta_2 \) as follows. Let \( p(S) \) denote the total weight of elements in \( S \) according to \( p \). Then,
\[
\Delta_2 = \sum_{i \in S} |p'_i - q'_i| + \sum_{i \in \bar{S}} |p'_i - q'_i| = \sum_{i \in S} \left| p(S) n - q(S) n \right| + \sum_{i \in \bar{S}} |p_i + p(S) n - q_i - q(S) n|
\[
\leq \sum_{i \in |n|} \left| p(S) n - q(S) n \right| + \sum_{i \in \bar{S}} |p_i - q_i| = |p(S) - q(S)| + \sum_{i \in S} |p_i - q_i| \leq \|p - q\|_1
\]
and
\[
\|p - q\|_1 = \Delta_1 + \sum_{i \in S} \left| p'_i - \frac{p(S)}{n} - q'_i + \frac{q(S)}{n} \right|
\[
\leq \Delta_1 + |q(S) - p(S)| + \sum_{i \in \bar{S}} |p'_i - q'_i| \leq 2\Delta_1 + \Delta_2.
\]

Next, we show that the algorithm estimates \( \Delta_1 \) in a brute-force manner to within an additive error of \( \epsilon/9 \). By Lemma 2.9, the error for element \( i \) in the sum \( \frac{1}{M} \sum_{i \in S} |f^p_i - f^q_i| \) from the algorithm is bounded by
\[
\frac{\epsilon}{26} (\max(p_i, b) + \max(q_i, b)) \leq \frac{\epsilon}{26} (p_i + q_i + 2eb/13),
\]
where the last inequality follows from the fact that \( p_i \) and \( q_i \) are at least \((1-\epsilon/13)b\).

Consider the sum over \( i \in S \) of these error terms. Notice that this sum is over at most \( 2/(1-\epsilon/13)b \) elements in \( S \). Hence, the total additive error is bounded by

\[
\sum_{i \in S} e \frac{\epsilon}{26} (p_i + q_i + 2b/13) \leq \frac{\epsilon}{26} (2 + 4\epsilon/(13 - \epsilon)) \leq \epsilon/9,
\]

where the last inequality holds since \( \epsilon \leq 2 \).

Note that \( \max(||p'||_\infty, ||q'||_\infty) \leq b + n^{-1} \leq 2b \) for \( \epsilon \geq 1/\sqrt{n} \). So, we can use the \( \ell_2 \)-Distance-Test on \( p' \) and \( q' \) with \( m = O(\epsilon^{-8/3} n^{2/3}) \) as shown by Theorem 2.7.

If \( ||p - q||_1 < \frac{\epsilon}{8\sqrt{n}} \), then so are \( \Delta_1 \) and \( \Delta_2 \). The first phase of the algorithm does not cause rejection because of the additive error bound on \( \Delta_1 \) given in (1).

Using a special case of Hölder’s inequality, which states that, for any vector \( v \), \( ||v||_2 \leq ||v||_1 \cdot ||v||_\infty \), we get \( ||p' - q'||_2 \leq \frac{\epsilon}{4\sqrt{n}} \). Therefore, the \( \ell_2 \)-Distance-Test accepts with probability at least \( 1 - \delta/2 \). Similarly, if \( ||p - q||_1 > \epsilon \), then either \( \Delta_1 > \epsilon/4 \) or \( \Delta_2 > \epsilon/2 \). Either the first phase of the algorithm or the \( \ell_2 \)-Distance-Test will reject with high probability.

To verify the running time bound, note that the running time for the first phase is \( O(n^{2/3} \epsilon^{-8/3} \log(n/\delta)) \) and that for \( \ell_2 \)-Distance-Test is \( O(n^{5/3} \epsilon^{-8/3} \log \frac{1}{\delta}) \). It is easy to see that our algorithm makes an error either when it makes a bad estimation of \( \Delta_1 \) or when \( \ell_2 \)-Distance-Test makes an error. So, the probability of error is bounded by \( \delta \). \( \square \)

2.3 Testing \( \ell_1 \) Distance from Uniformity

A special case of Theorem 2.2 gives a constant-time algorithm which provides an additive approximation of the \( \ell_1 \) distance of a distribution from the uniform distribution. For the problem of testing that \( p \) is close to the uniform distribution in \( \ell_1 \) distance (i.e., testing closeness when \( q \) is the uniform distribution), one can get a better sample complexity dependence on \( n \).

**Theorem 2.11.** Given \( \epsilon \leq 1 \) and a black-box distribution \( p \) over \([n]\), there is a test that takes \( O(\epsilon^{-4} \cdot \sqrt{n} \cdot \log(1/\delta)) \) samples, accepts with probability at least \( 1 - \delta \) if \( ||p - U_{[n]}||_1 \leq \epsilon/\sqrt{3n} \), and rejects with probability at least \( 1 - \delta \) if \( ||p - U_{[n]}||_1 > \epsilon \).

The proof of Theorem 2.11 relies on the following lemma, which can be proven using techniques from Goldreich and Ron [2000] (see also Lemma 2.5 in this paper).

**Lemma 2.12.** Given \( \epsilon \leq 1 \) and a black-box distribution \( p \) over \([n]\), there is an algorithm CollisionEstimate that takes \( O(\epsilon^{-2} \cdot \sqrt{n} \cdot \log(1/\delta)) \) samples and estimates \( ||p||_2^2 \) within an additive error of \( \epsilon ||p||_2^2 \), with probability at least \( 1 - \delta \).

**Proof of Lemma 2.12.** Consider the random variable \( r_p \) from the \( \ell_2 \)-Distance-Test. Since \( E[r_p] = \binom{n}{2} \cdot ||p||_2^2 \), we only need to show that it does not deviate from its expectation too much with high probability. Again, using Chebyshev’s inequality and Lemma 2.5,

\[
\Pr[|r_p - E[r_p]| > \epsilon E[r_p]] \leq \frac{O(m^2 ||p||_2^2 + m^4 ||p||_2^2)}{\epsilon^2 m^4 ||p||_2^2} \leq \frac{1}{4},
\]

where the last inequality follows for \( m = O(\epsilon^{-2}\sqrt{n}) \) from the fact that \( ||p||_2 \geq n^{-1/2} \). The confidence can be boosted to \( 1 - \delta \) using \( O(\log(1/\delta)) \) repetitions. \( \square \)
Testing Closeness of Discrete Distributions

Uniformity-Distance-Test \((p, \epsilon, \delta)\)

1. Accept if \(\text{CollisionEstimate}(p, \epsilon^2/5, \delta)\) returns an estimate at most \((1 + 3\epsilon^2/5)/n\).
2. Otherwise, reject.

Fig. 3. Algorithm Uniformity-Distance-Test

We note that, for an additive \(\epsilon\)-approximation of \(\|p\|_2\), an analogous argument to the proof above will yield an algorithm that uses \(O(\epsilon^{-4})\) samples.

Proof of Theorem 2.11. The algorithm, given in Figure 3, estimates \(\|p\|_2^2\) within \(\epsilon^2\|p\|_2^2/5\) using the algorithm from Lemma 2.12 and accepts only if the estimate is below \((1 + 3\epsilon^2/5)/n\).

First, observe the following relationship between the \(\ell_2\) distance to the uniform distribution and the collision probability:

\[
\|p - U_n\|_2^2 = \sum_i (p_i - \frac{1}{n})^2 = \sum p_i^2 - \frac{2}{n} \sum p_i + \frac{1}{n} = \|p\|_2^2 - \frac{1}{n} \tag{2}
\]

If \(\|p - U_n\|_1 \leq \epsilon/\sqrt{3n}\), then \(\|p - U_n\|_2^2 \leq \epsilon^2/3n\). Using (2), we see that \(\|p\|_2^2 \leq (1 + \epsilon^2/3)/n\). Hence, for \(\epsilon \leq 1\), the estimate will be below \((1 + \epsilon^2/5)(1 + \epsilon^2/3)/n\) with probability at least \(1 - \delta\).

Conversely, suppose the estimate of \(\|p\|_2^2\) is below \((1 + 3\epsilon^2/5)/n\). By Lemma 2.12, \(\|p\|_2^2 \leq (1 + 3\epsilon^2/5)/(1 - \epsilon^2/5)n) \leq (1 + \epsilon^2)/n\) for \(\epsilon \leq 1\). Therefore, by (2), we can write

\[
\|p - U_n\|_2^2 = \|p\|_2^2 - \frac{1}{n} \leq \epsilon^2/n.
\]

So, we have \(\|p - U_n\|_2 \leq \epsilon/\sqrt{n}\). Finally, by the relation between \(\ell_1\) and \(\ell_2\) norms, \(\|p - U_n\|_1 \leq \epsilon\).

The sample complexity of the procedure will be \(O(\epsilon^{-4} \cdot \sqrt{n} \cdot \log(1/\delta))\), arising from the estimation of \(\|p\|_2^2\) within \(\epsilon^2\|p\|_2^2/5\). \(\square\)

3. LOWER BOUNDING THE SAMPLE COMPLEXITY

In this section we consider lower bounds on the sample complexity of testing closeness of distributions. In a previous version of this paper [Batu et al. 2000], we claimed an almost matching \(\Omega(n^{2/3})\) lower bound on the sample complexity for testing the closeness of two arbitrary distributions. Although it was later determined that there were gaps in the proofs, recent results of [Valiant 2008] have shown that in fact the almost matching lower bounds do hold. Although new proof techniques were needed, certain technical ideas such as “Poissonization” and the characterization of “canonical forms of testing algorithms” that first appeared in the earlier version of this work did in fact turn out to be useful in the correct lower bound proof of [Valiant 2008]. We will outline those ideas in this section.

We begin by discussing a characterization of canonical algorithms for testing properties of distributions. Then we describe a pair of families of distributions that were suggested in the earlier version of this work, and were in fact used by Valiant [2008] in showing the correct lower bound. Next, we investigate the required dependence on \(\epsilon\). We also note that, more recently, the dependency of testing...
uniformity on distance parameter \( \epsilon \) and \( n \) has been tightly characterized to be \( \Theta(\sqrt{n}/\epsilon^2) \) by Paninski [2008].

### 3.1 Characterization of Canonical Algorithms for Testing Properties of Distributions

In this section, we characterize canonical algorithms for testing properties of distributions defined by permutation-invariant functions. The argument hinges on the irrelevance of the labels of the domain elements for such a function. We obtain this canonical form in two steps, corresponding to the two lemmas below. The first step makes explicit the intuition that such an algorithm should be symmetric, that is, the algorithm cannot benefit from discriminating among the labels. In the second step, we remove the use of labels altogether, and show that we can present the sample to the algorithm in an aggregate fashion. Raskhodnikova et al. [2009] use this characterization of canonical algorithms for proving lower bounds on the sample complexity of distribution support size and element distinctness problems. Valiant [2008] also uses this characterization in showing the lower bound.

Characterizations of property testing algorithms have been studied in other settings. For example, using similar techniques, Alon et al. [1999] show a canonical form for algorithms for testing graph properties. Later, Goldreich and Trevisan [2003] formally prove the result by Alon et al. In a different setting, Bar-Yossef et al. [2001] show a canonical form for sampling algorithms that approximate symmetric functions of the form \( f : A^n \rightarrow B \) where \( A \) and \( B \) are arbitrary sets. In the latter setting, the algorithm is given oracle access to the input vector and takes samples from the coordinate values of this vector.

Next, we give the definitions of basic concepts on which we build a characterization of canonical algorithms for testing properties of distributions. Then, we describe and prove our characterization.

**Definition 3.1 Permutation of a distribution.** For a distribution \( p \) over \([n]\) and a permutation \( \pi \) on \([n]\), define \( \pi(p) \) to be the distribution such that for all \( i \), 
\[
\pi(p)_{\pi(i)} = p_i.
\]

**Definition 3.2 Symmetric algorithm.** Let \( A \) be an algorithm that takes samples from \( k \) discrete black-box distributions over \([n]\) as input. We say that \( A \) is symmetric if, once the distributions are fixed, the output distribution of \( A \) is identical for any permutation of the distributions.

**Definition 3.3 Permutation-invariant function.** A \( k \)-ary function \( f \) on distributions over \([n]\) is permutation-invariant if for any permutation \( \pi \) on \([n]\), and all distributions \((p^{(1)}, \ldots, p^{(k)})\),
\[
f((p^{(1)}, \ldots, p^{(k)}) = f(\pi(p^{(1)}), \ldots, \pi(p^{(k)})).
\]

**Definition 3.4 \( k \)-ary property.** A \( k \)-ary property \( P \) of distributions over \([n]\) is a set of \( k \)-tuples of distributions on \([n]\).

**Lemma 3.5.** Let \( A \) be an arbitrary testing algorithm for a \( k \)-ary property \( P \) defined by a permutation-invariant function. Suppose \( A \) has sample complexity \( s(n) \), where \( n \) is the domain size of the distributions. Then, there exists a symmetric algorithm that tests the same property of distributions with sample complexity \( s(n) \).
Proof. Given the algorithm \( A \), construct a symmetric algorithm \( A' \) as follows: Choose a random permutation of the domain elements. Upon taking \( s(n) \) samples, apply this permutation to each sample. Pass this (renamed) sample set to \( A \) and output according to \( A \).

It is clear that the sample complexity of the algorithm does not change. We need to show that the new algorithm also maintains the testing features of \( A \). Suppose that the input distributions \((p^{(1)}, \ldots, p^{(k)})\) have the property \( \mathcal{P} \). Since the property is defined by a permutation-invariant function, any permutation of the distributions maintains this property. Therefore, the permutation of the distributions should be accepted as well. Let \( S_n \) denote the set of all permutations on \([n]\). Then,

\[
\Pr \left[ A' \text{ accepts } (p^{(1)}, \ldots, p^{(k)}) \right] = \sum_{\pi \in S_n} \frac{1}{n!} \Pr \left[ A \text{ accepts } (\pi(p^{(1)}), \ldots, \pi(p^{(k)})) \right],
\]

which is at least \( 2/3 \) by the accepting probability of \( A \).

An analogous argument on the failure probability for the case of the distributions \((p^{(1)}, \ldots, p^{(k)})\) that should be rejected completes the proof. \( \square \)

In order to avoid introducing additional randomness in \( A' \), we can try \( A \) on all possible permutations and output the majority vote. This change would not affect the sample complexity, and it can be shown that it maintains correctness.

**Definition 3.6 Fingerprint of a sample.** Let \( S_1 \) and \( S_2 \) be multisets of at most \( s \) samples taken from two black-box distributions over \([n]\), \( p \) and \( q \), respectively. Let the random variable \( C_{ij} \), for \( 0 \leq i, j \leq s \), denote the number of elements that appear exactly \( i \) times in \( S_1 \) and exactly \( j \) times in \( S_2 \). The collection of values that the random variables \( \{C_{ij}\}_{0 \leq i, j \leq s} \) take is called the fingerprint of the sample.

For example, let sample sets be \( S_1 = \{5, 7, 3, 3, 4\} \) and \( S_2 = \{2, 4, 3, 2, 6\} \). Then, \( C_{10} = 2 \) (elements 5 and 7), \( C_{01} = 1 \) (element 6), \( C_{11} = 1 \) (element 4), \( C_{02} = 1 \) (element 2), \( C_{21} = 1 \) (element 3), and for remaining \( i, j \)'s, \( C_{ij} = 0 \).

**Lemma 3.7.** If there exists a symmetric algorithm \( A \) for testing a binary property of distributions defined by a permutation-invariant function, then there exist an algorithm for the same task that gets as input only the fingerprint of the sample that \( A \) takes.

**Proof.** Fix a canonical order for \( C_{ij} \)'s in the fingerprint of a sample. Let us define the following transformation on the sample: Relabel the elements such that the elements that appear exactly the same number of times from each distribution (i.e., the ones that contribute to a single \( C_{ij} \) in the fingerprint) have consecutive labels and the labels are grouped to conform to the canonical order of \( C_{ij} \)'s. Let us call this transformed sample the standard form of the sample. Since the algorithm \( A \) is symmetric and the property is defined by a permutation-invariant function, such a transformation does not affect the output of \( A \). So, we can further assume that we always present the sample to the algorithm in the standard form.

It is clear that given a sample in the standard form, we can easily write down the fingerprint of the sample. Moreover, given the fingerprint of a sample, we can always construct a sample \((S_1, S_2)\) in the standard form using the following algorithm: (1) Initialize \( S_1 \) and \( S_2 \) to be empty and \( e = 1 \); (2) For every \( i \) and
j in the canonical order and for the next $C_{ij}$ consecutive values of $e$, include $i$ copies of $e$ in $S_1$, include $j$ copies of $e$ in $S_2$, and increment $e$ by $C_{ij}$. That fact that we can map samples in the standard form to fingerprints and the existence of this algorithm together show that a one-to-one and onto correspondence between sample sets in the standard form and all possible $\{C_{ij}\}_{0 \leq i,j \leq s}$ values.

Consider the algorithm $A'$ that takes the fingerprint of a sample as input. Next, by using algorithm from above, algorithm $A'$ constructs the sample in the standard form. Finally, $A'$ outputs what $A$ outputs on this sample.

**Remark 3.8.** Note that the definition of the fingerprint from Definition 3.6 can be generalized for a collection of $k$ sample sets from $k$ distributions for any $k$. An analogous lemma to Lemma 3.7 can be proven for testing algorithms for $k$-ary properties of distributions defined by a permutation-invariant function. We fixed $k = 2$ for ease of notation.

### 3.2 Towards a Lower Bound on the Sample Complexity of Testing Closeness

In this section, we present techniques that were later used by Valiant [2008] to prove a lower bound on the sample complexity of testing closeness in $\ell_1$ distance as a function of the size $n$ of the domain of the distributions. We give a high-level description of the proof, indicate where our reasoning breaks down and where Valiant [2008] comes in.

**Theorem 3.9 [Valiant 2008].** Given any algorithm using only $o\left(\frac{n^{2/3}}{\log n}\right)$ samples from two discrete black-box distributions over $[n]$, for all sufficiently large $n$, there exist distributions $p$ and $q$ with $\ell_1$ distance 1 such that the algorithm will be unable to distinguish the case where one distribution is $p$ and the other is $q$ from the case where both distributions are $p$.

By Lemma 3.5, we may restrict our attention to symmetric algorithms. Fix a testing algorithm $A$ that uses $o\left(\frac{n^{2/3}}{\log n}\right)$ samples from each of the input distributions.

Let us assume, without loss of generality, that $n$ is a multiple of four and $n^{2/3}$ is an integer. We define the distributions $p$ and $q$ as follows: (1) For $1 \leq i \leq \frac{n^{2/3}}{2}$, $p_i = q_i = \frac{1}{2\sqrt{n}}$. We call these elements the heavy elements. (2) For $\frac{n}{2} < i \leq \frac{3n}{4}$, $p_i = \frac{2}{n}$ and $q_i = 0$. We call these element the light elements of $p$. (3) For $\frac{3n}{4} < i \leq n$, $q_i = \frac{2}{n}$ and $p_i = 0$. We call these elements the light elements of $q$. (4) For the remaining $i$’s, $p_i = q_i = 0$. Note that these distributions do not depend on $A$.

The $\ell_1$ distance of $p$ and $q$ is 1. Now, consider the following two cases:

**Case 1:** The algorithm is given access to two black-box distributions: both of which output samples according to the distribution $p$.

**Case 2:** The algorithm is given access to two black-box distributions: the first one outputs samples according to the distribution $p$ and the second one outputs samples according to the distribution $q$.

To get a sense of why distinguishing between the two cases should be hard for any distance testing algorithm, note that when restricted to the heavy elements, both distributions are identical. The only difference between $p$ and $q$ comes from the light elements, and the crux of the proof is to show that this difference will
not change the relevant statistics in a statistically significant way. For example, consider the statistic which counts the number of elements that occur exactly once from each distribution. One would like to show that this statistic has a very similar distribution when generated by Case 1 and Case 2, because the expected number of such elements that are light is much less than the standard deviation of the number of such elements that are heavy.

Our initial attempts at formalizing the intuition above were incomplete. However, completely formalizing this intuition, Valiant [2008] subsequently showed that a symmetric algorithm with sample complexity $o(n^{2/3})$ can not distinguish between these two cases. By Lemma 3.5, the theorem follows.

Poissonization. For simplifying the proof, it would be useful to have the frequency of each element be independent of the frequencies of the other elements. To achieve this, we assume that algorithm $\mathcal{A}$ first chooses two integers $s_1$ and $s_2$ independently from a Poisson distribution with the parameter $\lambda = s = o(n^{2/3})$. The Poisson distribution with the positive parameter $\lambda$ has the probability mass function $p(k) = \exp(-\lambda)\lambda^k/k!$. Then, after taking $s_1$ samples from the first distribution and $s_2$ samples from the second distribution, $\mathcal{A}$ decides whether to accept or reject the distributions. In the following, we give an overview of the proof that $\mathcal{A}$ cannot distinguish between Case 1 and Case 2 with success probability at least $2/3$. Since both $s_1$ and $s_2$ will have values larger than $s/2$ with probability at least $1 - o(1)$ and the statistical distance of the distributions of two random variables (i.e., the distributions on the samples) is bounded, it will follow that no symmetric algorithm with sample complexity $s/2$ can.

Let $F_i$ be the random variable corresponding to the number of times the element $i$ appears in the sample from the first distribution. Define $G_i$ analogously for the second distribution. It is well known that $F_i$ is distributed identically to the Poisson distribution with parameter $\lambda = sr$, where $r$ is the probability of element $i$; furthermore, it can also be shown that all $F_i$’s are mutually independent (c.f., Feller [1968], p. 216). Thus, the total number of samples from the heavy elements and the total number of samples from the light elements are independent.

Canonical Testing Algorithms. Recall the definition of the fingerprint of a sample from Section 3.1. The random variable $C_{ij}$ denotes the number of elements that appear exactly $i$ times in the sample from the first distribution and exactly $j$ times in the sample from the second distribution. We can then assume that the algorithm is only given the fingerprint of the sample, and apply Lemma 3.7.

Arguing in this way can lead to several subtle pitfalls, which Valiant’s proof [2008] circumvents by developing a body of additional, very nontrivial, technical machinery to show that the distributions on the fingerprint when the samples come from Case 1 or Case 2 are indistinguishable.

A Lower Bound in Terms of $\epsilon$. We can get a lower bound of $\Omega(\epsilon^{-2})$ for testing the $\ell_2$ distance with a rather simple proof.

**Theorem 3.10.** Given any test using only $o(\epsilon^{-2})$ samples, there exist distributions $a$ and $b$ of $\ell_2$ distance $\epsilon$ such that the test will be unable to distinguish the case where one distribution is $a$ and the other is $b$ from the case where both distributions are $a$.
Proof. Let $n = 2$, $a_1 = a_2 = 1/2$ and $b_1 = 1/2 - \epsilon/\sqrt{2}$ and $b_2 = 1/2 + \epsilon/\sqrt{2}$. Distinguishing these distributions is exactly the question of distinguishing a fair coin from a coin of bias $\Theta(\epsilon)$ which is well known to require $\Theta(\epsilon^{-2})$ coin flips. \qed

4. APPLICATIONS TO MARKOV CHAINS

Random walks on Markov chains generate probability distributions over the states of the chain, induced by the endpoints of the random walks. We employ $\ell_1$-Distance-Test, described in Section 2, to test mixing properties of Markov Chains.

This application of $\ell_1$-Distance-Test is initially inspired by the work of Goldreich and Ron [2000], which presented an algorithm for bounded-degree graphs and conjectured that the algorithm distinguishes expander graphs from graphs that are far from expanders (in terms of the fraction of edges that needs to be changed). Their algorithm is based on comparing the distribution of the endpoints of random walks on a graph to the uniform distribution via collisions. Subsequently to this work, Czumaj and Sohler [2007], Kale and Seshadhri [2008], and Nachmias and Shapira [2007] have concluded that the algorithm of Goldreich and Ron is provably a test for expansion property of graphs.

4.1 Preliminaries and Notation

Let $M$ be a Markov chain represented by the transition probability matrix $M$. The point distribution on the $u$th state of $M$ corresponds to an $n$-vector $e_u = (0, \ldots, 1, \ldots, 0)$, with a one in only the $u$th location and zeroes elsewhere. The distribution generated by $t$-step random walks starting at state $u$ is denoted as a vector-matrix product $e_u M^t$.

Instead of computing such products in our algorithms, we assume that our $\ell_1$-Distance-Test has access to an oracle, next_node which on input of the state $u$ responds with the state $v$ with probability $M(u,v)$. Given such an oracle, the distribution $e_u M^t$ can be generated in $O(t)$ steps. Furthermore, the oracle itself can be realized in $O(\log n)$ time per query, given linear preprocessing time to compute the cumulative sums $M_c(j,k) = \sum_{i=1}^k M(j,i)$. The oracle can be simulated on input $u$ by producing a random number $\alpha$ in $[0,1]$ and performing binary search over the $u$th row of $M_c$ to find $v$ such that $M_c(u,v) \leq \alpha \leq M_c(u,v+1)$. It then outputs state $v$. Note that when $M$ is such that every row has at most $d$ nonzero terms, slight modifications of this yield an $O(\log d)$ implementation consuming $O(n + m)$ words of memory if $M$ is $n \times n$ and has $m$ nonzero entries. For the rest of this section, we will denote the running time of the next_node oracle by $T_{nn}(n,d)$, where $n$ is the number of states and $d$ is an upper bound on the number of nonzero entries in any row of $M$.

We define a notion of closeness between states $u$ and $v$, based on the distributions of endpoints of $t$ step random walks starting at $u$ and $v$ respectively.

Definition 4.1. We say that two states $u$ and $v$ are $(\epsilon, t)$-close if the distribution generated by $t$-step random walks starting at $u$ and $v$ are within $\epsilon$ in the $\ell_1$ norm, i.e. $\|e_u M^t - e_v M^t\|_1 \leq \epsilon$. Similarly we say that a state $u$ and a distribution $s$ are $(\epsilon, t)$-close if $\|e_u M^t - s\|_1 \leq \epsilon$.

We say $M$ is $(\epsilon, t)$-mixing if all states are $(\epsilon, t)$-close to the same distribution.
Mixing($M, t, \epsilon, \delta$)

(1) For each state $u$ in $M$
   Reject if $\ell_1$-Distance-Test$(e_u M^t, s_{M,t}, \epsilon, \delta/n)$ rejects.
(2) Otherwise, accept.

Fig. 4. Algorithm Mixing

Definition 4.2. A Markov chain $M$ is $(\epsilon, t)$-mixing if a distribution $s$ exists such that for all states $u$, $\|e_u M^t - s\|_1 \leq \epsilon$.

For example, if $M$ is $(\epsilon, O(\log n \log 1/\epsilon))$-mixing, then $M$ is rapidly-mixing [Sinclair and Jerrum 1989]. It can be easily seen that if $M$ is $(\epsilon, t_0)$-mixing then it is $(\epsilon, t)$-mixing for all $t > t_0$.

We now make the following definition:

Definition 4.3. The average $t$-step distribution, $s_{M,t}$ of a Markov chain $M$ with $n$ states is the distribution

$$s_{M,t} = \frac{1}{n} \sum_u e_u M^t.$$

This distribution can be easily generated by picking $u$ uniformly from $[n]$ and walking $t$ steps from state $u$. In an $(\epsilon, t)$-mixing Markov chain, the average $t$-step distribution is $\epsilon$-close to the stationary distribution. In a Markov chain that is not $(\epsilon, t)$-mixing, this is not necessarily the case.

Each test given below assumes access to an $\ell_1$ distance tester $\ell_1$-Distance-Test$(u, v, \epsilon, \delta)$ which given oracle access to distributions $e_u, e_v$ over the same $n$ element set decides whether $\|e_u - e_v\|_1 \leq f(\epsilon, n)$ or if $\|e_u - e_v\|_1 > \epsilon$ with confidence $1 - \delta$. The time complexity of $\ell_1$-Distance-Test is $T(n, \epsilon, \delta)$, and $f$ is the gap of the tester. The implementation of $\ell_1$-Distance-Test given earlier in Section 2 has gap $f(\epsilon, n) = \epsilon/(4\sqrt{n})$, and time complexity $T = O(\epsilon^{-8/3}n^{2/3} \log \frac{n}{\delta})$.

4.2 A Test for Mixing and a Test for Almost-Mixing

We show how to decide if a Markov chain is $(\epsilon, t)$-mixing; then, we define and solve a natural relaxation of that problem.

In order to test whether $M$ is $(\epsilon, t)$-mixing, one can use $\ell_1$-Distance-Test to compare each distribution $e_u M^t$ with $s_{M,t}$, with error parameter $\epsilon$ and confidence $\delta/n$. The running time is $O(n t \cdot T(n, \epsilon, \delta/n) \cdot T^m(n, n))$. The algorithm is given in Figure 4.

The behavior of the test is as follows: If every state is $(f(\epsilon, n)/2, t)$-close to some distribution $s$, then $s_{M,t}$ is $f(\epsilon, n)/2$-close to $s$. Therefore every state is $(\epsilon, t)$-close to $s_{M,t}$ and the tester passes. On the other hand, if there is no distribution that is $(\epsilon, t)$-close to all states, then, in particular, $s_{M,t}$ is not $(\epsilon, t)$-close to at least one state and so the tester fails. Thus, we have shown the following theorem.

**Theorem 4.4.** Let $M$ be a Markov chain. Given $\ell_1$-Distance-Test with time complexity $T(n, \epsilon, \delta)$ and gap $f$ and an oracle for next node, there exists a test with time complexity $O(\epsilon n \cdot T(n, \epsilon, \delta/n) \cdot T^m(n, n))$ with the following behavior: If $M$ is $(f(\epsilon, n)/2, t)$-mixing, then $\Pr[M \text{ is accepted}] \geq 1 - \delta$; if $M$ is not $(\epsilon, t)$-mixing, then $\Pr[M \text{ is accepted}] \leq \delta$. 

AlmostMixing(M, t, ε, δ, ρ)
Repeat O(1/ρ · ln(1/δ)) times
(1) Pick a state u in M uniformly at random.
(2) Reject if 1-Distance-Test(eu, M, sM, t, ε, δ, ρ) rejects.
Accept if none of the tests above rejected.

For the implementation of 1-Distance-Test given in Section 2, the running time of Mixing algorithm is \( O(\varepsilon^{-8/3} n^{5/3} t \log \frac{n}{\varepsilon}) \). It distinguishes between chains which are \( \varepsilon/2 \)-mixing and those which are not \( \varepsilon \)-mixing. The running time is sublinear in the size of M if \( t < o(n^{1/3}/\log(n)) \).

A relaxation of this procedure is testing that most starting states reach the same distribution after \( t \) steps. If \( (1 - \rho) \) fraction of the states \( u \) of a given M satisfy \( \|\vec{s} - e_u M\|_2 \leq \varepsilon \), then we say that M is \( (\rho, \varepsilon, t) \)-almost mixing. The algorithm in Figure 5 tests whether a Markov chain is \( (\rho, \varepsilon, t) \)-almost mixing.

Thus, we obtain the following theorem.

**Theorem 4.5**. Let M be a Markov chain. Given 1-Distance-Test with time complexity \( T(n, \varepsilon, \delta) \) and gap f and an oracle for next node, there exists a test with time complexity \( O(\varepsilon^{-1} \log \frac{n}{\varepsilon^2} T(n, \varepsilon, \delta) \cdot T^{\text{max}}(n, n)) \) with the following behavior: If M is \( (f(n)/2, t) \)-mixing, then \( \Pr[M \text{ is accepted}] > 1 - \delta \); if M is not \( (\rho, \varepsilon, t) \)-almost mixing, then \( \Pr[M \text{ is accepted}] < \delta \).

**4.3 A Property Tester for Mixing**

The main result of this section is a test that determines whether a Markov chain’s matrix representation can be changed in an \( \varepsilon \) fraction of the non-zero entries to turn it into a \((7\varepsilon, 2t)\)-mixing Markov chain. This notion falls within the scope of property testing [Rubinfeld and Sudan 1996; Goldreich et al. 1998; Ron 2001; Fischer 2001], which in general takes a set \( S \) with distance function \( \Delta \) and a subset \( P \subseteq S \) and decides if an elements \( x \in S \) is in \( P \) or if it is far from every element in \( P \), according to \( \Delta \). For the Markov chain problem, we take as our set \( S \) all matrices M of size \( n \times n \) with at most \( d \) non-zero entries in each row. Hence, the outdegree of each state is upper bounded by \( d \), whereas the indegree is not constrained. The distance function is given by the fraction of non-zero entries in which two matrices differ, and the difference in their average \( t \)-step distributions.

**4.3.1 Preliminaries**

We start with defining a distance function on a pair of Markov chains on the same state space.

**Definition 4.6.** Let M1 and M2 be n-state Markov chains with at most \( d \) non-zero entries in each row. For positive integer \( t \), define distance function \( \Delta_t(M_1, M_2) = (\epsilon_1, \epsilon_2) \) if and only if M1 and M2 differ on \( \epsilon_1 \)dn entries and \( \|s_{M_1,t} - s_{M_2,t}\|_1 = \epsilon_2 \). We say that \( M_1 \) and \( M_2 \) are \((\epsilon_1, \epsilon_2, t)\)-close if \( \Delta_t(M_1, M_2) \leq (\epsilon_1, \epsilon_2) \).

A natural question is whether all Markov chains are \((\epsilon_1, \epsilon_2, t)\)-close to an \((\epsilon, t)\)-mixing Markov chain, for certain values of the parameters. For example, given a

\[^2\text{We say } (x, y) \leq (a, b) \text{ if } x \leq a \text{ and } y \leq b.\]
strongly connected and dense enough Markov chain, adding the edges of a constant-degree expander graph and choosing \( t = \Theta(\log n) \) yields a Markov chain which \((\epsilon, t)\)-mixes. However, for sparse Markov chains or small \( \epsilon \), such a transformation does not work. Furthermore, the situation changes when asking whether there is an \((\epsilon, t)\)-mixing Markov chain that is close both in the matrix representation and in the average \( t \)-step distribution: specifically, it can be shown that there exist constants \( \epsilon, \epsilon_1, \epsilon_2 < 1 \) and Markov chain \( M \) for which no Markov chain is both \((\epsilon_1, \epsilon_2, \log n)\)-close to \( M \) and \((\epsilon, \log n)\)-mixing. In fact, when \( \epsilon_1 \) is small enough, the problem becomes nontrivial even for \( \epsilon_2 = 1 \). The Markov chain corresponding to random walks on the \( n \)-cycle provides an example which is not \((\alpha \sqrt{n}/2, 1, n)\)-close to any \((\epsilon, n)\)-mixing Markov chain for some small \( \alpha > 0 \), because, even after the addition of \( \alpha \sqrt{n} \) edges to the \( n \)-cycle, there will be nodes on the cycle that are distance \( \Omega(\sqrt{n}) \) apart for which no edge was added on the path between them. Then, a random walk from the midpoint of the path between such a pair would require \( \Omega(n) \) steps to converge to the stationary distribution.

**Overview.** As before, our algorithm proceeds by taking random walks on the Markov chain and comparing final distributions by using the \( \ell_1 \)-Distance-Test. We define three types of states. First, a **normal** state is one from which a random walk of \( t \) steps arrives at nearly the average \( t \)-step distribution. In the discussion which follows, \( t \) and \( \epsilon \) denote constant parameters fixed as input to the algorithm.

**Definition 4.7.** Given a Markov Chain \( M \), a state \( u \) of the chain is **normal** if it is \((\epsilon, t)\)-close to \( s_{M,t} \); that is, if \( \| e_u M^t - s_{M,t} \|_1 \leq \epsilon \). A state is **bad** if it is not normal.

Using this definition, the first two algorithms given in this section can be described as testing whether all (resp., most) states in \( M \) are normal. Additionally, we need to distinguish states which not only produce random walks which arrive near \( s_{M,t} \) but which have low probability of visiting a bad state. We call such states **smooth** states.

**Definition 4.8.** A state \( u \) in a Markov chain \( M \) is **smooth** if (a) \( u \) is \((\epsilon, \tau)\)-close to \( s_{M,t} \) for \( \tau = t, \ldots, 2t \) and (b) the probability that a \( 2t \)-step random walk starting at \( u \) visits a bad state is at most \( \epsilon \).

Our property test merely verifies by random sampling that most states are smooth.

**4.3.2 The Test.** We present algorithm **TestMixing** in Figure 6, which on input Markov chain \( M \) and parameter \( \epsilon \) determines whether at least \((1 - \epsilon)\) fraction of the states of \( M \) are smooth according to two distributions: the uniform distribution and the average \( t \)-step distribution. Assuming access to \( \ell_1 \)-Distance-Test with complexity \( T(n, \epsilon, \delta) \), this test runs in time \( O(\epsilon^{-2} t^2 \cdot T(n, \epsilon, 1/n) \cdot T_{run}(n, d)) \).

The main lemma of this section says that any Markov chain that is accepted by our test with at least some constant probability is \((2\epsilon, 4t, t)\)-close to a \((7\epsilon, 2t)\)-mixing Markov chain. First, we describe the modification of \( M \) that we later show is \((7\epsilon, 2t)\)-mixing.
TestMixing($M, t, \epsilon$)

(1) Let $k = \Theta(1/\epsilon)$.

(2) Choose $k$ states $u_1, \ldots, u_k$ uniformly at random.

(3) Choose $k$ states $u_{k+1}, \ldots, u_{2k}$ independently according to $s_{M,t}$.

(4) For $i = 1$ to $2k$
   (a) $u = \tilde{e}_u$.
   (b) For $w = 1$ to $O(1/\epsilon)$ and $j = 1$ to $2t$
      i. $u = \text{next_node}(M, u)$
      ii. $\tilde{\ell}_1\text{-Distance-Test}(e_u M^t, s_{M,t}, \epsilon, \frac{1}{\epsilon^2})$
   (c) For $\tau = t$ to $2t$, $\ell_1\text{-Distance-Test}(\tilde{e}_u M^t, s_{M,t}, \epsilon, \frac{1}{\epsilon^2})$

(5) Pass if all tests pass.

**Definition 4.9.** $F_u$ is a function from $n \times n$ matrices to $n \times n$ matrices such that $F_u(M)$ returns $M$ by modifying the rows corresponding to bad states of $M$ to $\tilde{e}_u$, where $u$ is any smooth state.

An important feature of the transformation $F_u$ is that it does not affect the distribution of random walks originating from smooth states very much.

**Lemma 4.10.** Given a Markov chain $M$ and states $u, u' \in M$ which are smooth, if $\tilde{M} = F_u(M)$, then, for any time $t \leq \tau \leq 2t$, $\|e_u M^\tau - e_u \tilde{M}^\tau\|_1 \leq 2\epsilon$ and $\|s_{M,t} - e_u \tilde{M}^\tau\|_1 \leq 3\epsilon$. 

**Proof.** Define $\Gamma$ as the set of all walks of length $\tau$ from $u$ in $M$. Partition $\Gamma$ into $\Gamma_B$ and $\hat{\Gamma}_B$, where $\Gamma_B$ is the subset of walks that visit a bad state. Let $\chi_{w,i}$ be an indicator function which equals $1$ if walk $w$ ends at state $i$, and $0$ otherwise. Let weight function $W(w)$ be defined as the probability that walk $w$ occurs. Finally, define the primed counterparts $\Gamma', W'$, etc. for the Markov chain $\tilde{M}$. Now the $ith$ coordinate of $e_u M^\tau = \sum_{w \in \Gamma_B} \chi_{w,i} W(w) + \sum_{w \in \hat{\Gamma}_B} \chi_{w,i} W(w)$. A similar expression can be written for each element of $e_u M^\tau$. Since $W(w) = W'(w)$ whenever $w \in \Gamma_B$, it follows that

$$\|e_u M^\tau - e_u \tilde{M}^\tau\|_1 \leq \sum_i \sum_{w \in \Gamma_B} \chi_{w,i} |W(w) - W'(w)| \leq 2 \sum_i \sum_{w \in \Gamma_B} \chi_{w,i} W(w) \leq 2\epsilon.$$

Additionally, since $\|s_{M,t} - e_u M^\tau\|_1 \leq \epsilon$ by the definition of smooth, it follows that $\|s_{M,t} - e_u \tilde{M}^\tau\|_1 \leq 2\epsilon$ and $\|e_u M^\tau - e_u \tilde{M}^\tau\|_1 \leq 3\epsilon$. 

We can now prove the main lemma.

**Lemma 4.11.** If, according to both the uniform distribution and the distribution $s_{M,t}$, the total weight of the states of a Markov chain $M$ that are smooth is at least $1 - \epsilon$, then the matrix $M$ is $(2\epsilon, 4\epsilon, t)$-close to a matrix $\tilde{M}$ which is $(7\epsilon, 2t)$-mixing.

**Proof.** Let $\tilde{M} = F_u(M)$ for some smooth state $u \in M$. $\tilde{M}$ and $M$ differ on at most $\epsilon n(d+1) \leq 2\epsilon nd$ entries. This gives a bound on the first component of distance $\Delta_1(M, \tilde{M})$. For the second component, we analyze

$$\|s_{M,t} - s_{\tilde{M},t}\|_1 = \frac{1}{n} \sum_u \|e_u M^t - e_u \tilde{M}^t\|_1.$$
as follows. The sum is split into two parts, over the states that are smooth and those states that are not. For each smooth state \( u \), Lemma 4.10 says that \( \|e_u M^t - e_u \tilde{M}\|_1 \leq 2\epsilon \). The states that are not smooth account for at most \( \epsilon \) fraction of the states in the sum, and thus can contribute no more than \( \epsilon \) absolute weight to the distribution \( s_{M,t} \). Hence, the sum can be bounded by

\[
\|s_{M,t} - s_{M,t}\|_1 \leq \frac{1}{n} (2(1 - \epsilon)en + 2\epsilon n) \leq 4\epsilon.
\]

In order to show that \( \tilde{M} \) is \((7\epsilon, 2t)\)-mixing, we prove that, for every state \( u \),

\[
\|s_{M,t} - e_u \tilde{M}^{2t}\|_1 \leq 7\epsilon.
\]

The proof considers three cases: \( u \) is smooth, \( u \) is bad, and \( u \) is normal. The last case is the most involved.

If \( u \) is smooth in Markov chain \( M \), then, by Lemma 4.10, we obtain that

\[
\|s_{M,t} - e_u \tilde{M}^{2t}\|_1 \leq 3\epsilon.
\]

Similarly, if \( u \) is bad in Markov chain \( M \), then, in Markov chain \( \tilde{M} \), any path starting at \( u \) transitions to a smooth state \( v \) in one step. Since

\[
\|s_{M,t} - e_v \tilde{M}^{2t-1}\|_1 \leq 3\epsilon
\]

by Lemma 4.10, the desired bound follows.

If \( u \) is a normal state that is not smooth, then we need a more involved analysis of the distribution \( e_u \tilde{M}^{2t} \). We define \( \Gamma \) to be the set of all \( 2t \)-step walks in \( M \) starting at \( u \) and divide \( \Gamma \) into three sets, which we consider separately.

For the first set, take \( \Gamma_B \subseteq \Gamma \) to be the set of walks that visit a bad state (according to \( M \)) before time \( t \). Let \( d_b \) be the distribution over endpoints of these walks; that is, let \( d_b \) assign to state \( i \) the probability that any walk \( w \in \Gamma_B \) ends at state \( i \). Let \( w \in \Gamma_B \) be any such walk. If \( w \) visits a bad state at time \( \tau < t \), then in the Markov chain \( M \), \( w \) visits a smooth state \( v \) at time \( t + 1 \). Another application of Lemma 4.10 implies that \( \|e_v \tilde{M}^{2t-\tau-1} - s_{M,t}\|_1 \leq 3\epsilon \). Since this is true for all walks \( w \in \Gamma_B \), we find \( \|d_b - s_{M,t}\|_1 \leq 3\epsilon \).

For the second set, let \( \Gamma_S \subseteq \Gamma \setminus \Gamma_B \) be the set of walks not in \( \Gamma_B \) that visit a smooth state (according to \( M \)) at time \( t \). Let \( d_s \) be the distribution over endpoints of these walks. Any walk \( w \in \Gamma_S \) is identical in the chains \( M \) and \( \tilde{M} \) up to time \( t \), and then, in the chain \( \tilde{M} \), it visits a smooth state \( v \) at time \( t \). Since \( \|s_{M,t} - e_v M^t - s_{M,t}\|_1 \leq 3\epsilon \), we have \( \|d_s - s_{M,t}\|_1 \leq 3\epsilon \).

Finally, let \( \Gamma_N = \Gamma \setminus (\Gamma_B \cup \Gamma_S) \) and let \( d_n \) be the distribution over endpoints of walks in \( \Gamma_N \). Then, \( \Gamma_N \) consists of a subset of the walks from a normal state \( u \) (according to \( M \)) that do not visit a smooth state at time \( t \). By the definition of normal, \( u \) is \((\epsilon, t)\)-close to \( s_{M,t} \) in Markov chain \( M \). By the premise of the lemma, at most \( \epsilon \) weight of \( s_{M,t} \) is assigned to states which are not smooth. Therefore, the total weight of the walks in \( \Gamma_N \) is at most \( 2\epsilon \).

Now define the weights of these distributions as \( \omega_b, \omega_s, \) and \( \omega_n \). That is, \( \omega_b \) is the probability that a walk from \( u \) in \( \tilde{M} \) visits a bad state before time \( t \); \( \omega_s \) is the probability that a walk does not visit a bad state before time \( t \), but visits a smooth state at time \( t \); and \( \omega_n \) is the probability that a walk does not visit a bad state but visits a normal, non-smooth state at time \( t \) (which was shown to be at most \( 2\epsilon \)).
above). Then, $\omega_b + \omega_s + \omega_n = 1$. Finally,

$$\|e_nM^{2\epsilon} - s_{M,t}\|_1 = \|\omega_b \cdot d_b + \omega_s \cdot d_s + \omega_n \cdot d_n - s_{M,t}\|_1$$

$$\leq \omega_b \cdot \|d_b - s_{M,t}\|_1 + \omega_s \cdot \|d_s - s_{M,t}\|_1 + \omega_n \cdot \|d_n - s_{M,t}\|_1$$

$$\leq (\omega_b + \omega_s) \cdot \max\{\|d_b - s_{M,t}\|_1, \|d_s - s_{M,t}\|_1\} + \omega_n \cdot \|d_n - s_{M,t}\|_1$$

$$\leq (\omega_b + \omega_s) \cdot 3\epsilon + 2 \cdot \omega_n$$

$$\leq 7\epsilon.$$

This concludes the proof that $M$ is $(7\epsilon, 2t)$-mixing. □

Given this, we finally can show our main theorem.

**Theorem 4.12.** Let $M$ be a Markov chain. Given $\ell_1$-Distance-Test with time complexity $T(n, \epsilon, \delta)$ and gap $\epsilon$ and an oracle for next_node, there exists a test such that if $M$ is $(f(\epsilon, \nu), t)$-mixing then the test accepts with probability at least $2/3$. If $M$ is not $(2\epsilon, 4\epsilon, t)$-close to any $\tilde{M}$ which is $(7\epsilon, 2t)$-mixing then the test rejects with probability at least $2/3$. The runtime of the test is $O(\frac{1}{\epsilon^2} \cdot t^2 \cdot T(n, \epsilon_\nu, \frac{1}{6\epsilon}) \cdot T^{\nu}(n, d))$.

**Proof.** The test is given in Figure 6. Since all the states in a Markov chain $M$ that are $(f(\epsilon, \nu), t)$-mixing are smooth, the test accepts $M$ with probability at least $2/3$. Furthermore, any Markov chain with at least $(1 - \epsilon)$ total weight on the smooth states according to both the uniform distribution and the distribution $s_{M,t}$ is $(2\epsilon, 4\epsilon, t)$-close to a Markov chain which is $(7\epsilon, 2t)$-mixing, by Lemma 4.11. The algorithm, by testing $O(1/\epsilon)$ states chosen randomly from each distribution, verifies that the smooth states have high total weight according to both of these distributions. □

### 4.4 Extension to Sparse Graphs and Uniform Distributions

The property test for mixing can also be made to work for general sparse Markov chains by a simple modification to the testing algorithm. Consider Markov chains with at most $m \ll n^2$ nonzero entries, but with no nontrivial bound on the number of nonzero entries per row. Then, the definition of the distance should be modified to $d_2(M_1, M_2) = (\epsilon_1, \epsilon_2)$ if $M_1$ and $M_2$ differ on $\epsilon_1 \cdot m$ entries and the $\|s_{M_1,t} - s_{M_2,t}\|_1 = \epsilon_2$. The algorithm **TestMixing** in Figure 6 does not suffice for testing that $M$ is $(\epsilon_1, \epsilon_2, t)$-close to an $(\epsilon, t)$-mixing Markov chain $\tilde{M}$, since in our proof, the rows corresponding to bad states may have many nonzero entries and thus $M$ and $\tilde{M}$ may differ in a large fraction of the nonzero entries. Thus, bad states may be responsible for a disproportionate number of the nonzero entries. To fix this, when the algorithm accepts a Markov chain, it should also ensure that, on average, the bad states do not have high outdegrees. Let $D$ be a distribution on states in which the probability of each state is proportional to the cardinality of the support set of its row. Natural ways of encoding this Markov chain allow constant-time generation of states according to $D$. By adding a test that whether most states according to $D$ are smooth, one can show that $\tilde{M}$ is close to an $(\epsilon, t)$-mixing Markov chain $M$.

Because of our ability to test $\epsilon$-closeness to the uniform distribution in $O(n^{1/2} \epsilon^{-4})$ steps presented in Section 2.3 and by Goldreich and Ron [2000], it is possible to speed up our test for mixing for those Markov chains known to have uniform Journal of the ACM, Vol. V, No. N, Month 20YY.
stationary distribution, such as Markov chains corresponding to random walks on regular graphs. An ergodic random walk on the vertices of an undirected graph instead may be regarded (by looking at it “at times $t + 1/2$”) as a random walk on the edge-midpoints of that graph. The stationary distribution on edge-midpoints always exists and is uniform. Hence, for undirected graphs, we can speed up mixing testing by using a tester for closeness to the uniform distribution.

ACKNOWLEDGMENTS

We are very grateful to Oded Goldreich and Dana Ron for sharing an early draft of their work with us and for several helpful discussions. We would also like to thank Naok Abe, Richard Beigel, Yoav Freund, Russell Impagliazzo, Jeff Ketchersid, Kevin Matulef, Alexis Maciel, Krzysztof Onak, Sofya Raskhodnikova, and Tassos Viglas for helpful discussions. Finally, we thank Ning Xie for pointing out errors in the proofs in an earlier version.

REFERENCES


Csiszár, I. 1967. Information-type measures of difference of probability distributions and indirect observations. Studia Scientiarum Mathematicarum Hungarica.


Testing Closeness of Discrete Distributions

Mexico, S. Goldwasser, Ed. IEEE Computer Society Press, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 656–671.


