New Bounds for Circular Trace Reconstruction

Arnav Burudgunte
Department of Computer Science, Purdue University, West Lafayette, IN, USA

Paul Valiant
Department of Computer Science, Purdue University, West Lafayette, IN, USA

Hongao Wang
Department of Computer Science, Purdue University, West Lafayette, IN, USA

Abstract

The "trace reconstruction" problem asks, given an unknown binary string x and a channel that repeatedly returns "traces" of x with each bit randomly deleted with some probability p, how many traces are needed to recover x? There is an exponential gap between the best known upper and lower bounds for this problem. Many variants of the model have been introduced in hopes of motivating or revealing new approaches to narrow this gap. We study the variant of *circular* trace reconstruction introduced by Narayanan and Ren (ITCS 2021), in which traces undergo a random cyclic shift in addition to random deletions.

We show an improved lower bound of $\tilde{\Omega}(n^5)$ for circular trace reconstruction. This contrasts with the (previously) best known lower bounds of $\tilde{\Omega}(n^3)$ in the circular case and $\tilde{\Omega}(n^{3/2})$ in the linear case. Our bound shows the indistinguishability of traces from two *sparse* strings x, y that each have a *constant* number of nonzeros. Can this technique be extended significantly? How hard is it to reconstruct a *sparse* string x under a cyclic deletion channel? We resolve these questions by showing, using Fourier techniques, that $\tilde{O}(n^6)$ traces suffice for reconstructing any constant-sparse string in a circular deletion channel, in contrast to the best known upper bound of $\exp(\tilde{O}(n^{1/3}))$ for general strings in the circular deletion channel. This shows that new algorithms or new lower bounds must focus on *non-constant-sparse* strings.

2012 ACM Subject Classification Mathematics of computing → Probabilistic inference problems

Keywords and phrases Trace reconstruction, algorithmic statistics, Fourier analysis

Digital Object Identifier 10.4230/LIPIcs.ITCS.2026.30

Related Version *Full Version*: `https://arxiv.org/abs/2512.02412` [5]

Funding This work is partially supported by NSF award CCF-2127806 and by Office of Naval Research award N000142412695.

1 Introduction

Given a binary string x and a fixed probability p, the deletion channel $\mathbf{Del}(x)$ deletes each character in x with probability p and returns the remaining string, which is called a *trace* of the deletion channel. The trace reconstruction problem asks: how many traces from $\mathbf{Del}(x)$ are needed to recover x reliably. This problem has been studied extensively in various forms and under various assumptions [3, 2, 6, 7, 15, 20]. In the standard setting, where the deletion probability p is a constant and the original string x is an arbitrary binary string of length n, there remains an exponential gap between the best known upper bound $\exp(\tilde{O}(n^{1/5}))$ and the best known lower bound $\tilde{\Omega}(n^{3/2}))$ for worst-case reconstruction [6, 7]. In hopes of narrowing this gap, many related problems have been proposed and studied, such as matrix trace reconstruction [15], population recovery [2], and coded trace reconstruction [10, 4].

We study one such mild variant, the problem of *circular trace reconstruction* introduced by [17]. In this problem, the original string x undergoes a random cyclic shift after deletion, and the reconstruction algorithm must return any string which is cyclically equivalent to x.

Circular trace reconstruction is known to be at least as hard as the linear version (essentially a consequence of the data processing inequality). Also, the best known bounds in each model are fairly similar: prior to our work, the best known lower bound was $\tilde{\Omega}(n^3)$ in the circular model compared to $\tilde{\Omega}(n^{3/2})$ in the linear model [6, 17]. The best known upper bound for arbitrary strings is $\exp(\tilde{O}(n^{1/3}))$ in the circular model (assuming n can be written as a product of two or fewer primes), compared to $\exp(\tilde{O}(n^{1/5}))$ in the linear model [7, 17].

1.1 Our Contributions

We improve the lower bound for circular trace reconstruction from $\tilde{\Omega}(n^3)$ to $\tilde{\Omega}(n^5)$.

We prove this result by exhibiting two cyclically distinct strings x, y that have a *constant* number of nonzero entries, yet which need $\tilde{\Omega}(n^5)$ traces to distinguish. This strong lower bound motivates the general question: how difficult is it to reconstruct constant-sparse strings x from circular traces? To what degree can this lower bound of $\tilde{\Omega}(n^5)$ be improved?

We resolve these questions almost-tightly by showing that $\tilde{O}(n^6)$ traces suffice to distinguish any two *sparse* circular strings. Our results imply that stronger lower bounds can only be obtained from non-constant-sparse strings, and also that new algorithmic improvements should focus on the non-constant-sparse case.

1.2 Technical Overview

We focus on the specific case of sparse circular strings, defined as binary strings which contain k 1s for some constant k. This family of strings has the useful property that any k-sparse string x is a cyclic shift of $10^{x_1}10^{x_2}\dots10^{x_k}$; therefore, x can be represented as the integer sequence $(x_1, \dots, x_k)$, where x_j specifies the number of 0s (the "gap") between the jth and $(j+1)$th 1s in x. With probability $(1-p)^k$ the deletion channel will preserve all 1s in x, yielding a trace $\tilde{x}$ which can also be represented by a sequence of k gaps. Crucially, in our setting of constant p, k, this probability $(1-p)^k$ is constant. When this occurs, the distribution over traces $\tilde{x}$ depends only on the original integer sequence $x_1, \dots, x_k$ of gaps in x. Both our upper and lower bounds are based on properties of this integer sequence.

Cyclic Statistics. Our main technical tool is a family of shift-invariant functions which we call *cyclic statistics*. At a high level, a cyclic statistic of a sequence $x = (x_1, \dots, x_k)$ is a cyclically-invariant polynomial of the variables $x_1, \dots, x_k$; the statistic is obtained by summing a monomial function over all cyclic shifts of x. The *order* of this statistic is the degree of the corresponding monomial. If two strings x and y are cyclic shifts of each other, their cyclic statistics are clearly identical. Surprisingly, we prove that the converse holds for order 6: if x and y are cyclically distinct, there exists a cyclic statistic of order ≤ 6 in which they differ. We give a lower bound showing that this result is nearly tight.

▶ **Lemma 1** (Informal). *Let x and y be two cyclically distinct integer sequences. Then x and y differ in some cyclic statistic of order at most 6. Conversely, there exist integer sequences x and y with identical cyclic statistics up to order 4.*

We prove the upper bound in Lemma 1 using Fourier techniques (see Lemma 19 in Section 3). We show that if x and y have identical cyclic statistics up to order 6, their Fourier transforms $\hat{x}$ and $\hat{y}$ must satisfy a certain system of sparse linear equations, each involving 6 or fewer variables. We then analyze this system and show that any solution $\hat{x}, \hat{y}$ has the property that there exists an integer c for which

$$\hat{x}_j = \hat{y}_j \exp\left(2\pi i \cdot \frac{cj}{k}\right) \quad \text{for all } j \in [k]. \tag{1}$$

By basic properties of Fourier transforms, this implies that y is a cyclic shift of x.

The significance of the order 6 stems from a number theoretic result. Our original constraints involve an equation of two variables for each $j \in [k]$, but our analysis only applies for j relatively prime to k. We show that each $j \in [k]$ can be written as the sum of 2 or 3 integers relatively prime to k, transforming our constraint into a 6-variable equation, which implies Equation 1, given all cyclic statistics up to order 6 vanish.

The lower bound – that there exist x and y with identical cyclic statistics up to order 4 – is proved directly. We provide a pair of sequences of length 12 satisfying this property, obtained via computer search: $x_{(j)} = (0, 2, 3, 2, 1, 1, 1, 1, 2, 3, 2, 0)$ and the sequence $y_{(j)} = 3 - x_{(j)}$.

Cyclic statistics are preserved in expectation by the cyclic deletion channel, leading to our algorithmic results. They also characterize the distribution of traces obtained from the deletion channel, from which we obtain our information-theoretic lower bound. We give a brief overview of this connection and our results below.

Lower Bound. In order to build our lower bound, we relate the probability distribution over traces of x to the cyclic statistics of $(x_1, \ldots, x_k)$. Roughly, we show that the probability of generating a trace $\tilde{x}$ from x can be expressed as a polynomial in n of degree L whose degree $L - m$ coefficient depends only on the order m cyclic statistics of x. We use this fact to show that if two cyclically distinct strings x and y have identical cyclic statistics up to order m, the Hellinger distance between the distributions of the traces generated from x and y is $\tilde{O}(n^{-m-1})$. Therefore, it requires $\tilde{\Omega}(n^{m+1})$ samples to distinguish the two distributions.

▶ **Theorem 2** (Informal version of Lemma 33). *Let $(x_1, \ldots, x_k)$ and $(y_1, \ldots, y_k)$ be two sequences that are permutations of each other and have identical cyclic statistics up to order m. Then for any constant deletion probability p, any algorithm which distinguishes the strings*

$$x = 10^{n+x_1}10^{n+x_2}\ldots10^{n+x_k} \text{ and } y = 10^{n+y_1}10^{n+y_2}\ldots10^{n+y_k}$$

requires $\tilde{\Omega}(n^{m+1})$ traces from a cyclic deletion channel.

By Lemma 1, there exist two cyclically distinct sequences x and y with identical cyclic statistics up to order 4. Plugging the corresponding strings into Theorem 2, we obtain our $\tilde{\Omega}(n^5)$ lower bound for circular trace reconstruction.

Upper Bound. We build an algorithm that can recover any k-sparse string x using $\tilde{O}(n^6)$ samples from the deletion channel, showing that our techniques from above *cannot* be extended to yield a significantly stronger lower bound.

▶ **Theorem 3** (Informal version of Theorem 26). *Let x and y be two cyclically distinct constant-sparse binary strings of length n. Then there exists an algorithm which, for any constant deletion probability p, distinguishes x from y using $\tilde{O}(n^6)$ traces from a cyclic deletion channel.*

As explained above, our algorithm exploits the fact that the retention probability $q \coloneqq 1-p$ is a constant. Therefore, with constant probability, no 1 is deleted in x. Conditioning on this, every trace $\tilde{x}$ can be expressed as a sequence of binomial random variables; that is, $\tilde{x} = (\tilde{x}_1, \ldots, \tilde{x}_k)$ with $\tilde{x}_j \sim \mathbf{Bin}(x_j, q)$. By Lemma 1, any two cyclically distinct sequences x and y differ in some cyclic statistic of order at most 6. As all cyclic statistics of integer sequences are integers, the difference between x and y in this statistic is at least 1. Thus we aim to estimate cyclic statistics from traces to within constant error.

A naïve estimator is the empirical average of a certain cyclic statistic of the traces. However, this naïve algorithm requires $\Omega(n^{11})$ samples. This is because, in the worst case, each $\tilde{x}_j$ may have mean and variance both $\Omega(n)$. Thus the variance of the product of 6 such random variables may be $\Omega(n^{11})$. However, instead, if we can shift the binomials such that they become random variables whose absolute value is $\tilde{O}(\sqrt{n})$, the absolute value of their product will be bounded by $\tilde{O}(n^3)$ leading to variance $\tilde{O}(n^6)$. We accomplish this by greedily partitioning the set of expected gaps $\{qx_j\}$ into clusters, so that all elements in the same cluster are $\tilde{O}(\sqrt{n})$-close to each other and $\tilde{\Omega}(\sqrt{n})$-far from all other clusters. Since each binomial is tightly concentrated around its expectation, we can map each $\tilde{x}_j$ to the correct cluster with high probability, and subtract the cluster center from $\tilde{x}_j$ to obtain a random variable whose absolute value is bounded by $\tilde{O}(\sqrt{n})$ with high probability.

A final hurdle is that subtracting cluster centers may erase any differences in the cyclic statistics of x and y. See the discussion at the start of Section 4 and Example 22 for more algorithmic intuition, and discussion of why *cyclic statistics mod ℓ* are needed to yield a $\tilde{O}(n^6)$ algorithm. See Algorithm 1 and Theorem 26 for our main algorithmic result.

1.3 Structure of the Rest of the Paper

In Section 1.4 we discuss related work. Section 2 introduces the key definitions used in this paper. The rest of the paper consists of three technical sections that may be read independently, in any order. Section 3 builds tools to analyze cyclic statistics, leading to our characterization that "any two cyclically distinct strings x, y must differ in *some* cyclic statistic of order $m \leq 6$." Section 4 contains our algorithmic results, using the characterization of Section 3 to give an algorithm that distinguishes any constant-sparse strings with $\tilde{O}(n^6)$ traces. Section 5 shows our lower bounds, based on a Hellinger distance analysis, showing that two strings x, y that are permutations of each other and have identical cyclic statistics up to order m require $\tilde{\Omega}(n^{m+1})$ traces to distinguish.

1.4 Related Work

The trace reconstruction problem was introduced in [16, 3]. [3] shows strong results for reconstructing *random* strings x, in contrast to the worst-case setting we consider here. [14] gives an algorithm for the (worst-case) trace reconstruction problem with constant deletion probability using $\exp(\tilde{O}(n^{1/2}))$ traces. This result was improved independently by [11] and [18], which present an algorithm using $\exp(O(n^{1/3}))$ traces. The state of the art is [7], providing an algorithm using $\exp(\tilde{O}(n^{1/5}))$ traces.

On the lower bound side, [3] provides a $\Omega(n)$ bound, showing that distinguishing 0^n10^{n-1} from $0^{n-1}10^n$ requires $\Omega(n)$ traces. [13] improved the lower bound to $\tilde{\Omega}(n^{5/4})$ by replacing the 0's in the previous strings by alternating $01010101\ldots$, and [6] tightened this analysis to yield $\tilde{\Omega}(n^{3/2})$, the best known lower bound, but still exponentially far from $\exp(\tilde{O}(n^{1/5}))$.

Restricting the class of algorithms to yield improved lower bounds, [11] shows that all "mean-based" algorithms require $\exp(\Omega(n^{1/3}))$ samples to reconstruct strings. Here, a mean-based algorithm is any algorithm that only uses the empirical means of individual bits of the traces. Extending this line of work, [9] considers "k-mer-based" algorithms which rely on counting the occurrences of contiguous k-bit strings (where "mean-based" algorithm are the case $k = 1$). [9] shows that, even for k up to $n^{1/5}$, any k-mer-based algorithm requires $\exp(\Omega(n^{1/5}))$ samples to reconstruct the string. All known algorithms for the worst-case trace reconstruction problem are k-mer-based, including the algorithm presented by [7]. More recently, [8] strengthened this result to show an analogous $\exp(\Omega(n^{1/5}))$-sample lower bound for all statistical query (SQ) algorithms whose queries are "$\tilde{O}(n^{1/5})$-local." Thus if there is

an algorithm using fewer than $\exp(n^{1/5})$ samples, it must use "non-local information." We point out that the algorithm of the present paper, though we are in the rather different setting of *sparse* and *circular* trace reconstruction, looks at more global properties of the input (see Algorithm 1 and Theorem 26).

As the general worst-case trace reconstruction problem still has an exponential gap, much work has developed and investigated variants of the problem. [15] investigates many different variants, including matrix reconstruction and trace reconstruction for sparse strings. Most relevant to our work: in the regime where k and p are both constants, they analyze the "trivial" algorithm of directly estimating the size of each gap, giving an $O(n \log n)$ upper bound. (Directly estimating the size of each gap does not work in our circular trace reconstruction setting, since each trace gets randomly cyclically shifted, and thus a gap in a trace could have come from *any* of the k gaps in the original string, and thus we cannot aggregate statistics about any individual gap.) [1] provides an algorithm using $O(n \log n)$ samples for the trace reconstruction problem for "separated" strings, which means the number of 0's between two 1's is at least polylog n. The work of [12] and [21] shows that when the string is confined in a known Hamming ball or an edit distance ball of radius k, there are algorithms that can recover the string with $n^{O(k)}$ traces. [19] investigates the average-case trace reconstruction problem, that is, when the string x is uniformly randomly chosen instead of adversarially generated.They show that, in contrast to the worst-case, it only requires $\exp(O(\log^{1/2} n))$ traces in the average case. [10] investigates the coded version of the trace reconstruction problem, that is, if we code the original string with a high-rate error-correction code, how many traces do we need to recover it? [4] shows that the coded trace reconstruction is nearly equivalent to the average-case trace reconstruction problem, and $\exp(O(\log^{1/3}(1/\epsilon))$ traces are enough to reconstruct the string.

The work of [17] introduces the circular trace reconstruction problem. In this problem, the traces are generated by passing through a deletion channel and then undergoing a random cyclic shift. This problem aims to recover the string or any of its cyclic shifts. They show, via a reduction, that this problem is at least as hard as the linear trace reconstruction problem and provide a $\exp(O(n^{1/3}))$ upper bound when n is a prime or a product of two primes in the worst-case, and a $n^{O_p(1)}$ upper bound for the average case, along with a lower bound of $\tilde{\Omega}(n^3)$. This lower bound comes from showing the indistinguishability of two constant-sparse strings whose cyclic statistics cancel up to order $m = 2$; the current paper extends this construction to $m = 4$ and shows this is almost the best possible.

2 Definitions

We study traces of k-sparse binary strings, meaning the set of binary strings with k nonzeros; we assume k is bounded by a constant. For sequences $x = (x_1, \ldots, x_k)$ and $y = (y_1, \ldots, y_k)$, we say y is a *cyclic shift* of x if there exists an integer c such that $x_j = y_{j+c}$ for all $j \in [k]$, interpreting indices mod k. If no such c exists, then we say x and y are *cyclically distinct*.

Every k-sparse binary string x is a cyclic shift of some string of the form $10^{x_1} \ldots 10^{x_k}$. For our analysis, it will often be useful to represent each string as an integer sequence $x = (x_1, \ldots, x_k)$, where x_j corresponds to the number of zeros in the "gap" between the j^{th} and $j + 1^{\text{st}}$ 1s in x. In this paper, we will generally use the integer sequence representation of x; when we require the binary string instead, we will explicitly write $x = 10^{x_1} \ldots 10^{x_k}$.

We use $[k]$ to refer to the set of integers $\{1, \ldots, k\}$ and $\mathbf{Bin}(z, q)$ to denote a binomial random variable with z trials and success probability q. For two probability measures μ and ν over a set Ω, the Hellinger distance between μ and ν is defined as $d_H(\mu, \nu) := \sqrt{\sum_{x \in \Omega} \left(\sqrt{\mu(x)} - \sqrt{\nu(x)}\right)^2}$.

Most of the technical analysis of this paper concerns understanding properties of "cyclic statistics" of sequences $x_1, \ldots, x_k$, which we define here:

▶ **Definition 4.** *Let $i_1, \ldots, i_m \in [k]$, and ℓ be a divisor of k (with $\ell = 1$ allowed). This defines a* cyclic statistic mod ℓ *of $x = (x_1, \ldots, x_k) \in \mathbb{Z}^k$ defined by $S_{i_1,\ldots,i_m;\ell}(x) := \sum_{j=1}^{k/\ell} x_{i_1+j\ell} \ldots x_{i_m+j\ell}$, (where all indices are interpreted mod k). When $\ell = 1$, we refer to the function as merely a cyclic statistic and write $S_{i_1,\ldots,i_m}(x)$.*

Problem definitions. We investigate the problem of recovering a string from its traces generated by a circular deletion channel, which is defined as follows.

▶ **Definition 5** (Circular deletion channel)**.** *Fix $p \in (0, 1)$. A circular deletion channel, denoted $\boldsymbol{Del}(x)$ takes as input $x \in \{0, 1\}^n$ and induces the probability distribution defined by the following process:*

1. *Delete each bit in x independently with probability p to yield a string $\tilde{x}$.*
2. *Return a uniformly random cyclic shift of $\tilde{x}$.*

In this paper, we focus on the case where p is a universal constant. We are now ready to define the (circular) trace reconstruction problem.

▶ **Definition 6** (Trace reconstruction problem)**.** *Let $x \in \{0, 1\}^n$. Given sample access to traces generated by the deletion channel $\boldsymbol{Del}(x)$, an algorithm $\mathcal{A}$ which returns x' solves the trace reconstruction problem if, with probability at least $2/3$, x' is a cyclic shift of x.*

▶ **Definition 7** (Distinguishing problem)**.** *Let $x, y \in \{0, 1\}^n$ with x and y cyclically distinct. An algorithm $\mathcal{A}$ distinguishes x from y if, given sample access to a deletion channel $\boldsymbol{Del}(z)$, $\mathcal{A}$ returns x with probability at least $2/3$ when z is a cyclic shift of x, and $\mathcal{A}$ returns y with probability at least $2/3$ when z is a cyclic shift of y.*

▶ Remark 8. We note that the two problems above are nearly equivalent for k-sparse strings in the sense that an $O(f(n))$ upper bound for the distinguishing problem implies an $O(f(n) \log n)$ algorithm for reconstruction. This is due to the following standard technique. Suppose algorithm $\mathcal{A}$ distinguishes all pairs $x, y \in \mathcal{X}$ with probability at least $2/3$ using $f(n)$ samples. To reconstruct an unknown string z, we draw $O(f(n) \log n^{2k}) = O_k(f(n) \log n)$ samples from the deletion channel $\mathbf{Del}(z)$, and test each possible pair of cyclically distinct strings x, y against each other using $\mathcal{A}$. By a Chernoff bound, $\mathcal{A}$ will return a valid answer on each pair with probability $1 - O(1/n^{2k})$. Since there are $O(n^k)$ cyclically distinct k-sparse strings of length n, *all* $O(n^{2k})$ tests will succeed with constant probability, in which case we return the unique string z' which is chosen by $\mathcal{A}$ against all other candidates.

3 Cyclic Statistics Characterize Cyclically Distinct Sequences

We begin by studying cyclic statistics of integer sequences, a key component of our upper and lower bounds. The goal of this section is to answer the question: given two cyclically distinct integer sequences x and y, what is the smallest value of m such that there exists a cyclic statistic $S_{i_1,\ldots,i_m}$ satisfying $S_{i_1,\ldots,i_m}(x) \neq S_{i_1,\ldots,i_m}(y)$? We show $m \leq 6$ always suffices.

Our proof relies on the Fourier transforms of x and y, denoted $\hat{x}$ and $\hat{y}$ respectively. We partition the Fourier coefficients $\hat{x}_j$ into equivalence classes based on $\gcd(j, k)$. We show that for each equivalence class, the Fourier coefficients in x are jointly zero or nonzero *together*. Notably, as it turns out, for any x and y with matching second-order cyclic statistics, $\hat{y}$ has identical pattern of zeros and nonzeros as $\hat{x}$ (see Lemma 13).

Armed with this insight, we restrict our analysis to the set of *nonzero* Fourier coefficients in x and y, and define a set of variables $\{z_j = \frac{1}{2\pi i} \log \frac{\hat{x}_j}{\hat{y}_j}\}$ for each index j with nonzero Fourier coefficients in $\hat{x}$ and $\hat{y}$. We show that equivalence of cyclic statistics (up to order 6) imposes a system of linear constraints on the variables z_j, and that any set of solutions must be strictly real and there exists an integer c such that $z_j \equiv \frac{cj}{k} \bmod 1$ for all j. It follows immediately that $\hat{x}_j = \hat{y}_j \exp\left(2\pi i \cdot \frac{cj}{k}\right)$ for all j. By basic properties of Fourier transforms, this implies that x is a cyclic shift of y, proving our upper bound.

The organization of this section is as follows. In Section 3.1, we relate cyclic statistics to the Fourier transform and prove that for any x and y with identical cyclic statistics up to order 6, $\hat{x}$ and $\hat{y}$ have the same pattern of zeros and nonzeros. In Section 3.2 we set up several number theoretic results necessary to analyze the system of linear constraints. Finally, in Section 3.3, we apply these facts to the set of variables $\{z_j\}$ defined above to obtain the main result of this section, Lemma 19, which states that any two cyclically distinct strings x, y must differ in *some* cyclic statistic of order $m \leq 6$.

3.1 Cyclic Statistics and the Fourier Transform

Suppose two integer sequences x and y have identical cyclic statistics up to order m. We show this implies an equivalence between certain m-way products of Fourier coefficients.

▶ **Lemma 9.** *Let x and y be two integer sequences of length k with identical mth order cyclic statistics. Consider their Fourier transforms $\hat{x}$ and $\hat{y}$. For any m-tuple $i_1, \ldots, i_m$ with $i_1 + \cdots + i_m \equiv 0$ mod k, we have $\hat{x}_{i_1} \cdots \hat{x}_{i_m} = \hat{y}_{i_1} \cdots \hat{y}_{i_m}$.*

Proof. Letting $j_2' \equiv j_2 - j_1 \bmod k \ldots, j_m' \equiv j_m - j_1 \bmod k$, and interpreting indices mod k:

$$\begin{aligned}
\hat{x}_{i_1} \cdots \hat{x}_{i_m} &= \left(\sum_{j_1=0}^{k-1} x_{j_1} e^{\frac{2\pi i}{k} j_1 i_1}\right) \cdots \left(\sum_{j_m=0}^{k-1} x_{j_m} e^{\frac{2\pi i}{k} j_m i_m}\right) \\
&= \left(\sum_{j_1=0}^{k-1} x_{j_1} e^{\frac{2\pi i}{k} j_1 i_1}\right) \left(\sum_{j_2'=0}^{k-1} x_{j_2'+j_1} e^{\frac{2\pi i}{k}(j_2'+j_1) i_2}\right) \cdots \left(\sum_{j_m'=0}^{k-1} x_{j_m'+j_1} e^{\frac{2\pi i}{k}(j_m'+j_1) i_m}\right) \\
&= \sum_{j_2', \ldots, j_m'=0}^{k-1} \left(e^{\frac{2\pi i}{k}(j_2' i_2 + \cdots + j_m' i_m)} \sum_{j_1=0}^{k-1} x_{j_1} x_{j_2'+j_1} \cdots x_{j_m'+j_1}\right) \\
&= \sum_{j_2', \ldots, j_m'=0}^{k-1} e^{\frac{2\pi i}{k}(j_2' i_2 + \cdots + j_m' i_m)} S_{0, j_2', \ldots, j_m'}(x)
\end{aligned}$$

where the exponential terms vanish in the second-to-last equality because $i_1 + \cdots + i_m \equiv 0 \bmod k$ and so $e^{\frac{2\pi i}{k} j_1 i_1} \cdots e^{\frac{2\pi i}{k} j_1 i_m} = 1$. Since $S_{0, j_2', \ldots, j_m'}(x) = S_{0, j_2', \ldots, j_m'}(y)$ for all $j_2', \ldots, j_m'$ by assumption, we have that the corresponding m-way products of Fourier terms are equal. ◀

Lemma 9 implies that for any $i_1, \ldots, i_m$ satisfying $i_1 + \cdots + i_m \equiv 0 \bmod k$, if the Fourier coefficients $\hat{x}_{i_1}, \ldots, \hat{x}_{i_m}$ are nonzero, then $\hat{y}_{i_1}, \ldots, \hat{y}_{i_m}$ must be nonzero as well. We now partition the elements of $[k]$ into equivalence classes based on their gcd with k.

▶ **Definition 10.** *Let α be a divisor of k. We define $G_\alpha \coloneqq \{j \in [k] : \gcd(j, k) = \alpha\}$.*

We show that for a single integer sequence x, the Fourier coefficients of x in each G_α are either *all* zero or *all* nonzero, allowing us to analyze them together. This can be shown using a well-known property of cyclotomic fields: for every primitive dth root of unity of form $\omega_d = e^{2\pi i/d}$, the Galois group $\mathrm{Gal}(\mathbb{Q}(\omega_d)/\mathbb{Q})$ is isomorphic to $(\mathbb{Z}/d\mathbb{Z})^\times \coloneqq \{j \in \{0, \ldots, d-1\} : \gcd(j, k) = 1\}$, the multiplicative group of integers mod d. The isomorphism is specified by Fact 11.

▶ **Fact 11.** *Let $d \in \mathbb{N}$ and let $\omega_d \coloneqq e^{2\pi i/d}$ be a primitive dth root of unity. Consider $\mathbb{Q}(\omega_d)$, the field extension of $\mathbb{Q}$ generated by ω_d. For all $a \in (\mathbb{Z}/d\mathbb{Z})^\times$, there exists an automorphism $\sigma_a \in Gal(\mathbb{Q}(\omega_d)/\mathbb{Q})$ such that $\sigma_a(\omega_d) = \omega_d^a$.*[1]

▶ **Lemma 12.** *Let x be an integer sequence of length k with Fourier transform $\hat{x}$. For any $\alpha \in [k]$ and $j, j' \in G_\alpha$, if $\hat{x}_j = 0$, then $\hat{x}_{j'} = 0$.*

Proof. Suppose $\hat{x}_j = 0$. Let $d \coloneqq \frac{k}{\alpha}$ and consider the a primitive dth root of unity $\omega_d \coloneqq e^{2\pi i/d}$. The jth Fourier coefficient of x is given by

$$\hat{x}_j = \sum_{\ell=0}^{k-1} x_\ell e^{(2\pi i/k)j\ell} = \sum_{\ell=0}^{k-1} x_\ell e^{(2\pi i/d)\cdot(d/k)\cdot j\ell} = \sum_{\ell=0}^{k-1} x_\ell \omega_d^{(j/\alpha)\ell}$$

Let z be a sequence of length d generated by interpreting each index of x mod d and summing together all elements that map to a given location. (In other words, $z_j \coloneqq \sum_{\ell:\ell\equiv j \bmod d} x_\ell$.) Then the Fourier transform of z at location $\frac{j}{\alpha}$ is given by

$$\hat{z}_{j/\alpha} = \sum_{\ell=0}^{d-1} z_\ell \omega_d^{(j/\alpha)\ell} = \sum_{\ell=0}^{k-1} x_\ell \omega_d^{(j/\alpha)(\ell \bmod d)} = \sum_{\ell=0}^{k-1} x_\ell \omega_d^{(j/\alpha)\ell} = \hat{x}_j$$

Similarly, $\hat{z}_{j'/\alpha} = \hat{x}_{j'}$. Because z is an integer sequence, $\hat{z}_{j/\alpha}$ and $\hat{z}_{j'/\alpha}$ are elements of the cyclotomic field $\mathbb{Q}(\omega_{d/\alpha})$. Choose $b \in (\mathbb{Z}/d\mathbb{Z})^\times$ such that $b\frac{j}{\alpha} \equiv \frac{j'}{\alpha} \bmod d$. By Fact 11, there exists $\sigma_b \in \mathrm{Gal}(\mathbb{Q}(\omega_d)/\mathbb{Q})$ such that $\sigma_b(\omega_d) = \omega_d^b$. Therefore, we have

$$\hat{x}_{j'} = \hat{z}_{b\cdot(j/\alpha) \bmod d} = \sum_{\ell=0}^{d-1} z_\ell \cdot \omega_d^{b(j/\alpha)\ell} = \sum_{\ell=0}^{d-1} z_\ell \cdot \sigma_b\left(\omega_d^{(j/\alpha)\ell}\right) = \sigma_b(\hat{z}_{j/\alpha}) = \sigma_b(\hat{x}_j) = 0$$

as desired. ◀

Consider two sequences x and y with identical second order cyclic statistics. As a simple consequence of Lemmas 9 and 12, $\hat{x}$ and $\hat{y}$ are jointly zero or jointly nonzero everywhere in the indices belonging to a single G_α.

▶ **Lemma 13.** *Let x and y be two integer sequences of length k with identical second order cyclic statistics. Then for each $\alpha \in [k]$, either (1) $\hat{x}_j \neq 0$ and $\hat{y}_j \neq 0$ for all $j \in G_\alpha$ or (2) $\hat{x}_j = \hat{y}_j = 0$ for all $j \in G_\alpha$.*

Proof. Suppose $\hat{x}_j \neq 0$ for some $j \in G_\alpha$. By Lemma 12, we must have $\hat{x}_{j'} \neq 0$ for all $j' \in G_\alpha$. Note that $k - j \in G_\alpha$, so $\hat{x}_{k-j} \neq 0$. Moreover, $j + (k - j) \equiv 0 \bmod k$, so by Lemma 9, we have $\hat{y}_j \cdot \hat{y}_{k-j} = \hat{x}_j \cdot \hat{x}_{k-j} \neq 0$. This requires that $\hat{y}_j \neq 0$. Applying Lemma 12 to y, we have that $\hat{y}$ is nonzero for all indices in G_α, proving the first statement.

If $x_j = 0$, then similarly $\hat{y}_j \cdot \hat{y}_{k-j} = \hat{x}_j \cdot \hat{x}_{k-j} = 0$, requiring that either $\hat{y}_j = 0$ or $\hat{y}_{k-j} = 0$. By Lemma 12, $\hat{x}$ and $\hat{y}$ must be zero everywhere in G_α, proving the second statement. ◀

3.2 Number Theoretic Prerequisites

Recall that our goal is to analyze a system of constraints involving the nonzero Fourier coefficients of x and y, where the constraints arise from the assumption that x and y have identical cyclic statistics up to order 6. These constraints will turn out to take the form

[1] For more on cyclotomic fields, see `https://en.wikipedia.org/wiki/Cyclotomic_field`.

$z_{j_1} + \cdots + z_{j_m} \in \mathbb{Z}$, for a certain set of complex valued variables $\{z_j\}$ and certain m tuples $j_1, \ldots, j_m$. We wish to show that any set of variables satisfying these constraints must be strictly real and form an arithmetic sequence mod 1. In Lemma 16, we show this is true; we apply this fact in Section 3.3 to show that x and y are cyclically equivalent.

Before proving this relation, we require certain number theoretic facts. First, we show that for any integer d and $j \in \mathbb{Z}/d\mathbb{Z}$, j can be written as a sum (mod d) of either 2 or 3 numbers relatively prime to d. Here and below, we use $\mathbb{Z}/d\mathbb{Z}$ to refer to the cyclic group of integers modulo d, represented as the set $\{0, \ldots, d-1\}$.

▶ **Fact 14.** *The following facts are true for any prime power p^a and integer j:*

- *If $p \neq 2$, there exists a pair $b_1, b_2 \in [p^a]$ that are relatively prime to p such that $b_1 + b_2 \equiv j \mod p^a$. Also, there exists a triple $b_1, b_2, b_3 \in [p^a]$ that are relatively prime to p such that $b_1 + b_2 + b_3 \equiv j \mod p^a$.*
- *If $p = 2$ and j is even, there exists a pair $b_1, b_2 \in [p^a]$ that are relatively prime to p such that $b_1 + b_2 \equiv j \mod p^a$.*
- *If $p = 2$ and j is odd then there exists a triple $b_1, b_2, b_3 \in [p^a]$ that are relatively prime to p such that $b_1 + b_2 + b_3 \equiv j \mod p^a$.*

Proof. First, consider the case in which $p \neq 2$. At least one of $j-1, j+1$ must be relatively prime to p^a since their difference is 2, which itself is relatively prime to p. Since $j \equiv (j-1) + 1 \bmod p^a$, $j \equiv (j+1) + (p^a - 1) \bmod p^a$, and both 1 and $p^a - 1$ are relatively prime to p^a, at least one of these two expressions expresses j as the sum of 2 numbers both relatively prime to p^a. Similarly, at least one of $j-2, j+2$ must be relatively prime to p^a since their difference is 4, which itself is relatively prime to p. Thus since $j \equiv (j-2)+1+1 \bmod p^a$ and $j \equiv (j+2) + (p^a - 1) + (p^a - 1) \bmod p^a$, and since both 1 and $p^a - 1$ are relatively prime to p^a, at least one of these two expressions expresses j as the sum of 3 numbers both relatively prime to p^a.

For the case of $p = 2$, we note that if j is even, then $j - 1$ is relatively prime to 2^a and thus we express $j = (j-1) + 1$. Otherwise, if j is odd, then $j - 2$ is relatively prime to 2^a, and we express $j = (j-2) + 1 + 1$. ◀

▶ **Lemma 15.** *Let $d \in \mathbb{N}$. The following is true for all $j \in \mathbb{Z}/d\mathbb{Z}$:*

- *If d is even and j is odd, then j can be written as the sum mod d of 3 numbers relatively prime to d.*
- *Else j can be written as the sum mod d of 2 numbers relatively prime to d.*

Proof. Consider the prime factorization of d: $d = p_1^{a_1} \cdot \ldots \cdot p_\ell^{a_\ell}$. Given $j \in \mathbb{Z}/d\mathbb{Z}$, we use the Chinese remainder theorem to express j via its residues mod $p_i^{a_i}$, for each prime factor p_i. Define $j_i \coloneqq j \bmod p_i^{a_i}$. We note that j is relatively prime to d if and only if each j_i is relatively prime to its corresponding prime power $p_i^{a_i}$.

We use Fact 14 (and the Chinese remainder theorem) to canonically represent each $j \in \mathbb{Z}/d\mathbb{Z}$ as the sum mod d of either 2 or 3 numbers that are each relatively prime to d. If either d is odd or j and d are both even, we represent j as the sum of 2 numbers, by applying Fact 14 to represent each j_i as $j_i \equiv b_i^{(1)} + b_i^{(2)} \bmod p_i^{a_i}$ with $b_i^{(1)}$ and $b_i^{(2)}$ relatively prime to $p_i^{a_i}$. By the Chinese remainder theorem, the system, $b^{(1)} \equiv b_i^{(1)} \bmod p_i^{a_i}, \forall i \in [\ell]$, has a unique solution $b^{(1)}$ relatively prime to d; similarly, there exists a unique $b^{(2)}$ relatively prime to d satisfying $b^{(2)} \equiv b_i^{(2)} \bmod p_i^{a_i}$ for all i. Finally, $b^{(1)} + b^{(2)} \equiv j \bmod d$, which proves the first case. If j is odd and d is even, we use the analogous procedure to represent j as the sum of 3 numbers relatively prime to d, which proves the second case. ◀

Using the representation given by Lemma 15, we analyze the system of complex-valued linear equations described at the beginning of this subsection.

▶ **Lemma 16.** *Let $d \in \mathbb{N}$ and $z_j \in \mathbb{C}$ for each $j \in [d]$ relatively prime to d. Suppose that for every $m \leq 6$, and any m-tuple $(j_1, \ldots, j_m)$ relatively prime to d satisfying $j_1 + \cdots + j_m \equiv 0 \pmod{d}$, we have $z_{j_1} + \cdots + z_{j_m} \in \mathbb{Z}$. Then there exists an integer c such that for all j relatively prime to d we have $z_j - \frac{cj}{d} \in \mathbb{Z}$.*

Proof. For all $j \in \mathbb{Z}/d\mathbb{Z}$, let $[[j]]$ denote the canonical representation of j given by Lemma 15 as the sum mod d of either 2 or 3 numbers relatively prime to d (so $[[j]]$ is a set of size 2 or 3). Let $z_{[[j]]}$ denote the sum of the 2 or 3 corresponding variables. For convenience, we write the set $d - [[j]] \coloneqq \{d - j' : j' \in [[j]]\}$. Then the conditions of the lemma imply the following set of conditions:

$$\forall j \text{ relatively prime to } d: \ z_j + z_{d-j} \in \mathbb{Z},$$
$$\forall j \text{ relatively prime to } d: \ z_{[[j]]} + z_{d-j} \in \mathbb{Z},$$
$$\forall j \in \mathbb{Z}/d\mathbb{Z}: \ z_{[[j]]} + z_1 + z_{d-[[j+1]]} \in \mathbb{Z}$$

To see the implication, it suffices to show that the left hand side of each constraint is a sum of at most 6 real variables whose indices are relatively prime to d. For all j with $\gcd(j, d) = 1$, we have $\gcd(d - j, d) = 1$, so the first constraint satisfies the condition. For all j, $z_{[[j]]}$ can be written as a sum of 2 or 3 variables whose indices are relatively prime to d, so the second constraint requires at most 4 variables. Finally, if d is odd, the representation $[[j]]$ comprises 2 variables for all j, in which case the third constraint requires 5 variables. If d is even, the values j and $j + 1$ comprise 1 even number and 1 odd number, and $d - [[j + 1]]$ has the same parity as $j + 1$, resulting in at most 6 terms.

Note that these three sets of conditions collectively imply that $z_{[[j]]} + z_1 - z_{[[j+1]]} \in \mathbb{Z}$ for all $j \in \mathbb{Z}/d\mathbb{Z}$. Therefore, any set of solutions $z_{[[j]]}$ must form an arithmetic sequence of increment z_1 plus some integer, i.e., $z_{[[j]]} - jz_1 \in \mathbb{Z}$ for all j. Therefore, $dz_1 \in \mathbb{Z}$, which requires that z_1 is a multiple of $\frac{1}{d}$ (and is therefore real). Writing $z_1 = \frac{c}{d}$ for some integer c, we have $z_{[[j]]} \equiv \frac{cj}{d} \bmod 1$ for all $j \in \mathbb{Z}/d\mathbb{Z}$. Finally, the first and second conditions imply that $z_j = z_{[[j]]}$ for all j relatively prime to d, which proves the lemma. ◀

The following technical lemma is used in the proof of the main result of this section, Lemma 19. Intuitively, in the proof of Lemma 19 we aim to show that variables $z_0, \ldots, z_{k-1}$ form an arithmetic sequence. Instead we show that, for each value α, those z_j with index j satisfying $gcd(j, k) = \alpha$ form an arithmetic sequence of increment c_α. We use the following lemma to show that this implies there is a consistent arithmetic sequence across all j, with some increment c.

▶ **Definition 17.** *For a positive integer $a \in \mathbb{N}$ and prime number p, define the p-adic valuation of a to be $\nu_p(a) \coloneqq \max\{\ell \in \mathbb{Z}_{\geq 0} : p^\ell \mid a\}$.*

▶ **Lemma 18.** *Let $k \in \mathbb{N}$ and $A \subseteq [k]$. Let $c_\alpha \in \mathbb{Z}$ for each $\alpha \in A$, and suppose the following are true for all pairs $\alpha, \alpha' \in A$:*

1. *$lcm(\alpha, \alpha') \in A$.*
2. *$(c_{\alpha'} - c_\alpha) \cdot lcm(\alpha, \alpha') \equiv 0 \bmod k$*

Then there exists $c \in \mathbb{Z}$ such that $(c - c_\alpha)\alpha \equiv 0 \bmod k$ for all $\alpha \in A$.

Proof. Consider the prime factorization of k, $k = p_1^{a_1} \ldots p_\ell^{a_\ell}$. For each $i \in [\ell]$, we will show there exists c_i satisfying the desired property mod $p_i^{a_i}$, and conclude the lemma by the Chinese remainder theorem.

Fix $i \in [\ell]$. Define α_i to be the element of A with the lowest p_i-adic valuation: $\alpha_i = \arg\min_{\alpha \in A} \nu_{p_i}(\alpha)$. Let c_i be an integer satisfying $(c_i - c_{\alpha_i})\alpha_i \equiv 0 \bmod p_i^{a_i}$, and consider any $\alpha' \in A$. By a well-known property of p-adic valuations, we have $\nu_{p_i}(\mathrm{lcm}(\alpha', \alpha_i)) = \max\{\nu_{p_i}(\alpha'), \nu_{p_i}(\alpha_i)\} = \nu_{p_i}(\alpha')$. The prime power $p_i^{\nu_{p_i}(\alpha')}$ divides α', so it also divides $\mathrm{lcm}(\alpha', \alpha_i)$. Therefore, we have the following two facts:

$$(c_{\alpha_i} - c_{\alpha'})p^{\nu_{p_i}(\alpha')} \equiv 0 \bmod p_i^{a_i} \qquad \text{(by assumption (2) of the lemma)}$$

$$(c_i - c_{\alpha_i})p_i^{\nu_{p_i}(\alpha_i)} \equiv 0 \bmod p_i^{a_i} \qquad \text{(by definition of } c_i\text{)}$$

Using the fact that $\nu_{p_i}(\alpha') \geq \nu_{p_i}(\alpha_i)$ to combine the two equations, we have $(c_i - c_{\alpha'})p_i^{\nu_{p_i}(\alpha')} \equiv 0 \bmod p_i^{a_i}$. Finally, we multiply both sides by the integer $\alpha' p_i^{-\nu_{p_i}(\alpha')}$ to yield $(c_i - c_{\alpha'})\alpha' \equiv 0 \bmod p_i^{a_i}$. Choosing $c_i = c_{\alpha_i}$ for all i, the equation above holds for all pairs i, α. We now wish to find c such that $c \equiv c_i \bmod p_i^{a_i}$ for all i. Such an integer c is guaranteed to exist by the Chinese remainder theorem, which completes the proof. ◀

3.3 Main Characterization

We now prove the main result of this section.

▶ **Lemma 19.** *Let x and y be two integer sequences of length k with identical cyclic statistics up to order 6. Then x and y must be identical up to a cyclic shift.*

Proof. It suffices to show that there exists an integer c such that $\hat{x}_j = \hat{y}_j \cdot \exp\left(2\pi i \cdot \frac{cj}{k}\right)$ for all $j \in [k]$. By Lemma 13, this property holds for all j where $\hat{x}_j = 0$. Therefore, we consider only the divisors α such that the Fourier coefficients are nonzero everywhere in G_α for both x and y. For all j with nonzero Fourier coefficients $\hat{x}_j$ and $\hat{y}_j$, define $z_j \coloneqq \frac{1}{2\pi i} \cdot \log\left(\frac{\hat{x}_j}{\hat{y}_j}\right)$, where log is the unique complex-valued function satisfying $e^{\log z} = z$ and $\mathrm{Im} \log z \in (0, 2\pi]$ for all $z \in \mathbb{C}\backslash\{0\}$. To prove the lemma, we show that there exists an integer c such that for all j with $\hat{x}_j \neq 0$ and $\hat{y}_j \neq 0$, $\mathrm{Im} z_j = 0$ and $z_j \equiv \frac{cj}{k} \bmod 1$.

Fix $\alpha \in [k]$ and let $d \coloneqq \frac{k}{\alpha}$. Note that for all $j \in G_\alpha$, $\gcd\left(\frac{j}{\alpha}, d\right) = 1$. Therefore, each $j \in G_\alpha$ corresponds to a unique number relatively prime to d. Consider any m-tuple $j_1, \ldots, j_m \in G_\alpha$ such that $\frac{j_1}{\alpha} + \ldots + \frac{j_m}{\alpha} \equiv 0 \bmod d$. By Lemma 9, we have $z_{j_1} + \ldots + z_{j_m} \in \mathbb{Z}$. Applying Lemma 16 (and using the fact that $k = d\alpha$), there exists an integer c_α such that for all $j \in G_\alpha$, we have $z_j - \frac{c_\alpha j}{k} \in \mathbb{Z}$. In other words, z_j is real and $z_j \equiv \frac{c_\alpha j}{k} \bmod 1$.

We have shown that for all G_α with nonzero Fourier coefficients, we can write $kz_j \equiv c_\alpha j \bmod k$. We will now show that this is true for a *single* consistent integer $c = c_\alpha$ across all α. Consider $\alpha \neq \alpha'$ where $\hat{x}$ and $\hat{y}$ are nonzero everywhere in $G_\alpha \cup G_{\alpha'}$. Let $c_\alpha, c_{\alpha'}$ respectively be the corresponding coefficients, and $\ell = \mathrm{lcm}(\alpha, \alpha')$. Since ℓ is a multiple of α, Lemma 15 implies that we can write ℓ/α as a sum mod k/α of 2 or 3 numbers relatively prime to k/α. Multiplying by α, we can write ℓ as a sum mod k of 2 or 3 numbers whose gcd with k is α; denote this representation as $[[\ell]]$. Since ℓ is also a multiple of α', it must also have a representation $[[\ell]]'$ as a sum mod k of 2 or 3 numbers whose gcd with k is α'.

We write $z_{[[\ell]]}$ and $z_{[[\ell']]}$ to denote the sum of the variables corresponding to the indices in $[[\ell]]$ and $[[\ell']]$ respectively, and use the convention that $k - [[\ell]] = \{k - j : j \in [[\ell]]\}$. Since $\frac{j}{\alpha} + \frac{k-j}{\alpha} \equiv 0 \bmod \frac{k}{\alpha}$ for all $j \in G_\alpha$, we have $z_\ell + z_{k-[[\ell]]} \equiv 0 \bmod 1$ for all ℓ. From the previous paragraph, we have that the sum of the elements in $[[\ell]]$ equals the sum of the elements in $[[\ell]]'$ mod k. We thus conclude the condition that ℓ has a *consistent* coefficient in both representations, as $z_{[[\ell]]} = z_{[[\ell]]'}$, and reexpress this as a sum of 4 to 6 coefficients, moving everything to the left-hand side, as $z_{[[\ell]]} + z_{k-[[\ell]]'} \equiv 0 \bmod 1$. Recall

that $[[\ell]]$ only contains indices j with $\gcd(j,k)=\alpha$, and the corresponding variables z_j satisfy $kz_j \equiv c_\alpha j \bmod k$. Similarly, $[[\ell]]'$ only contains indices j' with $\gcd(j',k)=\alpha'$, and each $z_{j'}$ satisfies $kz_{j'} \equiv c_{\alpha'} j' \bmod k$. Therefore, the constraint $(c_{\alpha'} - c_\alpha)\ell \equiv 0 \bmod k$ holds for any pair of $\alpha, \alpha' \in A \subseteq [k]$, where $A = \{\alpha : \hat{x}, \hat{y} \text{ are nonzero everywhere in } G_\alpha\}$.

Applying Lemma 18, there exists $c \in \mathbb{Z}$, such that $(c - c_\alpha)\alpha \equiv 0 \bmod k$ for all $\alpha \in A$. Therefore, for every $\alpha \in A$, for every $j \in G_\alpha$, $c\alpha\frac{j}{\alpha} \equiv c_\alpha \alpha \frac{j}{\alpha} \bmod k$. Notice that by definition, $\frac{j}{\alpha}$ is an integer and thus, for every $\alpha \in A$, for every $j \in G_\alpha$, $cj \equiv c_\alpha j \bmod k$. Therefore, for every j where $\hat{x}_j$, $\hat{y}_j$ are nonzero, we have $kz_j \equiv c_\alpha j \equiv cj \bmod k$. Dividing by k, we have $z_j \equiv \frac{cj}{k} \bmod 1$ for every j where $\hat{x}_j$, $\hat{y}_j$ are nonzero, as desired. ◀

4 Upper Bound

We now give an algorithm for distinguishing any two cyclically distinct strings with a constant number of 1s. Given Lemma 19, a naïve approach for testing whether a set of traces is generated from x or y is to identify a cyclic statistic in which x and y differ, and then estimate this statistic from traces. For a string with k 1s, the deletion channel will preserve all 1s with constant probability; a resulting trace can be represented as a sequence of k "gaps," for which we can compute cyclic statistics. In particular, for a string $x = (x_1, ..., x_k)$ and retention probability $q := 1 - p$, a trace can be represented as a sequence of binomial random variables $\tilde{x} = (\tilde{x}_1, ..., \tilde{x}_k)$, where $\tilde{x}_j \sim \mathbf{Bin}(x_j, q)$.

Unfortunately, computing a cyclic statistic of $\tilde{x}$ requires multiplying up to 6 of these binomial random variables. In the worst case, we might have $x_j = \Omega(n)$ for all j, causing our estimator to have variance roughly n^{11}. Since cyclic statistics are integers, we may need to estimate the desired cyclic statistic to within constant error, and thus the naïve algorithm would require $\Omega(n^{11})$ samples.

We give a more efficient algorithm using the following insight: while the product of 6 independent binomials with $\Omega(n)$ trials each has variance roughly $\Omega(n^{11})$, the variance is considerably lower if each binomial is shifted to have mean $\tilde{O}(\sqrt{n})$. To accomplish this, we preprocess x and y by greedily grouping the values $qx_1, \ldots, qx_k, qy_1, \ldots, qy_k$ that are within $\tilde{O}(\sqrt{n})$ of each other into a cluster. When processing a trace $\tilde{z}$ from either x or y, we compute the cluster center closest to each element $\tilde{z}_j$, and subtract that center from $\tilde{z}_j$. Because the gap sizes $\tilde{z}_j$ are highly likely to be within $\tilde{O}(\sqrt{n})$ of their expectations, we can find the "correct" cluster with high probability. The resulting estimator has variance $\tilde{O}(n^6)$.

Of course, subtracting an arbitrary sequence s from the observed trace $\tilde{z}$ raises another concern: the sequences $qx - s$ and $qy - s$ might now be cyclically equivalent, erasing any differences in their cyclic statistics. We circumvent this problem with a slightly different estimator which only considers cyclic statistics *mod* ℓ where ℓ is chosen so that s is preserved by a cyclic shift of ℓ.

The remainder of this section is organized as follows. In Section 4.1, we give an algorithm for generating a well separated partition of the expectations $qx_1, \ldots, qx_k, qy_1, \ldots, qy_k$. In Section 4.2, we show that cyclic statistics mod ℓ are preserved by subtraction of a sequence s invariant to cyclic shifts of ℓ. Finally, in Section 4.3, we present our cyclic statistics-based algorithm and prove its correctness.

4.1 Determining Centers

As a preprocessing step, our algorithm partitions the means of $2k$ binomial random variables (the gaps in traces from x and y) into well separated clusters. For our purposes, clusters are considered to be well separated if they are $\Omega(\sqrt{n}\log n)$ far apart; we give a formal definition below.

▶ **Definition 20.** *Let $x_1, \ldots, x_k \in \mathbb{R}$ and $C > 0$. Then a function $c : \mathbb{R} \to [k]$ is a C-separated partition of $x_1, \ldots, x_k$ if it satisfies the following:*

1. *For any $j, j' \in [k]$, if $c(x_j) \neq c(x_{j'})$, then $|qx_j - qx_{j'}| > 2C\sqrt{n}\log n$.*
2. *For any $j, j' \in [k]$, if $c(x_j) = c(x_{j'})$, then $|qx_j - qx_{j'}| \leq 2Ck\sqrt{n}\log n$.*
3. *For all $x \in \mathbb{R}$, $c(x) = c\left(\arg\min_{x_j : j \in [k]} |x_j - x|\right)$.*

We show that a C-separated partition exists for any $C > 0$ and $x_1, \ldots, x_k \in \mathbb{R}$. The proof is constructive: we give a greedy algorithm for producing a C-separated partition of a fixed set of points $x_1, \ldots, x_k$. The algorithm starts by assigning each point to its own cluster; it then iteratively merges clusters which violate the separation condition. When no violations remain, we prove that the algorithm has obtained a C-separated partition.

▶ **Lemma 21.** *Let $x_1, \ldots, x_k \in [n]$, and $C > 0$. Then there exists a C-separated partition of $x_1, \ldots, x_k$.*

4.2 Uniqueness of Cyclic Statistics

As discussed at the beginning of this section, our goal is to reduce the variance of our estimator by shifting each binomial random variable $\tilde{x}_1, \ldots, \tilde{x}_k$ by a value which is within a constant of its mean. In particular, let c be a C-separated partition of $qx_1, \ldots, qx_k, qy_1, \ldots, qy_k$, and $s = (c(qx_1), \ldots, c(qx_k))$. The binomial random variables concentrate tightly around their expectation, allowing us to approximately recover the partition of x and y generated by the partitioning algorithm with high probability. The logical approach is therefore to estimate cyclic statistics of $x - s$.

Unfortunately, this introduces a second problem. If x and y are distinct after applying the partition c elementwise, then they can be distinguished simply by recovering the partition. But if x and y appear similar under the partition – that is, $s_x = (c(qx_1), \ldots, c(qx_k))$ is a cyclic shift of $s_y = (c(qy_1), \ldots, c(qy_k))$ – then $x - s_x$ might be a cyclic shift of $y - s_y$, resulting in identical cyclic statistics and thus a failure to distinguish x from y.

▶ **Example 22.** Consider the case where $k = 6$, and where the sequence of shifts/clusters $s_x = s_y$ follows the pattern s, s', s', s, s', s'. Further, let $x = (\frac{s}{q}+0, \frac{s'}{q}+1, \frac{s'}{q}+2, \frac{s}{q}+0, \frac{s'}{q}+1, \frac{s'}{q}+2)$ and $y = (\frac{s}{q}+1, \frac{s'}{q}+2, \frac{s'}{q}+0, \frac{s}{q}+1, \frac{s'}{q}+2, \frac{s'}{q}+0)$. In this case, when we sample from $\mathbf{Del}(x)$ or $\mathbf{Del}(y)$, once we subtract $s_x = s_y$, we get two 6-tuples that are cyclically identical; however, algorithmically, we would hope to be able to distinguish traces from x vs y, since the *alignment* between the sequence of shifts $s_x = s_y$ vs the sequence of offsets $(0, 1, 2, 0, 1, 2)$ differs for x vs y. The right thing to do here, after subtracting $s_x = s_y$, is to look at *cyclic statistics mod 3*.

This motivates our general strategy of looking at cyclic statistics mod ℓ, where ℓ is the smallest period of repetition of the shift sequence.

▶ **Definition 23.** *A sequence $s = (s_1, \ldots, s_k) \in \mathbb{R}^k$ is symmetric mod ℓ if for all $j \in [k]$, $s_j = s_{(j+\ell) \bmod k}$.*

▶ **Definition 24.** *Let $x = (x_1, \ldots, x_k) \in \mathbb{Z}^k$, $i_1, \ldots, i_m \in [k]$, and ℓ be a divisor of k. For a sequence $s = (s_1, \ldots, s_k) \in \mathbb{R}^k$, the function $S_{i_1,\ldots,i_m;\ell}(x - s) := \sum_{j=1}^{k/\ell}(x_{i_1+j\ell} - s_{i_1+j\ell}) \cdots (x_{i_m+j\ell} - s_{i_m+j\ell})$ is an m-th order cyclic statistic mod ℓ, shifted by s.*

We show that for any period ℓ and set of shifts s that repeats mod ℓ, there is a ≤ 6-th order statistic that differs on x, y, when shifted by s and taken mod ℓ. We prove this by starting with Lemma 19 that guarantees the existence of a statistic of order ≤ 6 that

Algorithm 1 Test-Cyclic-Traces.

Input: $k \geq 0; x, y \in \mathbb{N}^k$; $q \in (0,1)$; sample access to deletion channel
Output: x or y
1 $C \leftarrow$ absolute constant determined by Lemma 27
2 $c \leftarrow \textsc{Partition}(qx_1, \ldots, qx_k, qy_1, \ldots, qy_k; C)$
3 $s_x \leftarrow (c(qx_1), \ldots, c(qx_k))$, $s_y \leftarrow (c(qy_1), \ldots, c(qy_k))$
4 **if** s_x *is a cyclic shift of* s_y **then**
5 **return** Test-Similar-Traces(k, x, y, q, s_x, C)
else
6 Draw $\Theta(1)$ traces from the deletion channel
7 $\tilde{z} \leftarrow$ any trace with exactly k 1s
8 **if** $c(\tilde{z}_1), \ldots, c(\tilde{z}_k)$ is a cyclic shift of s_x **then return** x **else return** y

distinguishes x from y; we show this implies existence of a distinguishing statistic of the same order mod ℓ. Taking the distinguishing statistic of *smallest* order lets us subtract off any sequence s that is symmetric mod ℓ, without modifying the discrepancy between x and y. This yields:

▶ **Lemma 25.** *Let x and y be two cyclically distinct integer sequences of length k. Let ℓ divide k, and let $s = (s_1, \ldots, s_k) \in \mathbb{R}^k$ be a sequence that is symmetric mod ℓ. Then there exists $i_1, \ldots, i_m \in [k]$ with $m \leq 6$ such that*

$$|S_{i_1,\ldots,i_m;\ell}(x - s) - S_{i_1,\ldots,i_m;\ell}(y - s)| = |S_{i_1,\ldots,i_m;\ell}(x) - S_{i_1,\ldots,i_m;\ell}(y)| \geq 1.$$

4.3 Our Algorithm

We are now ready to describe our proposed tester, Algorithm 1. Given two candidates x and y, a trace $\tilde{z}$ from the deletion channel can be described as a sequence of binomial random variables $\tilde{z}_1, \ldots, \tilde{z}_k$, each of which has an expectation in the set $\{qx_1, \ldots, qx_k, qy_1, \ldots, qy_k\}$. The algorithm begins by preprocessing the strings x and y to create a C-separated partition c of the expectations (Line 2). We then consider the sequences s_x and s_y created by applying c elementwise to x and y. We design two tests for two separate cases:

- If s_x is a cyclic shift of s_y, then traces from x and y appear relatively similar. In this case, we invoke a subroutine (Algorithm 2), which determines a shifted cyclic statistic in which x and y differ, then estimates this statistic from traces. Owing to the shift, the resulting estimate has low variance, allowing us to estimate the cyclic statistic from $\tilde{O}(n^6)$ traces.
- If s_x and s_y are cyclically distinct, then the algorithm can reliably distinguish x and y by examining only a single trace whose 1s are intact. Therefore, we draw a constant number of traces, choose an arbitrary trace $\tilde{z}$ with k 1s (Line 7), and cluster the gaps according to c. For C large enough, the resulting sequence will with high probability correspond exactly to either s_x or s_y.

Combining the results from each case yields our overall upper bound. We now state and prove the main result of Section 4. Our proof relies on Lemmas 27 and 28 in Section 4.3.2, which prove the correctness of Algorithm 2.

4.3.1 Proof of Upper Bound

▶ **Theorem 26.** *Let x and y be two cyclically distinct k-sparse binary strings of length n. Then Algorithm 1 distinguishes x from y with probability $\geq \frac{2}{3}$ for sufficiently large n, using $O(n^6 \log^{12} n)$ traces from a cyclic deletion channel.*

Proof. We analyze Algorithm 1 to show the theorem. We consider two cases:

Case 1: s_x is a cyclic shift of s_y. In this case, we invoke Algorithm 2, which draws $O(n^6 \log^{12} n)$ returns the correct answer with probability at least 2/3, by Lemma 28.

Case 2: s_x is not a cyclic shift of s_y. Then we draw $T = \log(1/4)/\log(1-q^k) = O(1)$ traces. With probability 3/4, at least one trace $\tilde{z}$ will contain exactly k 1s. Suppose $\tilde{z} \sim \mathbf{Del}(x)$. Due to Lemma 27, with probability at least $1 - O(n^{-10})$, we have $c(\tilde{z}_j) = c(qx_j)$ for all $j \in [k]$. Therefore, $c(\tilde{z}) = s_x$, and the algorithm returns x with probability at least $3/4 - O(n^{-10}) \geq 2/3$ for large enough n.

In both cases, the number of traces T is bounded by $O(n^6 \log^{12} n)$. ◀

4.3.2 Distinguishing Similar Strings

We now handle the case in which $s_x = s_y$, i.e., x and y are cyclically equivalent under the C-separated partition c constructed in Algorithm 1. We build an estimator based on shifted cyclic statistics and show that $\tilde{O}(n^6)$ traces suffice to distinguish x from y. In particular, let ℓ be the minimum value such that s_x and s_y are symmetric mod ℓ. For each $j \in [k]$, let $P_j \coloneqq \{x_{j'} : c(qx_{j'}) = j\} \cup \{y_{j'} : c(qy_{j'}) = j\}$. Define the function $g : [k] \to \mathbb{R}$ to be the average of the points in cluster j; that is, $g(j) = \frac{1}{|P_j|} \sum_{z \in P_j} qz$. The algorithm chooses the smallest value of m for which there is an order m cyclic statistic $S_{i_1,\ldots,i_m;\ell}$ such that $S_{i_1,\ldots,i_m;\ell}(x) \neq S_{i_1,\ldots,i_m;\ell}(y)$. We call a trace $\tilde{z}$ **useful** if it satisfies the following conditions:

1. There are exactly k 1s in $\tilde{z}$.
2. For all $j \in [k]$, $|\tilde{z}_j - g(c(\tilde{z}_j))| \leq (4k+1)C\sqrt{n}\log n$.

Intuitively, the second condition says that the gaps in $\tilde{z}$ can be matched up with cluster centers as we expect. Suppose there are T' useful traces. Our estimator will apply the statistic $S_{i_1,\ldots,i_m;\ell}(x)$ to all the useful traces after subtracting out their cluster centers:

$$f_{i_1,\ldots,i_m;\ell}(\tilde{z}) = \frac{1}{q^m} \sum_{j=1}^{k/\ell} (\tilde{z}_{i_1+j\ell} - g(c(\tilde{z}_{i_1+j\ell}))) \ldots (\tilde{z}_{i_m+j\ell} - g(c(\tilde{z}_{i_m+j\ell}))) \tag{2}$$

We define $\hat{f} = \frac{1}{T'} \sum_{\tilde{z}:\text{useful}(\tilde{z})} f_{i_1,\ldots,i_m;\ell}(\tilde{z})$ as the average of the above equation over useful traces. We show that, for any set of $\tilde{\Theta}(n^6)$ traces from a string z, $\left|\hat{f} - S_{i_1,\ldots,i_m;\ell}(z)\right| \leq 1/3$ with probability at least 2/3, which implies an efficient tester.

The main result of this subsection requires the following lemma, which roughly states that we can recover clusters from a random trace with high probability.

▶ **Lemma 27.** *Let $z_1, \ldots, z_{2k} \in \mathbb{N}$, $q \in (0,1)$, and $\tilde{z}_j \sim \boldsymbol{Bin}(z_j, q)$ for all $j \in [k]$. Then there exists an absolute constant $C > 0$ such that, for any C-separated partition c of $qz_1, \ldots, qz_{2k}$, $|\tilde{z}_j - qz_j| \leq C\sqrt{n}\log n$ and $c(\tilde{z}_j) = c(qz_j)$ for all $j \in [k]$ with probability at least $1 - O_k(n^{-10})$.*

Proof. Fix $j \in [k]$. Let c be a C-separated partition of $qz_1, \ldots, qz_{2k}$. By a Chernoff bound, there exists a constant C such that $|\tilde{z}_j - qz_j| \leq C\sqrt{n}\log n$ with probability at least $1 - n^{-10}$. When this occurs, the following holds for all $j' \in [k]$ with $c(qz_{j'}) \neq c(qz_j)$:

$$|\tilde{z}_j - qz_{j'}| \geq |qz_j - qz_{j'}| - |\tilde{z}_j - qz_j| > 2C\sqrt{n}\log n - C\sqrt{n}\log n = C\sqrt{n}\log n \geq |\tilde{z}_j - qz_j|$$

In this case, clearly $c(\tilde{z}_j) = c(qz_j)$. The contrapositive of this is that, if $c(\tilde{z}_j) \neq c(qz_j)$, we must have $|\tilde{z}_j - qz_j| > C\sqrt{n}\log n$. Taking a union bound over the k random variables, the probability that this occurs for some $j \in [k]$ is at most kn^{-10}, as desired. ◀

We are now ready to prove the correctness of Algorithm 2.

Algorithm 2 TEST-SIMILAR-TRACES.

Input: $k \geq 0; x, y \in \mathbb{N}^k$; $q \in (0,1)$; $s \in \mathbb{R}^k$; $C > 0$; sample access to deletion channel
Output: x or y

$\ell \leftarrow$ minimum value such that s is symmetric mod ℓ
Let $S_{i_1,\dots,i_m;\ell}$ be the statistic from Lemma 25
Draw $T = \Theta(n^6 \log^{12} n)$ traces $\tilde{z}^{(1)}, \dots, \tilde{z}^{(T)}$ from the deletion channel
Let $\hat{f}$ be the average of $f_{i_1,\dots,i_m;\ell}(\tilde{z}^{(t)})$ (from Equation (2)) for all useful traces $\tilde{z}^{(t)}$
return $\arg\min_{z \in \{x,y\}} \left| \hat{f} - S_{i_1,\dots,i_m;\ell}(z - \frac{1}{q} g(s)) \right|$

▶ **Lemma 28.** *Let $x = (x_1, \dots, x_k)$ and $y = (y_1, \dots, y_k)$ be two cyclically distinct binary strings of length n, and $q \in (0,1)$. Let c be a C-separated partition of $qx_1, \dots, qx_k, qy_1, \dots, qy_k$, and suppose $c(qx_1), \dots, c(qx_k)$ is a cyclic shift of $c(qy_1), \dots, c(qy_k)$. Then for sufficiently large C and n, there exists an algorithm (Algorithm 2) which distinguishes x from y with probability at least $2/3$ using $T = O(n^6 \log^{12} n)$ traces from a cyclic deletion channel.*

Proof. Fix C to be the constant determined by Lemma 27. Let $\mathcal{Z}$ denote the set of useful traces (defined at the top of Section 4.3.2). The first usefulness condition holds with probability exactly q^k for each trace. By Lemma 27, we have $|\tilde{z}_j - qz_j| \leq C\sqrt{n}\log n$ and $c(\tilde{z}_j) = c(qz_j)$ with probability at least $1 - O(n^{-10})$. In this case, we have $|\tilde{z}_j - g(c(\tilde{z}_j))| \leq |\tilde{z}_j - qz_j| + |qz_j - g(c(\tilde{z}_j))| \leq C\sqrt{n}\log n + 4Ck\sqrt{n}\log n$, which is precisely the second condition. Therefore, for large enough n, each trace falls in $\mathcal{Z}$ with constant probability. By a Chernoff bound, $|\mathcal{Z}| = \Omega(T)$ with probability $1 - o(1)$. For each $\tilde{z} \in \mathcal{Z}$, the algorithm computes $f_{i_1,\dots,i_m;\ell}(\tilde{z})$, where $f_{i_1,\dots,i_m;\ell}$ is defined in Equation (2). Let A be the event that $\tilde{z} \in \mathcal{Z}$ and $c(\tilde{z}_j) = c(qz_j)$ for all $j \in [k]$. Conditioned on A, it is possible to compute $f(\tilde{z})$ precisely, since we can choose a consistent cyclic shift of s and match $\tilde{z}$ to that shift up to symmetry. We therefore have

$$\begin{aligned}
\mathbf{E}[f(\tilde{z}) \mid A] &= \mathbf{E}\left[\frac{1}{q^m}\sum_{j=1}^{k/\ell} (\tilde{z}_{i_1+j\ell} - g(c(qz_{i_1+j\ell}))) \dots (\tilde{z}_{i_m+j\ell} - g(c(qz_{i_m+j\ell})))\right] \pm O(n^{-4}) \\
&= \frac{1}{q^m}\sum_{j=1}^{k/\ell} \mathbf{E}[\tilde{z}_{i_1+j\ell} - g(c(qz_{i_1+j\ell}))] \dots \mathbf{E}[\tilde{z}_{i_m+j\ell} - g(c(qz_{i_m+j\ell}))] \pm O(n^{-4}) \\
&\qquad\qquad (\tilde{z}_j\text{s are independent}) \\
&= \frac{1}{q^m}\sum_{j=1}^{k/\ell} (qz_{i_1+j\ell} - g(s_{i_1+j\ell})) \dots (qz_{i_m+j\ell} - g(s_{i_m+j\ell})) \pm O(n^{-4}) \\
&= S_{i_1,\dots,i_m;\ell}(z - \frac{1}{q} g(s)) \pm O(n^{-4})
\end{aligned}$$

where the $\pm O(n^{-4})$ term in the first line comes from the $O(n^{-10})$ probability that there are k 1s in $\tilde{z}$ but event A fails; this is multiplied by the universal $O(n^6)$ bound on $|f(\tilde{z})|$ for $m \leq 6$.

By Lemma 27, A holds with probability at least $1 - O(n^{-10})$ conditioned on $\tilde{z} \in \mathcal{Z}$. Additionally, the second condition implies that for all $\tilde{z} \in \mathcal{Z}$, we have $|f(\tilde{z})| \leq \frac{k}{q^m}(4k+1)^6 C^6 n^3 \log^6 n$. Combining these facts and applying the law of total probability yields

$$\left| \mathbf{E}[f(\tilde{z}) \mid \tilde{z} \in \mathcal{Z}] - S_{i_1,\dots,i_m;\ell}(z - \frac{1}{q} g(s)) \right| \leq O\left(\frac{n^3 \log^6 n}{n^{10}} + n^{-4}\right) = o(1)$$

Combining the bound on $|f(\tilde{z})|$ with Hoeffding's inequality, we have

$$\mathbf{Pr}\left[\left|\hat{f} - \mathbf{E}[f(\tilde{z} \mid z \in \mathcal{Z})]\right| \geq \frac{1}{4}\right] \leq 2\exp\left(-\frac{2\,|\mathcal{Z}|^2}{16\,|\mathcal{Z}| \cdot \left(\frac{k}{q^m}(4k+1)^6 C^6 n^3 \log^6 n\right)^2}\right)$$
$$= 2\exp\left(-\frac{|\mathcal{Z}|}{16\frac{k^2}{q^{2m}}(4k+1)^{12} C^{12} n^6 \log^{12} n}\right)$$

By our choice of T, we have $|\mathcal{Z}| = \Omega(n^6 \log^{12} n)$ with probability $1 - o(1)$. Therefore, for an appropriate choice of constants, the above equation is bounded by $0.1 + o(1)$. By the triangle inequality, the following holds for large enough n with probability at least 0.8: $\left|\hat{f} - S_{i_1,\ldots,i_m;\ell}(z - \frac{1}{q}g(s))\right| \leq \frac{1}{4} + o(1) \leq \frac{1}{3}$. By Lemma 25, there must exist a statistic $S_{i_1,\ldots,i_m;\ell}$ such that $\left|S_{i_1,\ldots,i_m;\ell}(x - \frac{1}{q}g(s)) - S_{i_1,\ldots,i_m;\ell}(y - \frac{1}{q}g(s))\right| \geq 1$ (namely, the statistic chosen on Line 2). Therefore, if $z = x$, the algorithm returns x with probability at least 0.9 (Line 5). The same is true for y, which completes the proof. ◀

5 Lower Bounds

Our goal in this section is to upper-bound the distance between the distributions $\mathbf{Del}(x)$ and $\mathbf{Del}(y)$ for two strings x, y with that are permutations of each other and have identical low-order cyclic statistics, which will yield a lower bound for the number of traces needed to distinguish these strings. We first analyze the probability of observing a given trace a from $\mathbf{Del}(x)$, viewed as a polynomial in n. We show that the higher-order coefficients of this polynomial depend on the lower-order cyclic statistics of x. Using this fact, we bound the distance between probabilities from $\mathbf{Del}(x)$ versus $\mathbf{Del}(y)$. We conclude by providing two strings x, y with identical cyclic statistics up to order 4, proving our $\tilde{\Omega}(n^5)$ lower bound.

▶ **Lemma 29.** *Let $x = 10^{n+x_1}10^{n+x_2}\ldots10^{n+x_k}$ with x^* an integer upper bound on $x_1, \ldots, x_k$, and let a be any cyclic shift of $10^{a_1}10^{a_2}\ldots10^{a_k}$. Then for a circular deletion channel $\boldsymbol{Del}(x)$ with deletion probability p, letting $n' = pn$, and letting $b_j = a_j - n(1-p)$ for $j \in [k]$, we have, for $Sym(x; n, p, b, x^*)$ some symmetric function of $x_1, \ldots, x_k$, that depends on n, p, b, x^*, that*

$$\mathbf{Pr}_{\tilde{x}\sim\boldsymbol{Del}(x)}[\tilde{x} = a] = Sym(x; n, p, b, x^*) \sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=x_j+1}^{x^*} (n' - b_{j+i} + h), \tag{3}$$

where all indices are taken modulo k.

Proof. Since all cyclic shifts of a trace have equal probability under the deletion channel, we assume without loss of generality that $a = 10^{a_1}10^{a_2}\ldots10^{a_k}$. By definition of a circular deletion channel, and letting $|a|, |x|$ denote the length of the underlying binary strings a, x respectively, we have

$$\mathbf{Pr}_{\tilde{x}\sim\mathbf{Del}(x)}[\tilde{x} = a] = \frac{1}{|a|}\left(\sum_{i=1}^{k}\prod_{j=1}^{k}\binom{n+x_j}{a_{j+i}}\right) p^{|x|-|a|}q^{|a|} \tag{4}$$

where indices are taken modulo k. We point out that the terms outside the parentheses, $\frac{1}{|a|}p^{|x|-|a|}q^{|a|}$ are a symmetric function of x (depending only on $|x|$), and thus can be absorbed

into the initial $Sym(x; n, p, b, x^*)$ term in the lemma statement. We slightly refactor each binomial term as

$$\binom{n+x_j}{a_{i+j}} = \frac{n!}{a_{i+j}!(n+x^*-a_{i+j})!}\left(\prod_{h=1}^{x_j}(n+h)\right)\left(\prod_{h=x_j+1}^{x^*}(n-a_{i+j}+h)\right)$$

We thus need only consider the sum over i of the product over j of the above expression. The first term $\frac{n!}{a_{i+j}!(n+x^*-a_{i+j})!}$ does not depend on x so is by definition symmetric in x and can be absorbed into the initial $Sym(x; n, p, b, x^*)$. The next term, $\prod_{h=1}^{x_j}(n+h)$, when we take its product over all j, becomes a symmetric function of $x_1, \ldots, x_k$ and can thus also be absorbed into $Sym(x; n, p, b, x^*)$. The final term becomes the last term in Equation 3, after noting that $n - a_{i+j} = n' - b_{i+j}$, by definition of n', b. ◀

The only non-symmetric portion of Equation 3 is the expression inside the sum, which is

$$\sum_{i=1}^{k}\prod_{j=1}^{k}\prod_{h=x_j+1}^{x^*}(n'-b_{j+i}+h). \tag{5}$$

Thus, when we compare traces from $\mathbf{Del}(x)$ versus $\mathbf{Del}(y)$ for sequences x, y that are permutations of each other, it is this last term that will distinguish between them. We now view this term as a polynomial in n', and show that the high-order coefficients of this polynomial can be expressed in terms of low-order cyclic statistics of x. Thus for two sequences x, y with identical low-order cyclic statistics, we expect the distributions of $\mathbf{Del}(x)$ versus $\mathbf{Del}(y)$ to have high-order terms in n' that exactly cancel, leading to our main lower bound.

▶ **Lemma 30.** *Expression 5 when viewed as a polynomial in n' of degree $L = \sum_{j=1}^{k} x^* - x_j$, for any $m \geq 0$ has ${n'}^{L-m}$ coefficient that is a linear combination of degree $\leq m$ cyclic statistics of x, each multiplied by some symmetric function of x (possibly depending on n, p, b, x^*).*

Proof. Define S to be the set of pairs (j, h) such that $j \in [k]$ and $h \in \{x_j + 1, \ldots, x^*\}$, where $|S| = L$ as defined in the lemma. Thus the main expression of the lemma equals $\sum_{i=1}^{k}\prod_{(j,h)\in S}(n' - b_{j+i} + h)$. In particular, the coefficient of ${n'}^{L-m}$ in this expression can be found by multiplying the non-n' parts of all combinations of m *distinct* terms:

$$\sum_{i=1}^{k}\sum_{\substack{\text{distinct } (j_1,h_1),\\ \ldots,(j_m,h_m)\in S}}(h_1 - b_{j_1+i})\ldots(h_m - b_{j_m+i}) \tag{6}$$

For each fixed i, this expression is a *symmetric* polynomial of the terms $(h - b_{i+j})$ for $(j, h) \in S$, and we thus use Newton's symmetric polynomial identities to conclude that we may reexpress this expression for fixed i as a linear combination of products of "power sums" of these terms; the power sum of degree ℓ is defined as $\sum_{(j,h)\in S}(h - b_{j+i})^{\ell}$. Namely, Equation 6 can be expressed as some linear combination of the following expressions, where $\ell_1 \ldots, \ell_s$ are some positive integers that sum to m:

$$\sum_{i=1}^{k}\prod_{r=1}^{s}\sum_{(j,h)\in S}(h - b_{i+j})^{\ell_r} \tag{7}$$

We break down the inner sum of this expression using the definition of S, and the binomial expansion (across different powers t), as

$$\sum_{(j,h)\in S}(h-b_{i+j})^{\ell_r}=\sum_{j=1}^{k}\sum_{t=0}^{\ell_r}\binom{\ell_r}{t}(-b_{i+j})^{\ell_r-t}\sum_{h=x_j}^{x^*}h^t.$$

For any exponent t, we consider the final sum as a degree $t+1$ polynomial in x_j, with notation $P_{t+1}(x_j):=\sum_{h=x_j}^{x^*}h^t$. Thus Equation 7 equals

$$\sum_{i=1}^{k}\prod_{r=1}^{s}\left(\sum_{j_r=1}^{k}\sum_{t_r=0}^{\ell_r}\binom{\ell_r}{t_r}(-b_{i+j_r})^{\ell_r-t_r}P_{t_r+1}(x_{j_r})\right).$$

We pull the sums over t_r outside:

$$\sum_{t_1=0}^{\ell_1}\cdots\sum_{t_s=0}^{\ell_s}\sum_{i=1}^{k}\prod_{r=1}^{s}\left(\sum_{j_r=1}^{k}\binom{\ell_r}{t_r}(-b_{i+j_r})^{\ell_r-t_r}P_{t_r+1}(x_{j_r})\right)$$

We split the product by whether, for each r, we have $t_r=\ell_r$, since when this is true the expression simplifies significantly (note that we also move the sum over i to the right, past the first parenthetical, which does not depend on i):

$$\sum_{t_1=0}^{\ell_1}\cdots\sum_{t_s=0}^{\ell_s}\left(\prod_{r\in[s]:t_r=\ell_r}\sum_{j_r=1}^{k}P_{\ell_r+1}(x_{j_r})\right)\left(\sum_{i=1}^{k}\prod_{r\in[s]:t_r<\ell_r}\sum_{j_r=1}^{k}\binom{\ell_r}{t_r}(-b_{i+j_r})^{\ell_r-t_r}P_{t_r+1}(x_{j_r})\right)$$

The first parenthetical is clearly a symmetric function of $x_1,\ldots,x_k$. To analyze the second expression, we apply a variable substitution in the inner sum, replacing j_r with j_r-i. For any fixed tuple $t_1,\ldots,t_s$, the last parenthetical equals

$$\sum_{i=1}^{k}\prod_{r\in[s]:t_r<\ell_r}\sum_{j_r=1}^{k}\binom{\ell_r}{t_r}(-b_{j_r})^{\ell_r-t_r}P_{t_r+1}(x_{j_r-i}).$$

We view the product as a polynomial in the variables $x_{1-i},\ldots,x_{k-i}$, which we denote as $Q(x_{1-i},\ldots,x_{k-i})$; the degree of this polynomial is thus bounded by $\sum_{r:t_r<\ell_r}(t_r+1)$, which is at most $\sum_r\ell_r=m$. Crucially, the sum over i now equals $\sum_{i=1}^{k}Q(x_{1-i},\ldots,x_{k-i})$, which, since we sum a degree m polynomial over all cyclic shifts of $x_1,\ldots,x_k$, is thus clearly a linear combination of cyclic statistics of degree $\le m$.

In conclusion, Equation 7 is thus a sum of cyclic statistics of x each time some symmetric function of x. Thus, the coefficient of n'^{L-m} in Equation 5, being a linear combination of expressions of the form of Equation 7, is also a sum of cyclic statistics of x each time some symmetric function of x, as desired. ◀

We now directly compare the probabilities of getting a certain trace a from $\mathbf{Del}(x)$ versus $\mathbf{Del}(y)$.

▶ **Lemma 31.** *Let $x=10^{n+x_1}10^{n+x_2}\ldots10^{n+x_k}$ and $y=10^{n+y_1}10^{n+y_2}\ldots10^{n+y_k}$. Suppose $(x_1,\ldots,x_k)$ is a permutation of $(y_1,\ldots,y_k)$ and that the two sequences have matching cyclic statistics up to some order $z\ge1$. Let $a=10^{a_1}10^{a_2}\cdots10^{a_k}$. If $a_j-n(1-p)\in\left[-C\sqrt{n\log n},C\sqrt{n\log n}\right]$ for all $j\in[k]$ then for sufficiently large n,c depending on C, k, p, and $x^*=\max\{x_1,\ldots,x_k\}$ we have*

$$\frac{\mathbf{Pr}_{\tilde{x}\sim\mathbf{Del}(x)}[\tilde{x}=a]}{\mathbf{Pr}_{\tilde{y}\sim\mathbf{Del}(y)}[\tilde{y}=a]}\in\left[1-(c\log n/n)^{\frac{z+1}{2}},1+(c\log n/n)^{\frac{z+1}{2}}\right].$$

Proof. As above, define $b_j = a_j - n(1-p)$ and define $n' = pn$. Since x^* upper bounds $x_1, \dots, x_k$, we have that x^* upper bounds $y_1, \dots, y_k$ too, since they are a permutation of $x_1, \dots, x_k$. By Lemma 29, we have

$$\mathbf{Pr}_{\tilde{x}\sim\mathbf{Del}(x)}[\tilde{x} = a] = Sym(x; n, p, b, x^*) \sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=x_j+1}^{x^*} (n' - b_{j+i} + h)$$

and the corresponding expression for y. Let $D_1 \coloneqq \sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=x_j+1}^{x^*} (n' - b_{j+i} + h)$ and $D_2 \coloneqq \sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=y_j+1}^{x^*} (n' - b_{j+i} + h)$. Because $(y_1, \dots, y_k)$ is a permutation of $(x_1, \dots, x_k)$, we have

$$\frac{\mathbf{Pr}_{\tilde{x}\sim\mathbf{Del}(x)}[\tilde{x} = a]}{\mathbf{Pr}_{\tilde{y}\sim\mathbf{Del}(y)}[\tilde{y} = a]} = \frac{\sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=x_j+1}^{x^*} (n' - b_{j+i} + h)}{\sum_{i=1}^{k} \prod_{j=1}^{k} \prod_{h=y_j+1}^{x^*} (n' - b_{j+i} + h)} = \frac{D_1}{D_2} = 1 + \frac{D_1 - D_2}{D_2}$$

It remains only to show that $\frac{D_1 - D_2}{D_2} \in \left[-(c \log n/n)^{\frac{z+1}{2}}, (c \log n/n)^{\frac{z+1}{2}}\right]$.

Let $L \coloneqq kx^* - \sum_{j=1}^{k} x_j$. Then D_1, D_2 are both degree L polynomials in n' with leading coefficient k. For $m \geq 1$ we now bound the contribution of the terms of degree $\leq L - m$.

By assumption, each $b_j \in \left[-C\sqrt{n \log n}, C\sqrt{n \log n}\right]$; thus for $C' = C + x^*$ and $n \geq 2$ we trivially have that $|-b_j + h| \leq C'\sqrt{n \log n}$, for $h \in \{1, \dots, x^*\}$. Thus by definition of D_1, D_2, the contribution of the ${n'}^{L-m}$ term to either D_1 or D_2 has magnitude at most $k\binom{L}{m}(C'\sqrt{n \log n})^m {n'}^{L-m}$. We bound $\binom{L}{m} \leq L^m$, and also bound $L \leq kx^*$. We use these bounds to bound the contribution to D_1 or D_2 of *all* terms of degree $\leq L - m$ in n': this is bounded by the geometric series $k \sum_{i \geq m} (kx^* C' \sqrt{n \log n})^i {n'}^{L-i}$. We choose n large enough so that each term of this series is at most $\frac{1}{3}$ of the previous term, namely, we choose n so that $\frac{\sqrt{n}}{\log n} \geq 3\frac{kx^* C'}{p}$. Because the ratio of terms is $\leq \frac{1}{3}$, the $i = 0$ term of the series is at least twice as large as the sum of magnitudes the remaining terms combined, so that, since the $i = 0$ term of D_2 is $k(n')^L$, we conclude that $D_2 \geq \frac{1}{2} k {n'}^L$. Second, the sum of the series starting with the $i = m$ term has magnitude at most $\frac{3}{2}$ times the $i = m$ term, namely,

$$k \sum_{i \geq m} (kx^* C' \sqrt{n \log n})^i {n'}^{L-i} \leq \frac{3}{2} k (kx^* C' \sqrt{n \log n})^m {n'}^{L-m} = \frac{3}{2} k {n'}^L \left(\frac{(kx^* C')^2 \log n}{p^2 n}\right)^{\frac{m}{2}}$$

We now invoke Lemma 30, which says that the ${n'}^{L-m}$ coefficients of D_1, D_2 respectively are linear combinations of degree $\leq m$ cyclic statistics of x, y respectively, times symmetric functions of x, y respectively. The assumption that x, y are permutations of each other means that all symmetric functions of x equal the corresponding symmetric functions of y; the assumption that x, y have identical cyclic statistics up to order z further implies that the ${n'}^{L-m}$ coefficients of D_1, D_2 must be identical for $m \leq z$. Thus we bound $|D_1 - D_2|$ by plugging in the bounds of the previous paragraph starting at the first nonzero degree, $m = z + 1$: $|D_1 - D_2| \leq 2 \cdot \frac{3}{2} k {n'}^L \left(\frac{(kx^* C)^2 \log n}{p^2 n}\right)^{\frac{z+1}{2}}$. Combining with our lower bounds from above that $D_2 \geq \frac{1}{2} k {n'}^L$, and setting $c = 6\frac{(kx^* C)^2}{p^2}$, we conclude $\frac{D_1 - D_2}{D_2} \in \left[-(c \log n/n)^{\frac{z+1}{2}}, (c \log n/n)^{\frac{z+1}{2}}\right]$ as desired. ◀

We use the following bound relating the Hellinger distance between μ, ν to the TV distance between t samples from μ or ν.

▶ **Lemma 32** (Lemma A.5 in [13]). *Let μ and ν be probability measures with $d_H(\mu, \nu) \leq 1/2$. Then $1 - D_{TV}(\mu^t, \nu^t) \leq \varepsilon$ if $t \leq \frac{\log(1/\varepsilon)}{9 d_H(\mu, \nu)}$.*

▶ **Lemma 33.** *Let $x = 10^{n+x_1}10^{n+x_2}...10^{n+x_k}$ and $y = 10^{n+y_1}10^{n+y_2}...10^{n+y_k}$ be two strings where $(x_1, ...x_k)$ and $(y_1, ..., y_k)$ are permutations of each other and have identical cyclic statistics up to order z. Then any algorithm which distinguishes $\boldsymbol{Del}(x)$ from $\boldsymbol{Del}(y)$ with probability $\frac{2}{3}$ requires $\Omega(n^{z+1}/\log^{z+1} n)$ samples.*

Proof. Let μ_0 represent $\mathbf{Del}(x)$ conditioned on deleting only 0s; with ν_0 defined correspondingly for $\mathbf{Del}(y)$. The distributions μ_0, ν_0 are only supported on traces that are cyclic shifts of $a = 10^{a_1}10^{a_2}\cdots 10^{a_k}$, for some gaps $a_1, \ldots, a_k$.

As above, define $b_i = a_i - n(1-p)$. We point out that, by a standard Chernoff bound, we have $b_i \in [-C\sqrt{n\log n}, C\sqrt{n\log n}]$ with probability at least $1 - n^{-(z+100)}$ for some universal constant C (assuming z is a constant).

Combining Lemma 31 with this Chernoff bound yields the following bound on the Hellinger distance between μ_0 and ν_0:

$$
\begin{aligned}
d_H^2(\mu_0, \nu_0) &= \sum_{a_1,\ldots,a_k} \left(\sqrt{\mu_0(10^{a_1}...10^{a_k})} - \sqrt{\nu_0(10^{a_1}...10^{a_k})}\right)^2 \\
&= \frac{1}{(1-p)^k} \sum_{a_1,\ldots,a_k} \left(\sqrt{\mathbf{Pr}_{a'\sim\mathbf{Del}(x)}[a'=a]} - \sqrt{\mathbf{Pr}_{a'\sim\mathbf{Del}(y)}[a'=a]}\right)^2 \\
&\le \frac{2kn^{-(z+100)}}{(1-p)^k} + \frac{1}{(1-p)^k} \sum_{\substack{a_1,\ldots,a_k \in \\ [n(1-p)-C\sqrt{n\log n}, n(1-p)+C\sqrt{n\log n}]}} \mathbf{Pr}_{a'\sim\mathbf{Del}(y)}[a'=a] \left(1 - \sqrt{\frac{\mathbf{Pr}_{a'\sim\mathbf{Del}(x)}[a'=a]}{\mathbf{Pr}_{a'\sim\mathbf{Del}(y)}[a'=a]}}\right)^2 \\
&= O\left(\frac{\log^{z+1} n}{n^{z+1}}\right)
\end{aligned}
$$

We can generate a sample from $\mathbf{Del}(x)$ by first sampling from μ_0 and then passing the sample through a second channel which deletes only 1s (and similarly for $\mathbf{Del}(y)$ and ν_0). By the data processing inequality, the second channel cannot increase the Hellinger distance between μ_0 and ν_0, so we have $d_H(\mathbf{Del}(x), \mathbf{Del}(y)) \le d_H(\mu_0, \nu_0) = O\left(\frac{\log^{z+1} n}{n^{z+1}}\right)$. Applying Lemma 32, we conclude that one requires $\Omega\left(\frac{n^{z+1}}{\log^{z+1} n}\right)$ traces to distinguish x from y with constant success probability. ◀

Finally, we provide two strings that have identical cyclic statistics up to 4-th order, and whose gaps are permutations of each other. They are the sequence $x_{(j)} = (0, 2, 3, 2, 1, 1, 1, 1, 2, 3, 2, 0)$ and the sequence $y_{(j)} = 3 - x_{(j)}$. Thus, distinguishing these two strings requires $\Omega(\frac{n^5}{\log^5 n})$ traces.

References

1 Anders Aamand, Allen Liu, and Shyam Narayanan. Near-Optimal Trace Reconstruction for Mildly Separated Strings. In Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis, editors, *52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025)*, volume 334 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 3:1–3:20, Dagstuhl, Germany, 2025. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.ICALP.2025.3.

2 Frank Ban, Xi Chen, Adam Freilich, Rocco A Servedio, and Sandip Sinha. Beyond trace reconstruction: Population recovery from the deletion channel. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 745–768. IEEE, 2019. doi:10.1109/FOCS.2019.00050.

3 Tugkan Batu, Sampath Kannan, Sanjeev Khanna, and Andrew McGregor. Reconstructing strings from random traces. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '04, pages 910–918, USA, 2004. Society for Industrial and Applied Mathematics. URL: http://dl.acm.org/citation.cfm?id=982792.982929.

4 Joshua Brakensiek, Ray Li, and Bruce Spang. Coded trace reconstruction in a constant number of traces. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 482–493. IEEE, 2020. doi:10.1109/FOCS46700.2020.00052.

5 Arnav Burudgunte, Paul Valiant, and Hongao Wang. New bounds for circular trace reconstruction, 2025. arXiv:2512.02412.

6 Zachary Chase. New lower bounds for trace reconstruction. In *Annales de l'Institut Henri Poincaré-Probabilités et Statistiques*, volume 57, pages 627–643, 2021.

7 Zachary Chase. Separating words and trace reconstruction. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 21–31, 2021. doi:10.1145/3406325.3451118.

8 Xi Chen, Anindya De, Chin Ho Lee, and Rocco A. Servedio. Trace Reconstruction from Local Statistical Queries. In Amit Kumar and Noga Ron-Zewi, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024)*, volume 317 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 52:1–52:24, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.APPROX/RANDOM.2024.52.

9 Kuan Cheng, Elena Grigorescu, Xin Li, Madhu Sudan, and Minshen Zhu. On k-mer-based and maximum likelihood estimation algorithms for trace reconstruction. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pages 879–884, 2024. doi:10.1109/ISIT57864.2024.10619392.

10 Mahdi Cheraghchi, Ryan Gabrys, Olgica Milenkovic, and Joao Ribeiro. Coded trace reconstruction. *IEEE Transactions on Information Theory*, 66(10):6084–6103, 2020. doi:10.1109/TIT.2020.2996377.

11 Anindya De, Ryan O'Donnell, and Rocco A Servedio. Optimal mean-based algorithms for trace reconstruction. *The Annals of Applied Probability*, 29(2):851–874, 2019.

12 Elena Grigorescu, Madhu Sudan, and Minshen Zhu. Limitations of Mean-Based Algorithms for Trace Reconstruction at Small Edit Distance. *IEEE Transactions on Information Theory*, 68(10):6790–6801, 2022. doi:10.1109/TIT.2022.3168624.

13 Nina Holden and Russell Lyons. Lower bounds for trace reconstruction. *The Annals of Applied Probability*, 30(2):503–525, April 2020. doi:10.1214/19-AAP1506.

14 Thomas Holenstein, Michael Mitzenmacher, Rina Panigrahy, and Udi Wieder. Trace reconstruction with constant deletion probability and related results. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 389–398, San Francisco, California and USA, 2008. Society for Industrial and Applied Mathematics. URL: http://dl.acm.org/citation.cfm?id=1347082.1347125.

15 Akshay Krishnamurthy, Arya Mazumdar, Andrew McGregor, and Soumyabrata Pal. Trace reconstruction: Generalized and parameterized. *IEEE Transactions on Information Theory*, 67(6):3233–3250, 2021. doi:10.1109/TIT.2021.3066010.

16 Vladimir I Levenshtein. Efficient reconstruction of sequences. *IEEE Transactions on Information Theory*, 47(1):2–22, 2002. doi:10.1109/18.904499.

17 Shyam Narayanan and Michael Ren. Circular trace reconstruction. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*, pages 18:1–18:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021. doi:10.4230/LIPIcs.ITCS.2021.18.

18 Fedor Nazarov and Yuval Peres. Trace reconstruction with $\exp(O(n^{1/3}))$ samples. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2017, pages 1042–1046, New York, NY, USA, 2017. Association for Computing Machinery. doi:10.1145/3055399.3055494.

19 Yuval Peres and Alex Zhai. Average-Case Reconstruction for the Deletion Channel: Subpolynomially Many Traces Suffice. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 228–239, Berkeley, CA, October 2017. IEEE. doi:10.1109/FOCS.2017.29.

20 Joey Rivkin, Gregory Valiant, and Paul Valiant. A generalized trace reconstruction problem: Recovering a string of probabilities. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, STOC '25, pages 1657–1667, New York, NY, USA, 2025. Association for Computing Machinery. doi:10.1145/3717823.3718315.

21 Jin Sima and Jehoshua Bruck. Trace Reconstruction with Bounded Edit Distance. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 2519–2524, Melbourne, Australia, 2021. IEEE Press. doi:10.1109/ISIT45174.2021.9518244.